Decentralised Wireless Data Dissemination for Vehicle-to-Vehicle Communications

Debra Ann Topham

A thesis submitted to
The University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

Electronic, Electrical and Computer Engineering
College of Engineering and Physical Sciences
The University of Birmingham
5th December 2011

ABSTRACT

This thesis is concerned with inter-vehicle communications supporting the deployment of future safety-related applications. Through use case analysis of the specific communications requirements of safety related and traffic efficiency applications, a data dissemination framework is proposed that is able to meet the various message delivery requirements. More specifically, this thesis focuses on the subset of the proposed framework, which provides geocasting, i.e. addressing a geographical area on the road network, and local zone connectivity, providing neighbour awareness, for safety related applications.

The enabling communications technology for inter-vehicle communications based on IEEE 802.11 wireless local area network devices and the associated lack of reliability it presents for the distribution of safety messages in broadcast mode, form the main topic of this thesis. A dissemination scheme for safety related inter-vehicular communication applications, using realistic vehicular traffic patterns, is proposed, implemented and evaluated to demonstrate mechanisms for efficient, reliable and timely delivery of safety messages over an unreliable channel access scheme.

The original contribution of this thesis is to propose a novel data dissemination protocol for vehicular environments, capable of simultaneously achieving significant economy of messaging, whilst maintaining near 100% reliable message delivery in a timely manner for a wide variety of highway traffic flow scenarios, ranging from sparsely, fragmented networks to dense, congested road networks. This is achieved through increased protocol complexity in inferring and tracking each vehicular node's local environment, coupled with implementing adaptation to both local data traffic intensity and vehicular density. Adaptivity is achieved through creating and employing an empirical channel access delay model and embedding the stochastic delay distribution in decisions made at the network layer; this method of adaptivity is novel in itself. Moreover, unnecessary retransmissions arising from the inherent uncertainty of the wireless medium are suppressed through a novel three-step mechanism.

To my husband, daughter and son,
parents and little sister.
To Tracey,
you are deeply missed and always in my thoughts.

ACKNOWLEDGEMENTS

Firstly, I would like to thank my supervisors Dr. Costas Constantinou and Dr. Theo Arvanitis, for their support, constructive discussions and endurance throughout this long, long process. Thank you to MIRA for supporting the PhD financially, Prof. Geoff Callow for support during the early stages and to Prof. David Ward for supporting and supervising this work throughout my time at MIRA.

I would like to thank Dr. Wu and Yiman Du from the Transportation Research Group at Southampton University for their time and support in the generation of the vehicle traffic flow files.

Thanks also to Dr. Edward Hoare who set me on the path to pursue a PhD in the first place. My life would have been considerably different had I not followed your advice.

I would like to thank Dr. Sridhar Pammu and Dr. David Checketts for their support with both OPNET and the SUN workstations over the years.

I would also like to thank all the past and present members of the Distributed Systems and Networks Laboratory, especially Bin, Janaid, Rebecca, Keita, Paul, David, Amalia, Hani and Yuri.

Thanks also to Prof. Chris Baber in his capacity as welfare tutor for his moral support during the difficult interludes.

Many thanks to my parents, sister and parents in-law for their love support and encouragement over the years, it has been greatly appreciated.

Finally, a very special thank you to my husband and daughter for their love, support and patience throughout the years, particulary to my daughter Sophia, who has grown up continually asking the question “when is mummy going to finish her PhD?” - I hope I haven’t put you off pursuing a PhD sometime in your future! To Alexandros, my little baby boy, who will not have to ask this question.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS AND ABBREVIATIONS

ACK	Acknowledgement
ACC	Adaptive Cruise Control
ACK	Acknowledgement
ACW	Anti-clockwise
ADAS	Advanced Driver Assistance Systems
ASTM	American Society for Testing and Materials
C2C-CC	Car-to-Car Communications Consortium
CBF	Contention Based Forwarding
CCH	Control Channel
CEN	European Committee for Standardisation
CLW	Clockwise
COIN	Clustering for Open IVC Networks
CS	Carrier Sensing
CSMA	Carrier Sense Multiple Access
CA	Collision Avoidance
CTB	Clear To Broadcast
CTS	Clear to Send
CW	Contention Window
DABS	Digital Audio Broadcast System
DCF	Distributed Coordination Function
DDA	Data Dissemination Area
DDF	Distance Deferral Forwarding
DDT	Distance Deferral Transmission
DFPAV	Distributed Fair Power Adjustment for Vehicular environments
DIFS	Distributed Interframe Spacing
DOT	Department of Transport
DPP	Directional Propagation Protocol
DRG	Distributed Robust Geocast
DSDV	Dynamic Destination-Sequenced Distance-Vector
DSRC	Dedicated Short Range Communications
DSRC-ITS	Dedicated Short Range Communications for Intelligent Transportation Systems

DSR	Dynamic Source Routing
DSSS	Direct Sequence Spread Spectrum
DTN	Delay Tolerant Networks
DTRA	Dynamic Transmission Range Assignment
DV-CAST	Distributed Vehicular Broadcasting
DVDE	Distributed Vehicle Density Estimation
EC	Equivalent Cell
ECH	Equivalent Cell Header
EDCA	Enhanced Distributed Channel Access
EIRP	Effective Isotropically Radiated Power
EMDV	Emergency Message Dissemination for Vehicular Environments
ETRRS	Enhanced Time Reservation-based Relay Node Selection
ETSI	European Telecommunications Standards Institute
FB	Fast Broadcast
FCC	Federal Communications Commission
FDA	Forwarding Decision Algorithm
FIFO	First In First Out
FSM	Finite State Machine
FZ	Forwarding Zone
GPS	Global Positioning System
GUI	Graphical user Interface
HGV	Heavy Goods Vehicle
ICI	Interface Control Information
ID	Identification
IEEE	Institute of Electrical and Electronic Engineers
ITS	Intelligent Transportation Systems
IVC	Inter-vehicle Communications
IVG	Inter-vehicle Geocast
LBM	Location-based Multicast
LGMS	Localised Group Membership Service
LPG	Local Peer Group
LPG-LEC	Local Peer Group Linked Equivalent Cells
LPG-RO	Local Peer Group Relative Ordering
LS	Location Service
LZ	Local Zone
LZR	Local Zone Routing
MAC	Multiple Access Communication
MANET	Mobile *ad hoc* Networks
MBL	Maximum Beaconing Load
MHVB	Multi-hop Vehicular Broadcast
MIMO	Multple-Input Multiple-Output
NAV	Network Allocation Vector
OAPB	Optimised Adaptive Broadcast
ODAM	Optimised Dissemination of Alarm Messages

OFDM	Orthogonal Frequency Division Multiplexing
OSI	Open Systems Interconnection
PCF	Point Coordination Function
PDF	Probability Density Function
PHY	Physical Layer of IEEE 802.11
***p*-IVG**	Probabilistic Inter-vehicle Geocast
PVGF	Perimeter Vehicle Geographic Forwarding
PVLZR	Perimeter Vehicle Local Zone Routing
QoS	Quality of Service
RBM	Role Based Multicast
RDS	Radio Data System
REACT	Routing for Emergency Applications in Car-to-car networks using Trajectories
REAR	Receipt Estimation Alarm Routing
RPC	Relative Position Classification
RSU	Road Side Unit
RTB	Request to Broadcast
RTS	Request to Send
RZR	Routing Zone of Relevance
SAE	Society of Automotive Engineers
SCH	Service Channel
SIFS	Short Interframe Space
STD	State Transition Diagram
TDA	Topology Discovery Algorithm
TRADE	TRAcking DEtection
TRRS	Time Reservation-based Relay Node Selection
TTL	Time to Live
UDG	Unit Disk Graph
UMB	Urban Multi-Hop Broadcast
V2V	Vehicle-to-Vehicle
V2I	Vehicle-to-Infrastructure
VANET	Vehicular *ad hoc* networks
VSC	Vehicle Safety Communications
WAVE	Wireless Access in Vehicular Environments
WLAN	Wireless Local Area Network
XML	Extensible Markup Language
ZRP	Zone Routing Protocol

MAIN SYMBOLS AND NOTATION

Notation	Definition
$a \in A$	The element a is a member of the set of A
$a \notin A$	The element a is not a member of the set of A
$A \subset B$	The set of A is a subset of B
$A \cap B$	All elements belonging to both sets A and B
$\forall$	For all
$\mid$	Such that, e.g. if $A = \{x \mid x$ is a positive, even integer $\}$ then A equals the set of all x such that x is a positive integer and even
$\{\emptyset\}$	Empty set
$X \wedge Y$	Either X or Y is true, or both
$X \vee Y$	Both X and Y are true
Ψ	Space on the Euclidean Plane
P_{x_i}	General position vector in Ψ
$\parallel P_u P_v \parallel$	Euclidean distance between two arbitrary points in Ψ
x_i	Node label
R	Maximum transmission range of a node
N	Set of all node labels in discrete space
S	Source or originating node of warning message
f_p	Function mapping node location in Ψ to its address in the discrete space
f_p^{-1}	Inverse mapping function mapping a nodes address in discrete space to its position in Ψ
LZ	The area covered by the transmission radius of a node
LN	Set of one-hop neighbouring node labels within a nodes LZ
FZ	Area of the LZ in the forwarding direction between R and a node's current position
FN	Set of one-hop neighbouring node labels within a nodes FZ
F_b	Forwarding boundary of data dissemination area

Notation	Definition
P_{F_b}	Position of F_b defined by S
$\overrightarrow{F_{dir}}$	Vector defining forwarding direction of message dissemination
t_{def}^{for}	DDF deferral transmission time for forwarding nodes
t_{int}^{for}	DDF deferral transmission time for intermediate nodes
τ_{MAC}	Stochastic variable describing channel access delay at a node
$\bar{\tau}_{MAC}$	Mean MAC channel access delay within a node's LZ
σ_{MAC}	Standard deviation (S.D) of MAC channel access delay within a node's LZ
τ_{proc}	Packet processing time
κ	Time dilation factor
τ_{jit}	Random jitter time drawn from the empirical MAC delay pdf
$G_{LZ}^{x_i}$	Offered traffic within a nodes LZ
Υ	Length of sliding window over which $G_{LZ}^{x_i}$ is determined
$\bar{\eta}$	Mean neighbour nodes within a nodes LZ during Υ
$\Phi_{M_{beacon}}$	Inter-arrival frequency of M_{beacon}
M_{DDF}	DDF warning message
M_{Supp}	DDF suppression packet
M_{beacon}	DDF beacon packet
$\Gamma_{M_{Supp}}$	Sum of all M_{Supp} received within Υ
$\Gamma_{M_{DDF}}$	Sum of all M_{DDF} received within Υ
$\Gamma_{M_{DDF}}$	Sum of any other packet types received within Υ

CHAPTER 1

INTRODUCTION

The increase in the number of vehicles, the demand for a higher level of safety and improved transportation coupled with the need to reduce the impact of transportation on the environment, has led to governments and various stakeholders promoting the need for Intelligent Transportation systems (ITS).

The European Commission set out a goal in 2001 to reduce fatalities on European roads by 50% by 2010. Similarly, in 2008 the U.S. Department of Transportation (DOT) published their vision for transportation [1] and challenged industry to reduce 90% of road traffic crashes by 2030. Governmental bodies, industry and academia have come together to address these challenges forming research initiatives such as, the "e-Safety" initiative (now known as the iMobility Forum) [2] through the COMeSafety program in Europe [3] and the IntelliDrive program in the U.S. [4].

Many of the proposed applications have a minimal communication latency requirement and can only realize their full potential through the use of vehicle-to-vehicle or vehicle-to-infrastructure communications, referred to as inter-vehicle communications (IVC). These approaches employ wireless communication as the enabling technology. Inter-vehicle communication can be used to disseminate information in a cooperative manner in order to provide, hazard warnings, information on current traffic situations, or for infotainment purposes. Although wide area broadcasting systems, such as DABS and RDS and cellular communication systems, provide means to alert drivers to hazards, and in the case of cellular systems provide basic infotainment services, they are not directly suited for IVC because of the latency involved in coordinating communication via a centralised system [5, 6].

In order to facilitate IVC a decentralised self-organising mobile *ad hoc* wireless network (MANET) can be used. Vehicles are able to organise themselves into a multi-hop network enabling vehicles that are outside direct communication range to communicate forming a

vehicular *ad hoc* network (VANET) [5, 6]. The enabling communications technology that has been proposed by various research initiatives for VANETs is based on IEEE 802.11 WLAN technology and is in the process of being standardised for vehicular environments at a global level [5].

Standardisation activities for the overall ITS system architecture and communication framework are coordinated by a number of entities which involve the IEEE (IEEE 802.11p and 1609 working groups) in the US, and ETSI [7] and CEN [8] with cooperation from the C2C-CC [9] and COMeSafety [3] programmes in Europe.

1.1 Overview and Challenges of IVC Communications

The provision of IVC using VANET technology has the potential to increase the benefits of ITS in facilitating vehicles exchanging locally relevant information. Applications employing IVC include collision warning, incident warning, traffic control and infotainment, to name but a few. Collision avoidance systems can provide advanced warning of obstructions beyond the visual range of the driver and incident warnings can prevent motorway multiple collisions. Traffic control applications can improve road efficiency through traffic management in order to reduce congestion. Moreover, IVC can also aid the driver in dynamically selecting a route to their destination based on latest traffic awareness information. Such applications have varying message delivery requirements.

Safety applications have emerged as providing the greatest interest to research initiatives since IVC can provide the greatest impact within this area. Moreover, the timeliness of delivery for such applications is a driving factor in defining the requirements in the standardisation activities of the wireless communication technology and overall system architecture.

IVC in combination with on board sensors can be used to support road safety applications through the exchange of periodic beacon messages and the dissemination of event-based messages [10]. Beacon messages contain information such as position and identification. The exchange of beacon messages between neighbouring vehicles allows a node[1] to establish a picture of its local surroundings enabling it to detect potentially hazardous conditions such as lane change manoeuvres or sudden breaking. In the case of event-driven messages, also referred to as safety messages [10], a vehicle which is involved in, or detects, an accident issues a safety warning message which is disseminated through the VANET within the affected region of the road network. In the case of event-driven messages the safety

[1]The term *node* and *vehicle* are used to mean the same thing and are used interchangeably in this thesis.

warning message could also originate from roadside nodes (also known as roadside units (RSUs)) which detect dangerous driving conditions such as icy or foggy conditions, etc.

It is widely acknowledged [10, 11, 12] that broadcasting will play an important role in the dissemination of both beacon and event-driven messages. In order to avoid congesting the bandwidth in broadcast mode, IEEE 802.11 based communication technology does not implement packet acknowledgments, packet retransmission and medium reservation, which all mitigate for an unreliable communications medium. This imposes a challenge on achieving high reliability and efficiency, specifically in relation to dense vehicular networks as a result of contention and interference resulting from nodes competing to access the radio communication medium.

In addition to the unreliable medium access scheme and varying application message delivery requirements, the operating characteristics of the vehicular environment also need to be considered in IVC. The vehicular environment is characterised by the highly dynamic nature of traffic flow, the bounded mobility of vehicles to the road network and the availability of positioning information from onboard positioning systems (e.g. GPS). The positioning information can be used in IVC to restrict the dissemination of a safety message to an application specific area. The two extremes of vehicle traffic dynamics from highly congested to high speed and sparsely distributed vehicular traffic, introduce challenges in overcoming channel congestion and network fragmentation problems respectively.

Thus achieving high communication reliability and efficiency in the face of an unreliable medium access scheme is an essential requirement for safety based ITS applications. Moreover, the radio communication channel between vehicles is subject to doppler, multipath and shadowing effects, and can have a major impact on data packet transmission reliability [13]. Such radio propagation effects are only partially mitigated for by physical layer protocols [14].

1.2 Thesis Aims and Objectives

The aim of this thesis is to propose, implement and evaluate a message dissemination mechanism for various IVC applications through considering the specific challenges of operating in the vehicular environment. The enabling communications technology for IVC based on IEEE 802.11 WLAN devices and the unreliability it presents for the distribution of safety messages in broadcast mode are investigated. A dissemination scheme for safety related IVC applications, using realistic vehicular traffic patterns, is implemented to demonstrate mechanisms for efficient, reliable and timely delivery of safety messages over an unreliable channel access scheme.

The main objectives of this thesis are to addresses the following open research questions:

- Can the proposed protocol be as reliable as possible in comparison to other competing IVC dissemination schemes at the same time as being more economical with messaging overheads and without incurring a significant penalty on delay?
- Can the proposed protocol demonstrably avoid the broadcast storm problem through limiting unnecessary retransmissions whilst still maintaining a reliable delivery ratio to addressed nodes within a geographic region, in addition to meeting delay constraints of IVC safety applications?
- Can the proposed protocol ensure reliable dissemination when exposed to the hidden terminal problem?
- Can the above challenges be met for road networks with varying vehicular traffic ranging from sparsely connected with frequent partitions to congested roads?

1.3 Thesis Contributions

A key novel contribution of this thesis is the development and implementation of a data dissemination scheme for safety related IVC applications called the data dissemination forwarding (DDF) protocol, which meets stated objectives. The DDF protocol is loosely based on the notion of contention-based forwarding according to distance, for the provision of directed and restricted broadcasting of safety messages. Each node makes forwarding decisions independently to avoid coordination overheads, based on implicit knowledge of local connectivity information which classifies neighbouring vehicles according to their position vector. The local connectivity information also provides necessary data for cooperative collision avoidance applications given the sharing of speed and position information.

Local variations in channel access delay can severely affect the functioning of distance-based forwarding schemes leading to unnecessary retransmissions and hence unnecessary messaging overheads. The DDF protocol provides ordered delivery according to distance in the presence of channel access delay variability. Moreover, the DDF protocol is able to adapt to channel access delay variability by monitoring vehicle traffic density and data packet intensity which is used to adapt retransmission forwarding times to local conditions in order to avoid unnecessary broadcasts. The proposed DDF protocol adaptation mechanism is novel, as it incorporates the stochastic MAC delay distribution in forwarding decisions made at the network layer.

Through an in-depth study on the conditions which could cause the dissemination process to collapse under various scenarios, such as that caused by the hidden terminal problem,

forwarding persistence is implemented to avoid such occurrences. Forwarding persistence ensures reliable delivery throughout the addressed geographic region. Reliability is implemented at the cost of adding extra complexity to the DDF protocol, however, this cost is far outweighed by the increased reliability.

Furthermore, the protocol is able to overcome network fragmentation which allows the DDF protocol to operate under fast moving, sparsely connected vehicular traffic flow conditions. Moreover, in the case where channel contention issues are preventing messages from being forwarded, in particular when vehicle density is high, a 'soft partition' mechanism is implemented. The soft partition mechanism is a novel concept that ensures the forwarding process does not terminate prematurely, in addition to preventing unnecessary retransmissions from adding to the local data congestion.

Moreover, a significant economy of messaging is achieved, not only through the adaptability of the DDF protocol, but also through a novel mechanism which suppresses any erroneous retransmitting nodes. The suppression mechanism limits the number of retransmissions from forwarding nodes which have not detected successful message dissemination beyond their current location, as a result of channel contention or collision issues.

An empirical study of the distribution of MAC access delays through the simulation of one-hop broadcast communications arising from the joint CSMA MAC and vehicle mobility process is presented. The distribution of MAC access delay provides the parameters used within the DDF protocol allowing it to adapt to local variations in vehicle density and data packet traffic intensity. The novelty of this study lies in the fact that it models the distribution of MAC delays as opposed to just the mean and standard deviation values.

In addition a data dissemination framework is proposed which meets the dissemination requirements for a variety of safety related and traffic efficiency applications. The DDF and local connectivity protocols form a branch of the requirements of this framework.

Finally, performance of the DDF protocol is evaluated through simulation using realistic vehicular traffic mobility patterns and compared against a similar protocol called ODAM and a basic flooding algorithm. A detailed comparative performance analysis using a variety of performance metrics is provided.

1.4 Thesis Organisation

In Chapter 2 IVC applications are introduced and predominant enabling communications access technology and current standardisation initiatives are discussed. The operation of IEEE 802.11 technology is a central enabling technology and provides challenges in implementing IVC. The operation of IEEE 802.11 is discussed and its impact when operating

in broadcasting mode considered. Routing methods used in the MANET field are introduced and their suitability in the VANET environment are discussed given application requirements and challenges of operating the vehicular environment. Through use case analysis of the specific communications requirements of safety related and traffic efficiency applications, a data dissemination framework is proposed which is able to meet the various message delivery requirements. Chapter 2 restates, in more detail, the goal of this thesis which concentrates on the portion of the data dissemination framework that delivers event driven safety messages using a geocasting technique along with the exchange of beacon messages to maintain local connectivity knowledge.

Chapter 3 provides a review of methods proposed in the literature for the provision of IVC for dissemination of safety related messages using broadcasting and geocasting techniques. More specifically, this thesis focuses on methods that limit congestion and overcome network fragmentation motivating the requirement to implement data dissemination schemes which are able to adapt to local variations in both vehicle traffic density and data traffic intensity.

In Chapter 4 a novel data dissemination protocol called data dissemination forwarding (DDF)is presented. Design decisions taken in specifying the proposed data dissemination protocol are introduced and justified. The remainder of this chapter provides a detailed description of the operation the DDF protocol and the method used in developing the calculation which allows the DDF protocol to adapt to local MAC access delay variations.

Chapter 5 presents the methodology used to evaluate the performance of the proposed protocol and a description of the tools which are used in the simulation environment.

In Chapter 6 an empirical analysis of the distribution of MAC channel access delay based on the simulation of the periodic exchange of beacon messages, between one hop neighbouring nodes is presented. The simulation is implemented using the road traffic network presented in Chapter 5 and the beacon exchange mechanism of the DDF protocol presented in Chapter 4. The variation of the MAC delay is investigated with varying vehicle densities and beacon interarrival rates.

In Chapter 7 comparison protocols and their implementation against which the DDF protocol will be compared, and the scenario used to evaluate the performance of each protocol is presented. The performance of the DDF protocol is evaluated over a number of scenarios with varying traffic density, size of data dissemination area (DDA) and locations around the simulated road traffic network. The evaluation results are presented and the performance of DDF protocol in relation to the comparison protocols is discussed in detail.

Finally, in Chapter 8 the thesis is concluded with a summary of the work carried out, and a discussion of the main findings and possible avenues for future work.

CHAPTER 2

INTERVEHICLE COMMUNICATIONS: APPLICATIONS, MEDIUM ACCESS AND DATA DISSEMINATION

2.1 Introduction

To achieve the vision of increased safety and efficiency on the road introduced in Chapter 1, time-sensitive, safety-critical applications in vehicular networks form an intrinsic part of this requirement. In order to realise this vision various governments around the world have allocated protected spectrum dedicated to vehicle-to-vehicle and vehicle-to-roadside applications. Coupled with the dedicated spectrum allocation and the maturity of IEEE 802.11 Wireless Local Area Network (WLAN) systems this led to governments, industry and academia carrying out various research initiatives which utilised IEEE 802.11 as the basis for the enabling communication technology. The results from various research initiatives feed into efforts being carried out by various standardisation bodies in defining basic system architectures and protocols for cooperative vehicular applications.

This chapter briefly considers the range of applications which can be deployed to increase road safety and efficiency using vehicle-to-vehicle communication technology. A review of cooperative vehicular standardisation activities in the US and EU is given which focuses on developments based on IEEE 802.11, since the deployment of such applications depends on fast access to the communication medium. The communications access mechanism proposed by standardisation bodies is described in more detail, since it has a direct impact on the research presented in this thesis in Chapters 4 and 6. The second part of this chapter presents a data dissemination framework which meets the requirements of a sub-set of cooperative vehicular applications. This chapter ends by introducing the application focus and data dissemination requirements given the challenges of the vehicular environment

coupled with the communication mechanisms proposed by standardisation bodies.

2.2 IVC Applications

In order to realise the aim of ITS there are many applications, such as advanced driver assistance systems (ADAS) and traffic management and information systems which can be implemented using decentralised peer-to-peer communication technology. Current ADAS systems such as adaptive cruise control (ACC) are autonomous in that they do not communicate with adjacent vehicles. ACC systems have a limited view of their surroundings (∼200m) and are not able to react to a situation that has occurred outside their field of view. However, by incorporating IVC, vehicles can communicate surrounding information to each other, allowing them to increase their field of view and react accordingly. Future ADAS systems could be used to implement automated driving scenarios for vehicle platooning, including coordination of manoeuvres and motorway merging. Traffic management applications could be realised without requiring centralised control through the coordination of vehicular data using IVC. Another category of applications to improve driver comfort is 'nice-to-have' services such as Internet access and local tourist information. Depending on the goal of these applications they can be categorised as providing safety or non-safety related information. Safety related applications provide local awareness knowledge as well as warning other drivers of incidents. Non-safety related applications provide 'nice-to-have' services such as remote diagnostics.

In [15] various applications scenarios were defined, which are shown in Table 2.1. The communication requirements for the different applications were considered through use case analysis (included in Appendix A) in order to define a data dissemination framework for a sub-set of these applications which are presented in §2.4. Typical message characteristics from [16, 17] have also been included in Table 2.1 for each application type. The message characteristics show the minimum transmission frequency, delivery type, maximum allowed per hop latency (which varies from 50 ms to 500 ms) and possible modes of communication for their respective applications.

In Europe the ETSI Technical Committee on ITS has identified in [17] a set of applications and use cases for cooperative vehicular systems in cooperation with the Car-to-Car Communications Consortium (C2C-CC) [18] and results from various EU research projects coordinated via the COMeSafety initiative [3] which are to be considered as a reference for standardisation and deployment activities. Four application classes are specified in [17]: active road safety; cooperative traffic efficiency; cooperative local services and global internet services. Active road safety applications are further divided into cooperative awareness and road hazard warning applications. Cooperative awareness applications are based on

the communication of periodic messages whereas road hazard warning applications are based on the communication of event driven applications triggered by specific events.

Application	Scenario	Message Delivery Type	Communication Mode	Min. Trx Frequency (Hz)	Critical Latency (ms)
Platooning	Intentions of lead vehicle Platoon objectives (set speed, etc.) Preceding and following vehicle data Coordination of lane change Merging of platoons Requests to leave or join a platoon Emergency requests to leave a platoon	Periodic Broadcast/unicast	Ad hoc, Infrastructure, V2V, V2I	2	< 100
Cooperative driving	Requests to schedule on/off ramp departure lane change etc Vehicle speed, heading and position Vehicle intentions	Periodic event driven broadcast, unicast	Ad hoc V2V	10	< 100
Collision warning	Location of conflicting object or vehicle	Periodic broadcast,unicast	Ad hoc, infrastructure, V2V, V2I	10	< 50
Collision avoidance	Negotiation of resolution actions	Event-driven broadcast, unicast	Ad hoc, infrastructure, V2V, V2I	10	< 100
Incident warning	Location and type of incident	Event-driven time limited geocast	Ad hoc, infrastructure, V2V, V2I	10	< 100
Floating car data	Hazardous location Weather conditions Traffic flows Congestion information	Event-driven time-limited geocast	Ad hoc V2V	1 - 10	< 100
Emergency vehicle	Direction and speed of approaching vehicle	Event-driven broadcast, possible geocast	Ad hoc, V2V	10	< 100
Traffic control	Traffic monitoring Route planning	Periodic broadcast, unicast	Ad hoc, Infrastructure, V2V, V2I	1	< 100
Remote diagnostics	ECU software upgrades Diagnostic scan Remote service actions	Event-driven, broadcast, unicast	Infrastructure, ad hoc, V2I, V2V		
Mobile internet	IP-based services, e.g. email				
Location based services	Details of local hotels, restaurants, weather, etc. Emergency calls (breakdown, medical) Local mapping updates	Broadcast, unicast on-demand	Infrastructure, Ad hoc, V2I, V2V, cellular	1	< 500
Mobile vending services	Selling of value-add items, e.g. music, games, data Secured financial transactions				
Vehicle-to-vehicle chatting	Gaming, Instant messaging				

Table 2.1: Summary of IVC applications

In the US the Vehicle Safety Communications (VSC) project compiled a comprehensive list of communication-based vehicle safety and non-safety application scenarios [19]. More than 75 application scenarios were identified and analysed resulting in 34 safety and 11 non-safety application scenario descriptions.

2.3 Communication Technology

This section considers the standardisation activities in the US and EU, in the context of vehicular communications, and focus on the access communication technology that is able to meet the communication latency requirements of the cooperative vehicle safety applications mentioned in §2.2.

2.3.1 Spectrum Allocation and Standards

In 1999 motivated by the need to reduce the number of vehicle accidents and to increase the efficiency of transportation systems, the US Federal Communications Commission (FCC) allocated 75 MHz of bandwidth in the 5.9 GHz band specifically to support dedicated short range communications (DSRC) for ITS. The 75 MHz of DSRC-ITS radio spectrum allocation is between 5.850-5.925 GHz and is dedicated to wireless communications between vehicles and vehicle-to-road-infrastructure. The DSRC-ITS allocation has been divided into seven 10 MHz channels with 5 MHz reserved as guard bands. The channels are configured into one control channel (CCH) and six service channels (SCH). The CCH is reserved for high priority safety applications and system management data, while the SCH channels are used mainly for safety and non-safety related applications.

The 5.9 GHz DSRC-ITS allocation in the US gave impetus to communication technology standardisation activities which were initiated by the ASTM (American Society for Testing and Materials) who modified the IEEE 802.11a standard [20] to better match the vehicular environment. The ASTM issued ASTM 2213-02 as the basis for 5.9 GHz American DSRC-ITS access technology in 2003. However, in 2004 the standardisation process was transferred to the IEEE 802.11 working group, where task group 'p' started developing an amendment to the IEEE 802.11 standard [21] for vehicular environments [22], known as IEEE 802.11p [14]. IEEE 802.11p addresses lower layer standardisation, whereas development of higher layer specifications was undertaken by the IEEE 1609 working group (WG). The IEEE 1609 WG have defined a set of standards on architecture, interfaces and messages to support DSRC-ITS communication. Collectively, IEEE 802.11p, the IEEE 1609 family of standards and SAE 27385 are called wireless access in vehicular environments (WAVE) and form the WAVE communications protocol stack shown in Figure 2.1.

In Europe under European Commission (EC) Mandate 453 [23, 24], the European Telecommunications Standards Institute (ETSI) in cooperation with the European Committee for Standardisation (CEN) are tasked with defining standards for cooperative ITS with cooperation from the CoMeSafety initiative [3] and results from various European research projects in conjunction with similar standardisation efforts within ISO TC204 WG16[25]

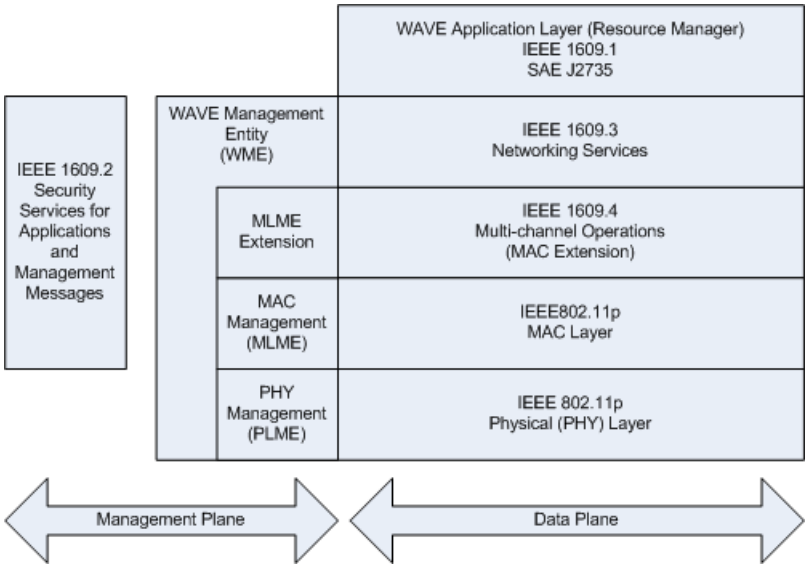


Figure 2.1: WAVE protocol stack (adapted from [22])

and IEEE WAVE WG [26]. The ITS spectrum allocation in Europe is described in ETSI standard ES 202 633 [27] with two specific bands defined for ITS: ITS-G5A from 5.875-5.905 GHz, which is dedicated to safety related and traffic efficiency applications; ITS-G5B, from 5.855 - 5.875 MHz dedicated to non-safety ITS applications. Each band is divided into 10 MHz channels; ITS-G5A is a protected band which is divided into one control channel (G5CC) and two service channels (G5SC1-G5SC2) and ITS-G5B is divided into two service channels G5SC3-G5SC4. The access technology for operation in the ITS-G5 band is a modified version of IEEE 802.11p. The MAC and PHY layers of an ITS-G5 station are defined in [27].

2.3.2 IEEE 802.11p

IEEE 802.11p is a variant of IEEE 802.11a and has been adapted to support the low latency communications requirements of safety applications operating in dynamically changing vehicular environments. IEEE 802.11p specifies both MAC and physical layer (PHY) of the WAVE protocol stack.

At the MAC level, IEEE 802.11p reduces the initial connection set-up overhead required in traditional 802.11 networks for the fast exchange of messages to meet strict application latency requirements, by omitting channel scanning and authentication procedures. The basic MAC mechanism is the same as IEEE 802.11 Distributed Coordination Function (DCF), which is based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme for distributed decentralised communication between wireless nodes. The DCF is central to the exchange of data between vehicles and is described in more

detail in §2.3.3. In addition, the MAC layer is extended and standardised in IEEE 1609.4 to include Enhanced Distributed Channel Access (EDCA), adapted from IEEE 802.1e, in order to support message prioritisation.

In the case of the physical layer IEEE 802.11p is a variant of IEEE 802.11a PHY, which is based on OFDM technology with increased power levels to provide communication over a distance of 100 - 1000 m with data rates of 3 - 27 Mbps. Four classes of Effective Isotropic Radiated power (EIRP) have been defined. For approaching emergency vehicles an EIRP up to 44.8 dBm has been specified and 33 dBm is employed for safety related applications. In comparison to IEEE 802.11a which specifies 20 MHz bandwidth, 10 MHz channels has been specified for WAVE in order to reduce the effect from Doppler spread caused by high mobility of vehicular traffic. As a result of halving the channel bandwidth in IEEE 802.11p, all parameters in the time domain are doubled in comparison to IEEE 802.11a. The increased guard intervals reduce intersymbol interference caused by multipath propagation resulting from mobility and the roadway environment.

2.3.3 IEEE 802.11 Distributed Coordination Function

The distributed coordination function (DCF) is the fundamental medium access control (MAC) protocol of the IEEE 802.11 standard [21]. The DCF allows multiple nodes to share the wireless medium in a distributed manner, without any centralised coordination. In order to mitigate for frame collisions between concurrently transmitting nodes within communication range of each other, DCF is a random access scheme based on the CSMA/CA protocol. A centralised MAC scheme for Access Points, known as the point coordination function (PCF), is also defined by the 802.11 standard, however, this scheme is not considered in this thesis since this thesis focuses on wireless distributed communication. Further information on PCF can be found in [21].

In CSMA/CA prior to transmitting a frame, the node will assess the status of the channel. If the channel is sensed idle for a period of time equal to a distributed interframe space (DIFS), the station transmits. However, if the channel is sensed to be busy or becomes busy during DIFS, the transmission is deferred using a backoff mechanism and the station persists to monitor the channel until it has been idle for a further DIFS. The backoff mechanism is the collision avoidance feature of the protocol and aims to minimise the probability of collision with other stations. A station will defer its transmission and wait for a random delay, called *random backoff interval*, before sensing the channel again. Additionally, a station must wait for a random backoff interval between two consecutive packet transmissions even if the medium is sensed to be idle in the DIFS time. This sequence of events aims to prevent a station from dominating the channel. The process

of determining whether the channel is sensed to be busy or idle is termed *physical carrier sensing*.

The time following a DIFS is slotted and a station is only allowed to transmit at the start of each slot. The slot time is the time needed at any station to detect the transmission of a packet from any other station. The slot time is dependent on the physical layer and accounts for the propagation delay, time to switch between receive and transmit states and the time to signal to the MAC layer the state of the channel.

The backoff mechanism in the DCF employs an exponential backoff scheme which sets the backoff timer to a random integer value uniformly distributed in the range $[0, CW]$, where CW is the contention window, the size of which is determined by the number of successive failed transmission attempts. CW is initially assigned the value of the minimum contention window, CW_{min}. After each unsuccessful transmission the size of CW is doubled up to a maximum value given by CW_{max}.

$CW_{max} = 2^m CW_{min}$, where m = max. number of retransmission attempts.

The backoff counter is decremented for each time interval slot that the channel remains idle until the backoff counter reaches zero. The station then transmits when the backoff time reaches zero. However, if the channel is sensed busy before the backoff timer reaches zero then the backoff count is suspended until the channel is sensed idle again for more than a DIFS. At this point the backoff mechanism resumes decrementing the backoff timer.

Figure 2.2 illustrates the operation of channel contention for unicast communication between two stations sharing the wireless medium. Station B having successfully transmitted a packet waits for a DIFS and then generates a random backoff time of 9 time slots, before transmitting the next packet. During station B's backoff time, station A has transmitted a packet which occurs in the middle of the slot time corresponding to a backoff value of 5. Station B suspends decrementing the backoff count at 5 and only resumes once the channel is sensed idle for DIFS. Following the successful receipt of a packet, the destination station will transmit an acknowledgment frame (ACK). The ACK is transmitted after a period of time called a short interframe space (SIFS) which follows immediately after the reception of the data packet. The SIFS interval is shorter than the DIFS which means that ACK frames are treated with greater priority since no other station is able to detect the channel idle for a DIFS until the end of the ACK. If the transmitting station does not receive an ACK before the ACK_Timeout (as defined by [21]) it reschedules the packet transmission according to the above backoff rules.

The packet transmission described in Figure 2.2 is the basic access mechanism of DCF which uses a 2-way handshake technique. DCF also defines an additional optional 4-way handshake technique for the exchange of unicast packet transmission. This mechanism is

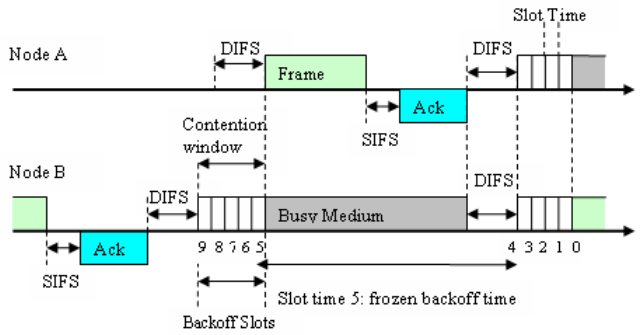


Figure 2.2: IEEE 802.11 physical carrier sensing (basic mechanism)

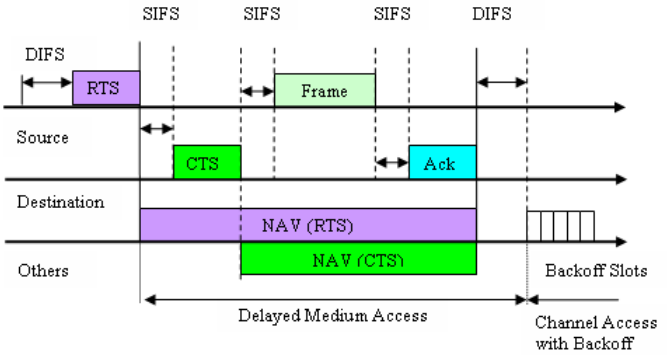


Figure 2.3: IEEE 802.11 IEEE 802.11 RTS/CTS access mechanism

called *virtual carrier sensing* and the procedure for the exchange of data frames is shown in Figure 2.3.

Prior to transmitting a data frame a station waits until the channel is sensed idle for DIFS and then enters the backoff stage (as described for the basic mechanism); instead of transmitting the packet it transmits a request to send frame (RTS) to the destination station. The destination then responds with a clear to send (CTS) frame after a SIFS period. Upon successful reception of the RTS frame, the transmitting station will transmit the data packet after a SIFS period. The RTS and CTS frames carry information on the length of the packet to be transmitted and can be received by any station within communication range. The stations update their network allocation vector (NAV) to the time left until the channel will become free and will refrain from accessing the channel during this time.

The RTS/CTS handshaking mechanism was introduced to avoid packet collisions resulting from the effect of the well known *hidden node* problem, which can't be avoided by the CSMA/CA mechanism alone. The hidden node problem arises when two (or more) nodes not in communication range (out of signal range), simultaneously attempt to send packets to the same receiving node, resulting in a packet collision at the receiving node.

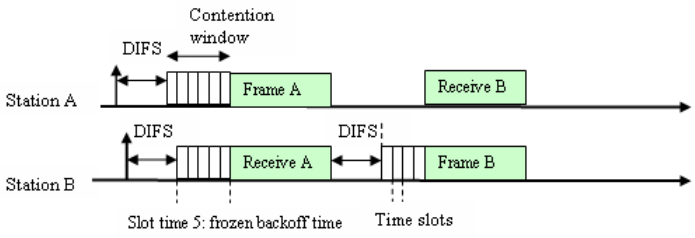


Figure 2.4: IEEE 802.11 broadcast mechanism

Broadcasting using 802.11

For the broadcasting or multicasting of messages using 802.11 there is no reliable message delivery service. There is no reservation of the medium before sending the data frame as shown for the virtual carrier sensing mechanism, or acknowledgement of successful reception at a receiving node. The transmitting node has no means of knowing whether receiving nodes received the packet successfully or otherwise, since there is no mechanism that can detect an erroneous transmission. Hence, there is no mechanism for retransmission at the MAC layer for broadcast packets. This means that the value of the CW remains at CW_{min} each time the backoff procedure is entered. Moreover, since the RTS/CTS of the virtual sensing mechanism is not used, broadcast frame exchange is exposed to the hidden terminal problem. The exchange of broadcast frames between competing stations is shown in Figure 2.4. Stations A and B both have a data packet to send (as shown by the vertical arrows in Figure 2.4) and enter the DIFS interval. At the end of the DIFS interval both stations detect that the channel is clear and proceed to generate a random backoff time. Stations A and B generate a CW equal to 6 and 9 time slots, respectively. Station A has a shorter backoff time than Station B and therefore gains access to the medium before Station B. Station A broadcasts its data packet, which is received by station B. Station B suspends its backoff count at time slot 5, and after successful reception of the packet followed by a DIFS interval it resumes its backoff timer count at time slot 6 before broadcasting its data frame. Frame exchange is similar to Figure 2.2 with the exception that there is no acknowledgment following successful reception of a data packet and therefore the SIFS time interval is not used.

2.4 Application Specific Routing Framework

This section presents a proposal for a data dissemination framework for cooperative vehicle-to-vehicle communication which considers the message delivery requirements for the applications of Table 2.1. Firstly, in order to arrive at this framework a brief review of classes of MANET routing schemes is presented and secondly their suitability is considered

and discussed given the challenges of operation in the vehicular environment, in addition to the specific application data delivery requirements. Finally, the framework is presented which leads to the specific focus of this thesis, as discussed in §2.5.

2.4.1 MANET to VANET

Review of MANET Routing Schemes

Many routing protocols have been proposed for *ad hoc* networks with the goal of providing efficient routing schemes. These schemes are generally classified into two broad categories, topological and position-based routing [28]:

Topology-based Routing

Topology-based routing protocols use information about the links that exist in the network to perform packet forwarding and can be further broken down into three categories, proactive, reactive and hybrid routing.

Proactive routing protocols (e.g. DSDV [29]) attempt to create a global view of network connectivity at each node through maintaining a routing table that stores routing information to all destinations, computed *a priori*. Routing information is exchanged periodically, or when changes are detected in the network topology. Routing information is thus maintained even for routes that are not in use. Proactive protocols are 'high-maintenance' in that they do not scale well with network size or rate of changes in topology [30, 31].

Reactive (also known as on-demand) routing protocols (e.g. DSR [32]), in contrast, create routes only when required by the source node and are based on a route request, reply and maintenance approach. Route discovery is by necessity based on flooding (it is assumed that the identity of the nodes is known *a priori*). On-demand routing generally scales better than proactive routing to large numbers of nodes since it does not maintain a permanent entry to each destination node in the network and a route is computed only when needed [30, 31]. However, a drawback of on-demand protocols is the latency involved in locating the destination node.

Hybrid routing protocols, are the third category of topological protocols which combine both proactive and reactive routing, e.g. the zone routing protocol (ZRP) [33, 34]. The ZRP maintains zones; within a zone proactive routing is used, whereas a reactive paradigm is used for the location of destination nodes outside a zone. The advantage of zone routing is its scalability since the 'global' routing table overhead is limited by zone size and route request overheads are reduced for nodes outside the local zone. Further detailed reviews of topological routing techniques can be found in [30] and references therein.

Position-based Routing

Unlike topology-based routing, position-based routing forwards packets based on position information reducing, and in some cases eliminating, the overhead generated by frequent topology updates [35]. Position-based routing requires the use of a system that provides positioning information and a location service (LS) [36]. Although mobile nodes can disseminate their positioning information via flooding algorithms, a location service is important for scalability [8]. A location service helps a source node to detect the location of the destination node. A review of location services can be found in [28, 36].

There are two types of packet forwarding paradigms commonly used within position-based routing; restricted flooding and geographic forwarding (also referred to as *greedy forwarding*)[37].

Restricted flooding techniques flood a packet through a region that has been set up using the position of the source and destination nodes. Although restricted flooding is still affected by topology changes the amount of control traffic is reduced by the use of position information, thus limiting the scope of route searches and reducing network congestion. When a route to the destination cannot be found, network-wide flooding of the route request message occurs resulting in high bandwidth utilisation and unnecessary network congestion.

Geographic routing, on the other hand, relies on the local state of the forwarding node to determine which neighbouring node is closest to the destination to forward the packet to and is thus not affected by the underlying topology of the network. The selection of the neighbouring node depends on the optimisation criteria of the algorithm. Even though geographic forwarding helps to reduce routing overhead as a result of topology updates, the lack of global topology information prevents it from predicting topology holes or network partitions [28].

IVC operating Constraints

Environment

The majority of *ad hoc* networking research, in the development and comparison of routing protocols, has evaluated performance based on a 2-dimensional rectangular plane where nodes change their speed and direction randomly. This differs from the mobility model required for an *ad hoc* IVC network in several ways. Firstly, the movements of vehicles are spatially restricted to the road structure, thus constraining the mobility pattern significantly. Secondly, the speed of vehicles is often much faster than the node speeds used in the literature. Thirdly and most importantly, the dynamic nature of vehicular traffic flow

(i.e. traffic flow patterns and density), must be used in order to evaluate the performance of the routing protocol for the target applications. The effect of differing mobility models on the relative performance of routing protocols has been highlighted in [38, 39]. This emphasises the fact that the performance of a routing protocol modelled without emulating the movement characteristics and spatial constraints of the target application cannot be assumed to exhibit the same quantitative results demonstrated in the literature in a different operational environment.

Another difference in mapping routing techniques to IVC is that no prior knowledge of the possible set of identifiers exists without maintaining either a centralised or a distributed database. As pointed out in [40], the possible number of node identifiers can easily exceed a practical size and will be constantly changing, thus making it unmanageable to maintain such a database. Hence, node ID must be considered to be *a prori* unknown. Since vehicles are increasingly being equipped with positioning systems (e.g. GPS) it can be assumed that future vehicles will be equipped with an accurate positioning system as standard, allowing vehicles to be addressed by position. Vehicle ID must therefore consist of two fields, a geographical location field and a unique node identification number, as a minimum addressing requirement. In applications requiring data to be addressed to a specific destination, vehicle IDs can be discovered through their current position and maintained by each neighbouring node only for as long as necessary. In this way, both conventional distributed and centralised node ID database solutions are avoided completely.

Application Routing Protocol Considerations

The potential size of an IVC *ad hoc* network, coupled with the dynamic nature of traffic flow, excludes the use of a purely proactive protocol for the following reasons. Firstly, continuous changes in vehicle connectivity will result in constant routing update packets being transmitted, compromising routing convergence (by the time a vehicle receives routing update information, it may already be 'stale'). Secondly, as a consequence of control traffic consuming network resources, the delivery of application data will be restricted. Thirdly, as the number of vehicles increases, the size of the routing update packet will increase proportionally, placing extra demands on network resources. However, proactive protocols may be suitable at a local level for a restricted number of vehicles, where timeliness of delivery is imperative to the application and the *relative* velocity between vehicles is low.

A purely reactive protocol assumes that the identity of a node is *a priori* known, in order for it to address a message to a particular destination. However, the creation and maintenance of a vehicle ID database is likely to be prohibitively complex. Even if the vehicle had such information, it would initially have no knowledge of a path to the destination.

Therefore, finding a path would delay transmission and, for applications where timeliness of delivery is imperative, this would not be acceptable. Although route caching can be supported, there will still be an initialisation period before information is built up, but the freshness of this information may be short-lived due to continuous changes in network connectivity. Keeping such short-lived information current, incurs significant overhead and this further limits the protocol's scalability. Thus, supporting a purely reactive routing protocol is unsuitable for IVC.

For low priority applications where delay is acceptable a modified version of the reactive routing scheme, taking into account addressing issues, may offer a suitable solution. However, for fast traffic flows, message delivery may not be possible if links are continuously changing. A hybrid scheme using pure versions of both the reactive and proactive routing paradigms would not be suitable without modifying their methodologies to take into account position information, although it would offer some scalability advantages.

One could automatically assume that since position-based routing delivers messages based on position and fulfils one of the addressing requirements of IVC, it would provide the best routing solution. However, in the case of restricted flooding, knowledge of a vehicle's position and ID are assumed to be *a priori* known so that the message can be flooded to the area where the vehicle is expected to be located. If the vehicle cannot be found then this may result in network-wide flooding in order to locate the required destination. This is clearly not acceptable to applications for which timeliness of delivery is imperative. The level of detail of geographic information required to support efficient restricted flooding must include not only relative position of neighbouring vehicles, but also their direction of motion relative to the vehicular traffic flow and the message destination region. As will be seen in §2.4.2, the justification for maintaining this level of geographical information complexity in the routing layer is dictated by the ITS applications themselves. For low priority applications, where a vehicle has prior knowledge of the destination, this technique may be suitable, although network-wide flooding for unicast transmissions must be avoided since the potential control overhead in locating a route could tie up network resources unnecessarily.

Geographical routing suffers from the requirement for a LS. Although the method used in routing the message to the destination is effectively stateless, the LS will be affected by the underlying connectivity and may delay the delivery whilst waiting for position information. The position information also needs to be accurate up to one-hop away from the destination. The algorithmic complexity and maintenance overheads in implementing a LS can be highly costly [28]. A modified version of the geographical forwarding scheme may be appropriate for certain groups of applications but the implementation of a LS is likely to be prohibitively complex for the range of application scenarios considered in this

thesis. The geocast scheme is a technique that can be applied to IVC for applications where information is of relevance to vehicles in a particular region, such as incident warning.

2.4.2 Data Dissemination Framework for Cooperative Vehicular Applications

Having investigated the requirements of various ITS application scenarios in Table 2.1 and their use case scenarios in Appendix A, it became evident that they have quite different QoS demands, message delivery requirements and differing regions to which the data is relevant. The region to which the message is relevant for each application is referred to as the routing zone of relevance (RZR), which is adapted from [41]. For example, ITS application scenarios such as vehicle platooning and cooperative driving will have a very low threshold in terms of acceptable communication delay, since any excess delay could mean the difference between the application either working as implemented, or potentially causing an accident. The RZR for these applications is considered to be in the near-vicinity of the source vehicle. On the other hand, applications such as mobile vending services and traffic information systems are not critically dependent on communication delays and have a wide area RZR. Thus, it is imperative that a data priority is assigned which depends on the safety-related implications of the application. The message delivery requirements for various application scenarios are also different; e.g. platoons may require group delivery, whereas incident warning applications may require a persistent broadcast to vehicles within a specific region and specific vehicle based application requests, such as a reply to a traffic information enquiry, may require unicast delivery.

In order to implement an IVC *ad hoc* network which meets the requirements of the application scenarios, in this thesis it has been identified that there is a need to use different routing scheme paradigms, the selection of which is dependent on the application and its specific priority rating, required RZR and message delivery requirements. The universal deployment of an accurate positioning system is assumed in future vehicles (e.g. GPS or Galileo) along with the assumption that there is neither a centralised, nor a distributed database maintaining a list of vehicle identifiers, as discussed in §2.4.1.

The message delivery requirements of the IVC applications can be classified into three different categories. The first class consists of those applications such as an incident warning or an approaching emergency vehicle warning, which require information to be broadcast to a geographic region. The required message delivery type used in this case is a geocasting delivery scheme. The second class consists of applications such as a response to a traffic information request where an expected RZR can be determined from the packet sent from the requesting vehicle, using a method similar to the "expected-zone" technique used in [42]. The response message will be specifically addressed to the requesting vehicle

using unicast delivery scheme. The third class covers applications such as platooning or cooperative driving where communication between a number of vehicles is required in order to coordinate manoeuvres between vehicles. The required delivery type is multicasting, also known as group delivery.

The second and third classes mentioned above will benefit from local connectivity information. Maintaining network connectivity information at the local level will aid delivery of unicast and multicast data for applications where timeliness of delivery is imperative and as the message approaches its destination. Maintaining network connectivity information requires periodic exchanges of control packets, called beacon messages. In a highly dynamic environment, where links are formed and broken frequently, the amount of control traffic required in order to maintain up-to-date connectivity restricts this type of protocol scheme from scaling well with an increase in network size [43]. However, knowledge of network connectivity is considered to be important for the implementation of all classes of IVC applications mentioned previously, for three reasons: firstly, local connectivity information is important for applications such as platooning, cooperative driving and any application requiring coordination between vehicles where timeliness of delivery is imperative; secondly, local connectivity knowledge will help reduce the number of retransmissions required in order for a packet to find its destination within the RZR; thirdly, knowledge of neighbour connectivity helps reduce unnecessary retransmissions for applications requiring message broadcasts.

The cooperative manoeuvre applications mentioned above require communication between specific vehicles, which are generally in the immediate vicinity of the source vehicle. Since these types of applications operate in the immediate vicinity of the source vehicle, network connectivity information is maintained at the local level in zones called "local zones" (LZ) centred on each node, similar to the zones maintained by the zone routing protocol (ZRP) [33]. The size of the zone changes dynamically, depending on local traffic density, local mobility and local data traffic overhead. Unlike ZRP, a reactive routing protocol is not used to provide routing between the border nodes of each zone as will be explained in the following description of framework properties.

The above discussion leads to the conclusion that in order to satisfy the communication routing requirements of the plethora of application scenarios under consideration, a suite of routing protocols needs to be deployed. This means adopting a hybrid routing framework approach that combines a routing scheme which maintains connectivity data at the local level with position-based routing and geocasting schemes beyond the LZ. Figure 2.5 shows a schematic diagram of the proposed IVC routing framework.

In the proposed IVC routing framework, independently of the message delivery type, when

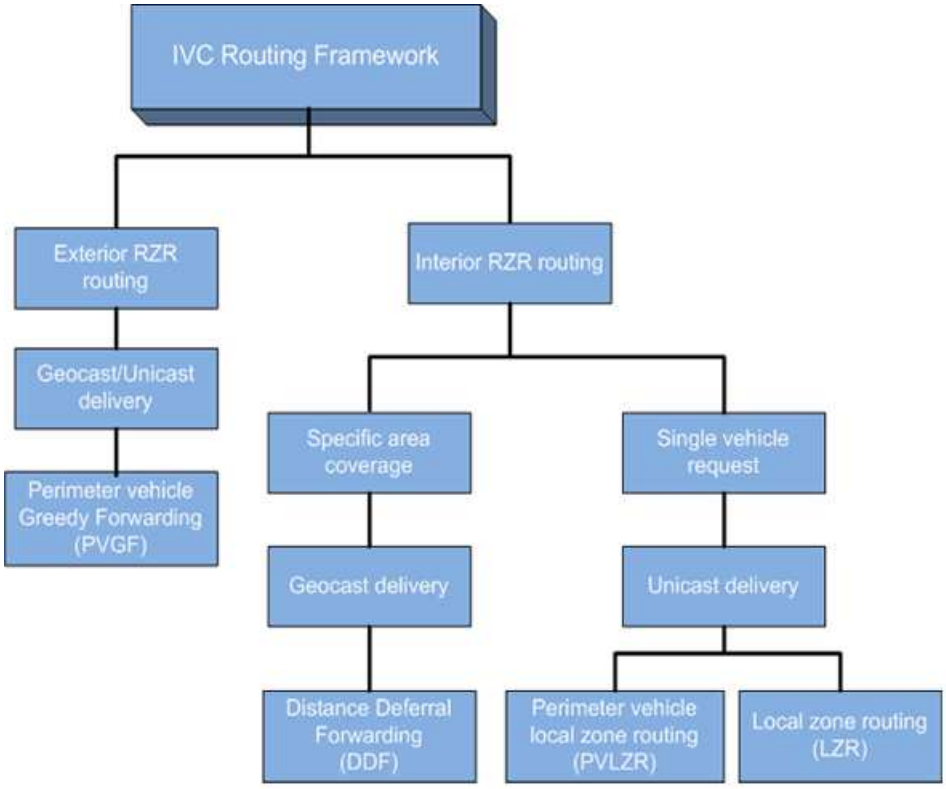


Figure 2.5: Data dissemination framework for cooperative vehicular applications proposed by this thesis

the source vehicle lies outside of the RZR the message is forwarded towards the RZR using a routing technique called perimeter vehicle greedy forwarding (PVGF). PVGF is based on the principles of greedy forwarding [44, 37, 36].

When the message reaches the RZR, the routing technique employed within the RZR changes depending upon the required message delivery scheme. If the message type is geocast, a routing technique called distance deferral forwarding (DDF) is used to deliver the message within the RZR.

However, if the message is addressed to particular vehicle(s) within the RZR the message is forwarded to the destination vehicle(s) using local zone routing (LZR) along with perimeter vehicle local zone routing (PVLZR). Both of these routing schemes utilise local connectivity information in order to locate the destination vehicle within the RZR. In the scenario where the source vehicle is a member of the RZR and addresses a message to a specific vehicle within the RZR, then both LZR and PVLZR are used to route the message to the destination(s). LZR is used to deliver a message if a node determines that the destination node is inside its LZ, otherwise PVLZR is used to forward the message. PVLZR is used to forward the message in the direction of the destination node, whereas LZR is used to deliver the message when a node determines that the destination node is within its LZ.

The novelty of the proposed hybrid routing protocol lies in the manner in which existing concepts from a variety of protocols are adapted and customised in the context of a vehicular environment. The definition and maintenance of the zones is original in this context and dictates the way in which a multitude of routing concepts are implemented and interoperate. The proposed framework consists of a variety of different mechanisms and therefore implementing the framework in its entirety is outside the scope of this PhD. For this reason, the research efforts in this thesis focus on the underlining supporting mechanisms, and one branch of the framework.

Framework Properties

The following section expands on the routing framework presented in Figure 2.5 and discusses its constituent algorithms in outline. The IVC framework is specific to the highway environment and assumes that the "beacon" packet header will also contain road identification information, facilitating vehicle classification per road.

When a data packet, m is transmitted by vehicle A, which can either be the source vehicle or the vehicle retransmitting m, the decision as to what type of routing protocol to apply depends upon the required type of message delivery (which is application dependent), as well as the location of A with respect to the RZR.

For IVC applications such as a traffic interrogation request further along the motorway or a broadcast transmission addressed to all vehicles in a particular region, the PVGF protocol is used to forward m towards the RZR.

Unlike greedy routing techniques [44, 37], where recovery techniques are employed to route around topology holes, such approaches are not required for the application of a vehicular *ad hoc* network, on a highway. If partitions (topology holes) occur in the network or no appropriate vehicle class exists in the neighbour table to forward m in the direction of the RZR, the packet is stored in the neighbour waiting table until a vehicle meeting the required vehicle classification is detected. The vehicle storing m takes advantage of the dynamic nature of vehicular traffic flow by waiting until it encounters an appropriate neighbour so that it can forward m in the direction of the RZR. The method employed in dealing with partitions is similar to the method applied in [45]. Once the message has reached the RZR, or the source vehicle is inside the RZR, the routing scheme changes according to the application delivery requirements depending on whether a unicast, multicast, geocast or anycast delivery is required.

When the message delivery type within the RZR is either unicast or multicast, the search for the destination utilises the local connectivity information maintained in the *LZ*. The

local zone routing is achieved using a proactive routing protocol, employing link state routing [46], in order to provide a view of the topology of the network at each node within the LZ. The local zone information is stored in the local zone table. The LZ is specified as a geographic region. The size of the LZ changes dynamically depending on local vehicular traffic density, local data traffic overhead and local mobility. The minimum size of the LZ, is defined by the maximum transmission range. If geocast delivery is required within the RZR, then the DDF protocol is used which utilises LZ information to make intelligent forwarding decisions.

When the destination node is found within a node's LZ then LZR is used to deliver m. The selection of the next hop node is made depending on the location of the destination. This decision is repeated at each node receiving m within the LZ until m reaches its destination. However, if the destination does not exist in the LZ then PVLZR is used. In PVLZR, m is forwarded to the node furthest away within the LZ, called the perimeter node. LZR is then used to deliver m to the perimeter node. At the perimeter node, if the destination is not within its local zone table the above procedure is repeated. Otherwise, if the destination is found within the LZ, LZR is used.

2.5 Application Focus

The research in this thesis focuses on data dissemination protocols in order to deliver messages for safety related applications, which aim to reduce road traffic accidents within the highway environment. Such protocols are required to provide LZ awareness information through the periodic exchange of beacon messages and to disseminate event-driven hazard warning messages.

Not only does the exchange of periodic beacon messages provide local awareness knowledge, which aids drivers to avoid potential dangerous situations within their LZ such as during lane change or ramp on/off maneuvers, it is an underlying mechanism within the proposed framework which is also utilised to aid intelligent forwarding decisions.

Event-driven messages are used to deliver hazard warning information to other drivers in order to warn them of an approaching hazard. The dissemination of event-driven messages is originated by a node either detecting or involved in an incident which requires broadcasting to other vehicles beyond its LZ within a geographical area. In the context of the proposed routing framework event driven messages are delivered using geocasting.

The aspect of research considered in this thesis is to provide reliable and efficient delivery of event-driven and periodic message exchange able to satisfy application delay constraints through efficient selection of forwarding nodes within a geographical area, considering the

challenges discussed below. Moreover, as both beacon and emergency messages share the control channel, the impact of contention on reliability and timeliness for what is effectively a single channel system plays a central role in the considerations of this thesis.

2.5.1 Challenges of Broadcasting in VANETS

The particular characteristics of vehicular *ad hoc* networks such as the highly dynamic nature of vehicular traffic flow (fast flowing through to low density fragmented road networks), short-lived communication links, limited channel bandwidth, etc., all introduce challenges for the provision of an efficient data dissemination schemes. Moreover, the lack of packet acknowledgment, packet retransmission and a medium reservation scheme that are consequences of using IEEE 802.11 makes it difficult to achieve high broadcast reliability and efficiency, particularly in dense networks as a result of contention and interference.

The simplest way of broadcasting information in wireless multihop networks is by flooding, whereby each node receiving the broadcast for the first time retransmits the message to all its neighbouring nodes. However, the drawback of such a simple approach is that flooding can result in an extremely high number of redundant broadcasts, frequent contention and collisions in transmission between neighbouring nodes. This problem is referred to as the broadcast storm problem [47, 48, 49]. In addition to avoiding the broadcast storm problem, channel contention issues will also delay the dissemination of messages which in the case of vehicular safety messages is an important design challenge.

Much of the research in defining efficient and reliable data dissemination for the geobroadcasting of vehicular safety messages focuses on meeting the challenges mentioned above. In Chapter 3 a number of these proposed schemes are reviewed.

2.6 Summary

This chapter presents a review of various applications which can be implemented using vehicle-to-vehicle wireless communication technology and considers their communication requirements through use case analysis. This analysis is then used to define a data dissemination framework for a sub-set of these applications which improve road safety and efficiency. Standardisation activities are considered in both the US and EU and focus on the US-led activities on WAVE technology based on IEEE 802.11p. The IEEE 802.11 access technology has an impact on research detailed later in this thesis and in order to provide a better understanding of its operation and implications its basic operation is presented.

The second part of this chapter defines a data dissemination framework for road safety and traffic efficiency applications, having firstly considered their specific communication requirements and constraints within the vehicular environment, and secondly the appropriateness of message routing schemes proposed in the MANET literature. This chapter concludes by presenting the research focus of this thesis, which is the geocasting of messages for safety related applications and also considers the challenges introduced by the the vehicular environment and the unreliable channel access mechanism. Furthermore, this chapter highlights the challenge and unreliability of the IEEE 802.11 access mechanism employed by IEEE 802.11p in broadcast mode, where the hidden node and broadcast storm problems need to be overcome in order to provide both efficient and reliable data dissemination which meets the timely communication constraints of safety related applications. Chapter 3 presents a review of published message dissemination schemes, in the open literature, that aim to overcome the above mentioned challenges in order to meet the constraints imposed by the IVC application requirements.

CHAPTER 3

REVIEW OF DATA DISSEMINATION SCHEMES

3.1 Introduction

Vehicle safety related applications are one of the most promising application areas for communications between vehicles. In these types of applications information is required to be disseminated to all surrounding vehicles, which brings about the requirement of a broadcast forwarding protocol. The simplest way of broadcasting data is by flooding, whereby each vehicle receiving the broadcast message for the first time retransmits it to all its neighbouring nodes. However, the drawback of such simple techniques is that they suffer from the well known *broadcast storm* problem, where a large amount of bandwidth is consumed by an excess number of retransmissions leading to collisions and channel congestion, particularly when node density is high. The consequences of this are to increase delivery latency and reduce the packet delivery ratio which is a serious consideration for the delivery of safety related data.

Not surprisingly, much of the research in the area of broadcast forwarding algorithms has focused on alleviating the broadcast storm problem through controlling the number of nodes that are allowed to forward the message. Many of these schemes focus on the criteria used in the selection of the forwarding node and can be generally characterised as being, distance, probabilistic and cluster based. This chapter provides a review of forwarding schemes relevant to vehicular networks and focus on their ability to provide reliable and timely delivery. This chapter also includes a short review of techniques which have been used by researchers to adapt transmit power in order to reduce channel congestion, and methods that aim to control channel congestion by adapting to local traffic conditions. The dynamic nature of vehicle traffic will require that safety messages to be disseminated to vehicles in low density situations where partitions occur between vehicles in the addressed area. Therefore, a review of mechanisms reported in the literature to overcome partitions during the dissemination of safety messages, is included. Although data security is not

a focus of this study, a brief review of the challenges it imposes on broadcasting safety related data is provided. This chapter concludes with a comparative discussion of the reviewed schemes highlighting the gaps in the state of the art in the public domain.

3.1.1 Distance Based Approaches

In comparison to other forwarding node techniques used in the literature, distance based selection has proven to be the most popular. Distance based techniques select the next forwarding node(s) to be the farthest away from the previous transmitting node in order to restrict the number of unnecessary retransmissions. Distance classification can be further broken down into deterministic and probabilistic based approaches.

Deterministic Based Approaches

In [50] Sun *et al.* present two protocols, TRAcking DEtection (TRADE) and Distance Deferral Transmission (DDT) which aim to reduce the bandwidth utilisation in comparison to traditional broadcasting protocols by limiting the number of rebroadcasting vehicles. The TRADE protocol classifies neighbouring vehicles into three different groups: same road ahead; same road behind; and different road, according to their relative position on the road network. TRADE uses position vectors to classify neighbouring vehicles into the above mentioned groups. Position information is obtained through the periodic exchange of beacon messages. The TRADE protocol forwards safety messages by selecting the border nodes, that is the nodes farthest away within the different neighbour groups, which are relevant to the required direction of message propagation. The safety message explicitly contains the IDs of the border vehicles selected to rebroadcast the safety message. In comparison, the DDT aims to further reduce the bandwidth utilisation by omitting the exchange of beacon messages and therefore the categorization of neighbouring vehicles. In DDT a vehicle receiving a message sets a defer timer which is inversely proportional to the distance from the source of the transmission. During the defer time a vehicle caches duplicate copies of the same message it overhears. When the defer time expires, if the vehicle determines that most of its transmission area has been covered by its neighbours it will drop the message, otherwise it will rebroadcast it. The performance of DDT and TRADE is compared against a traditional broadcast protocol for high and low vehicle densities and is assessed using bandwidth utilisation and reachability metrics. Not surprisingly, both DDT and TRADE outperformed broadcasting in terms of bandwidth utilisation. However, in terms of reachability DDT was found to perform better since retransmission decisions are based on local coverage. Sun *et al.* do not state the coverage limit, and do not consider congestion or collisions at the MAC layer. Moreover, they do not state how

they determine waiting defer time.

In [40, 45, 51], the creation of an implicitly defined multicast group, called role-based multicast (RBM), for IVC to provide the dissemination of a road accident message for an accident having occurred on both a divided and undivided highway for varying levels of equipped vehicles, is investigated. A technique similar to location-based multicast (LBM) [52] is used, except that all nodes can participate in forwarding the message, as long as messages do not exceed a finite number of hops. The number of message retransmissions is restricted through only allowing a vehicle to rebroadcast a message once a wait timer has expired (based on distance deferral transmission (DDT)) and it has neighbours in the forwarding direction. If this condition is not met then the message will only be transmitted once a new neighbouring vehicle has been detected. This technique aids message dissemination when network partitions occur. The multicast group is implicitly defined within the multicast region by those vehicles whose braking distance allows them to stop before the accident. The success is measured by determining the ratio of informed vehicles to the size of the multicast group. The routing mechanisms mentioned previously along with a localised group membership service (LGMS) were integrated in [53] to form an *ad hoc* IVC network for the detection of traffic jams on highways.

In [54] the authors present the Inter-Vehicle Geocast (IVG) protocol for the broadcasting of accident alarm messages within 'risk areas' which are defined by driving direction and positioning of vehicles relative to the originating vehicle. IVG protocol aims to reduce network congestion by reducing the number of retransmitting nodes in addition to overcoming network fragmentation. The authors of IVG use a method similar to DDT which restricts the number of retransmitting nodes, known as 'relay nodes' to the vehicles positioned farthest away from the sender node. Each time a vehicle receives an alarm message a vehicle starts a defer timer, which is inversely proportional to the distance from the sending node. If during the defer time the vehicle has not overheard the same message being rebroadcast by another vehicle within its transmission range, the vehicle will designate itself as a relay node and rebroadcast the alarm message. Fragmentation is overcome in IVG by allowing a relay node to rebroadcast the alarm message periodically. The frequency with which the message is broadcast is related to the braking distance of a vehicle to ensure that a vehicle approaching the risk area is warned in a timely manner.

The ODAM protocol presented in [55] proposes optimisations to IVG. ODAM's functionality is similar to IVG, however, it aims to improve performance by, restricting the occurrence of multiple relay nodes and adapting retransmission deferral timing to received packet delay. The authors of ODAM include a mean packet delivery delay in the calculation of the deferral time which attempts to make it adaptive to channel usage. Multiple relay nodes occur when equidistant nodes are assigned similar defer times which expire at

the same time, leading to the nodes designating themselves as relays and as a consequence attempt to access the communication channel at the same time. The authors aim to resolve this situation by only allowing the node with the lowest identification number to operate as the relay node. The authors investigate the number of informed vehicles against varying transmission range and compare the performance of ODAM with DDT and RBM. It was also noted that RBM fails to overcome fragmentation in the case of light traffic and DDT does not include a mechanism which overcomes fragmentation. ODAM was found to be more reliable in all scenarios tested. Since ODAM does not support beacon exchange messages, the calculation of the mean packet delay used in the retransmission deferral time will not give an accurate estimation of current activity since event messages are received less frequently and therefore the time window over which the mean delay is determined is quite large.

In a similar manner to ODAM the Distributed Robust Geocast (DRG) [56] protocol uses a distance contention based algorithm for the selection of the next relay node and overcomes network fragmentation through periodic retransmissions, the timing of which is related to maximum velocity of vehicles and the transmission range. However, unlike ODAM, DRG firstly considers the lack of implicit acknowledgment from a next forwarding node to have occurred as a result of channel losses instead of immediately assuming a partition has occurred. A relaying node after rebroadcasting a message will schedule a retransmission time according to the round trip time for the packet to reach the farthest node in the coverage area. If after a certain number of retransmissions a relay node does not receive an implicit acknowledgement, which satisfies the forwarding criteria, a network partition is assumed, and the retransmission backoff time is extended.

Tonguz *et al.* present a dissemination protocol called DV-CAST for the broadcasting of vehicle safety messages in [57] and results in [58]. DV-CAST uses local topology information in order for vehicles to determine the relevance of the received message in addition to determining whether it has neighbours within the required forwarding direction. A vehicle classifies neighbours within its local topology into two relative position groups depending on the direction of the traffic flow. The authors consider dense, regular and sparse traffic densities in the design of the protocol. In the case of dense traffic, the authors propose to use a probabilistic distance based suppression technique previously presented in [49] for message dissemination. If a vehicle is at the end of the cluster and has at least one neighbour in the opposite traffic direction, the authors classify the network as being sparsely connected and suggest the use of a forwarding scheme similar to that of [51]. However, if the vehicle at the end of the cluster does not have any neighbouring nodes in the opposite traffic direction then a store-and-forward technique is suggested. Although the proposed design for DV-CAST considers different densities of vehicular traffic and is able to over-

come network partitions, the waiting time used in the suppression algorithm does not consider local neighbour density changes adaptively and is still vulnerable to the spatial broadcast storm problem.

As can be seen from the simple distance based schemes, [50, 51, 55], the next relay node with the shortest waiting time is located at the border of the transmission range. However, in the case where local vehicle density is low, such that there are no nodes located close to the border of the transmission range, then the next relay node will undergo a longer waiting time. As a consequence the overall end-to-end delay will increase. The Time Reservation-based relay Node Selecting (TRRS) algorithm presented in [59] aims to address the previously mentioned issue through decreasing the overall end-to-end delay of emergency warning messages within a region, regardless of vehicle density. In TRRS all nodes receiving the message from a relay node randomly select a waiting time within a given time-window. Each node has a time window with a different lower and upper limit, the size of which is inversely proportional to the distance away from the previous relaying node. The authors also present Enhanced TRRS (ETRRS) which prevents a node which has received multiple duplicate packets from being selected as the next relay node. The authors compare the performance of TRRS and ETRRS with a simple distance based scheme and show that both their proposed schemes have a lower end-to-end delay for varying vehicle densities. Even though the simple distance based scheme generates the lowest network overheads, the authors conclude that overall their schemes are more efficient. However, the authors do not consider fragmentation within their simulations and the vehicle simulation does not consider overtaking or lane changing.

The UMB protocol [60] specifically addresses the hidden node, broadcast storm and reliability problems for vehicular multi-hop networks in the urban environment. The UMB protocol consists of two forwarding mechanisms; directional broadcast and intersection broadcast. The contention scheme which selects the next relaying node used in the directional broadcast mechanism is the core element of UMB, and is considered in this review only. Further details of the intersection broadcast mechanism can be found in [60, 61]. The directional broadcast mechanism uses a request-to-broadcast (RTB) and clear-to-broadcast (CTB) mechanism at the MAC layer in order to increase delivery reliability. The area covered by a transmitting node is divided into segments and the next relay node is selected from the farthest non-empty sector without the use of local topology information. If there is more than one node in the furthest segment, this segment is divided iteratively into subsegments. The source node transmits a RTB packet which contains the source position, direction of broadcast and sector size. On receiving the RTB packet nodes determine their distance to the source node. The nodes transmit a jamming signal called a black-burst. The length of the signal is proportional to a node's distance from

the source. At the end of the black-burst nodes listen to the channel. If the channel is sensed to be busy then they do not participate any further in the process since there are nodes further away in the broadcast direction. However, if a node senses that the channel is idle, their black burst was the longest and they reply to the source with a CTB packet. On reception of the CTB the source node forwards the message to the node sending CTB packet which then becomes the next relaying node. If there is more than one node in the farthest non-empty segment they will all transmit a CTB simultaneously, the source node will detect a collision and will retransmit a RTB packet. Only those nodes which sent a CTB packet participate in the contention process. In order to pick only one node the furthest non-empty segment is divided into subsegments. This process continues iteratively until the source node successfully receives a CTB packet. The performance of UMB is compared against two MAC layer flooding based protocols. UMB shows better performance in terms of reliable delivery and channel efficiency. Although the RTB/CTB scheme provides a high delivery reliability and reduction in channel utilisation, the channel contention mechanism leads to the potential relaying nodes waiting the longest before being able to retransmit. Moreover, in high traffic densities the iterative process used to select the relay node will lead to long retransmission latency.

Fasolo *et al.*, in [62] aim to minimise the potential latency which can be caused by the contention resolution mechanism used in UMB [60]. The authors present the Smart Broadcast (SB) protocol which is similar to UMB. SB also partitions the coverage range into sectors and uses a RTB/CTB contention-resolution mechanism for the selection of the next relay node. However, the SB protocol does not necessarily select the relay node from the farthest non-empty sector, since it does not spend time to resolve collisions. Unlike UMB, the SB protocol does not use a black burst mechanism, instead it selects a random backoff time from a contention window associated to the sector they belong to. The closer the sector is to the edge of the coverage region, the smaller the contention window size. Since the SB protocol does not employ a collision resolution scheme, the authors propose to increase forwarding reliability by allowing the source node to repeat the contention process if it has not received a CTB packet within a maximum contention window, with an added delay each time the process is repeated. When the source node receives a CTB packet it sends the message to the relay node which then broadcasts a RTB packet. After broadcasting the message to the relay node, the source node expects to hear a RTB packet from the relay node, which acknowledges the success of the forwarding process. The performance of RB is compared with three other protocols, however the comparative delay gains for the SB protocol are found to be marginal.

The Fast Broadcast (FB) protocol [63] aims to reduce the number of retransmissions using a distance based scheme by computing a back-off time based on a dynamically

estimated transmission range. The FB protocol consists of two mechanisms, the estimation phase and the broadcast phase. The estimation phase is responsible for determining the transmission range in the front and backwards directions (the authors assume a linear highway) dynamically, through the exchange of beacon messages. The broadcast phase is responsible for forwarding the warning message within a region. The sender of the message includes its current maximum transmission range value in the packet. A node receiving the message calculates a distance related waiting time based on a contention window value, which uses the estimated transmission range declared in the packet from the sender node. If when the waiting time expires the node has not overheard the message being forwarded the node updates the packet with its current maximum transmission range prior to rebroadcasting. The performance of FB is compared against a similar protocol which employs a static transmission range. Preliminary results show that determining the transmission range dynamically can reduce the number of forwarding retransmissions required to cover the addressed region.

In comparison to schemes which select the next relaying node through a contention process at the receiving node, the REACT scheme presented in [64] explicitly selects the next relay node at the transmitter through local topology information. The REACT protocol consists of two mechanisms, the forwarding decision algorithm (FDA) and the Topology Discovery Algorithm (TDA). The FDA is responsible for selection of the next forwarding node which must lie on the trajectory specified by the originator node. In order to reduce the likelihood that the next relay node is actually unreachable, the furthest node within a range which is less than the maximum transmission range is selected from the node's neighbour list. The ID of the next relay node is included in the broadcast packet. The forwarding node stores a copy of the forwarded message and sets a waiting timer (the value of the waiting timer is not discussed in [64]). If the node has not received an implicit acknowledgment from the selected forwarding node (i.e. overhead the broadcast from the selected forwarding node) then it will rerun the FDA and choose another forwarding node. This process is repeated until a valid forwarding node is found. This mechanism enables REACT to overcome network fragmentation and overcome network collision which can prevent the forwarding process from continuing successfully.

In [10, 65] the authors propose a strategy for the dissemination of event driven emergency messages within a geographical area called Emergency Message Dissemination for vehicular environments (EMDV). The authors build on previous investigations of Contention Based Forwarding (CBF) presented in [66] which they augment using concepts from position based routing to form the EMDV message dissemination strategy. In EMDV a node transmitting an emergency message selects the next relay node from its neighbour table which will provide the highest progress within its forwarding area, and includes the

address in the *NextHop* field of the message. As in the case of the REACT algorithm, the forwarding area is defined to be less than the maximum transmission range. A node receiving the message positioned in the forwarding area of the sending node firstly determines whether to perform immediate, or contention forwarding. If the address of the receiving node matches the *NextHop* field it will forward the message immediately without contention. Before retransmitting the message the node will determine from its neighbour list the next relay node farthest away within its forwarding area and update the *NextHop* field. However, if there are no neighbours located in its forwarding area or the border of the dissemination region is within its forwarding area, the next hop will be set to broadcast. The relay node will broadcast the message immediately without delay, increment a message count and set a contention time to verify that at least one node within its forwarding area retransmits the message. The contention time set by a relaying node is given by the sum of a maximum contention window and the value of maximum channel access time. Otherwise, if the node is not the next relay it will wait to rebroadcast the message after setting a contention time. The contention time is based on the distance between the sender and receiver of a maximum contention time, so that nodes closer to the boundary of the forwarding area have shorter contention times. Nodes receiving the message not positioned in the forwarding area of the sending node and currently in the contention state consider the message to be an implicit acknowledgment and increment the message counter. A node cancels the contention process when the message count is greater than a predetermined maximum. The message count is incremented each time a node transmits or receives a message from its own forwarding area.

The performance of EMDV is evaluated when operating the D-FPAV (Distributed Fair Power Adjustment for vehicular environments) algorithm which is developed by the same authors and controls the loading of periodic beacon messages on the communication channel. The D-FPAV algorithm is discussed further in §3.1.3. The authors investigate probability of reception, number of retransmissions and message delay whilst varying the the maximum number of times a node is allowed to retransmit a message and the maximum forwarding range in addition to varying the levels of fading on the radio propagation channel [65]. The maximum wait time was set to 100 ms and the max channel access time to 10 ms as appropriate values according to a study of one-hop broadcast communications [67]. With D-FPAV switched on, the probability of message reception is higher since it manages the beacon load more efficiently preventing collisions which would lead to more warning message retransmissions. Although increasing the maximum number of messages increases the reception probability with D-FPAV switched off, the number of warning message retransmissions increases significantly. The authors propose that the most efficient maximum message transmission for EMDV operating D-FPAV is 1 since 99.9% delivery rate can be achieved whilst requiring fewer warning message retransmissions. The EMDV

algorithm was tested under high vehicular traffic densities and does not consider overcoming network partitions, however, the authors suggest that the protocol could be amended to include a store-and-forward mechanism. Although the D-FPAV algorithm manages the beacon load to allow the EMDV algorithm to operate efficiently, the EMDV algorithm does not adapt its contention time to local traffic conditions which could further reduce the number of retransmissions. Moreover, the message forwarding process could die out before the message dissemination reaches the forwarding boundary since each node only transmits the message up to a pre-specified maximum number of times.

The MHVB protocol [68] aims to efficiently disseminate messages containing speed and velocity information for active safety applications, which are essentially beacon messages, over multiple hops. In [68] the MHVB protocol consists of two main mechanisms; a backfire algorithm and a traffic congestion algorithm. The backfire algorithm uses a distance based contention scheme which favours the next relaying node to be the furthest node away from the sending node. The transmission from the relaying node acts to suppresses retransmissions of the same message from nodes located in the region between itself and the previous sending node, which the authors refer to as "backfire" region. The backfire region is defined as a circular area positioned between sender and next relay node. The traffic congestion algorithm is used to detect vehicle congestion surrounding a node and enables it to decrease the frequency with which it transmits its own information which is inversely proportional to the number of surrounding vehicles. A node will adjust frequency of transmission when it detects the number of surrounding vehicles is above a certain threshold and its current speed is below a defined threshold. Through simple simulations [68] the authors found that the MHVB protocol did not scale satisfactorily as a result of packet losses due to collisions arising from too many packet retransmissions.

The MHVB protocol is enhanced in [69] in order to improve bandwidth utilisation efficiency. Firstly, the backfire region is defined in sectors via an angular parameter allowing control of the area and direction of the backfire region [69] in order to further limit the number of nodes which are allowed to retransmit. The second enhancement was to include a dynamic scheduling algorithm. The dynamic scheduling algorithm allows a receiving node located over 200 m away from the transmitting node to dynamically adjust the transmission time of its own information, so that the received information is retransmitted earlier. Simulations compared the performance of MHVB with and without the enhancements which were shown to improve the efficiency of MHVB.

Probabilistic Distance Based Approaches

In the case where nodes are co-located at the transmitting boundary of a sending node, multiple forwarding nodes may transmit the message if a discrimination technique is not employed. ODAM attempts to resolve this situation by allowing the node with the lowest ID to act as the forwarding node. However, this technique incurs unnecessary overhead in order to resolve the conflict. More recently, [70] refer to the potential effect of multiple relaying nodes as the 'spatial broadcast storm problem' and aim to alleviate it by proposing an extension to IVG which they call probabilistic IVG (p-IVG). The authors propose to make the rebroadcast decision of p-IVG probabilistic through knowledge of the surrounding vehicle density so that the number of vehicles which start their defer timers decreases with an increase in vehicle density. Vehicle density knowledge is learnt via the exchange of beacon messages. Reception rate, back-off rate and dissemination delay and hop count performance metrics are used to evaluate p-IVG against IVG and a simple flooding protocol. The results show that (p-IVG) adapts to varying traffic densities in terms of reception rates, whereas both IVG and flooding deteriorate with density as contention increases. The back-off rate in the case of p-IVG is considerably lower than IVG and flooding. Similarly, p-IVG outperforms IVG both in terms of dissemination delay and the number of hops required to cover the region. Clearly, the implementation of a mechanism which provides some form of discrimination between co-located nodes reduces channel contention resulting from unnecessary transmissions.

In [71] Chiasserini *et al.* aim to increase the timely delivery of safety messages by introducing a scheme at the MAC layer which provides channel access priority based on spatial differentiation. The spatial differentiation approach assigns different access priorities probabilistically through different contention window sizes. Essentially, the probability increases with the distance from the last relaying node and therefore nodes which are positioned further away have a higher probability of being selected as the next relaying node since they will have a shorter contention window. The authors develop a simplified analytical model for one dimensional vehicular networks and derive several metrics such as message blocking probability and average message delivery delay in order to study the performance of their MAC scheme. However, there are a number of assumptions which oversimplify this model: a discrete space is used to represent vehicle locations during end-to-end message relay time interval, an ideal radio environment is assumed and the effect of MAC collisions on protocol operation is ignored. Additionally, the authors do not consider an approach for overcoming network fragmentation and no knowledge of neighbourhood connectivity is maintained.

Three probabilistic distance based broadcasting suppression schemes are presented in [49]

aimed at reducing channel contention at the MAC layer. The schemes referred to as weighted p-persistence, slotted *1*-persistence and slotted p-persistence apply a combination of timer and probabilistic based methods in order to reduce the number of retransmitting vehicles. In the weighted p-persistence scheme whenever a vehicle receives a message it will rebroadcast it after defined waiting time according to probability P which depends on the distance between the transmitting vehicle and itself. Vehicles positioned farthest away are assigned a higher probability to rebroadcast the message than nearby vehicles. In the case of the slotted *1*-persistence and slotted p-persistence schemes, the waiting time is divided into slots. When a vehicle receives a message it will retransmit the message in an assigned time slot with probability 1 and a predefined probability, respectively, if it has not overheard another vehicle broadcasting the message during the waiting time. The further a receiving vehicle is from the transmitting vehicle the shorter the waiting time. The authors of [49] also quantify the impact of the broadcast storm problem in a four lane highway with varying traffic densities through metrics such as message delay and packet loss rate. The values for the average MAC delay are incorporated into the calculation of the waiting time for the comparison of the previously mentioned suppression schemes. However, this scheme does not provide sufficient statistics for local neighbourhood distribution and only provides the average number of vehicles for the length of the highway simulated.

Local vehicle traffic dynamics are used in [72, 73] to probabilistically select the next retransmitting nodes in order to reduce network contention time and the number of redundant retransmissions. The Optimised Adaptive Broadcast (OAPB) scheme [72] dynamically adjusts rebroadcast probability and delay according to an estimation of vehicle density within a two hop distance from each node. Local information is determined through the exchange of beacon messages. The performance of OAPB is compared against a deterministic broadcasting (DB) scheme with fixed retransmission probability. In comparison to DB with a high retransmission probability, the delivery rate is comparable. However, DB has a considerably higher overhead rate as congestion increases, whereas OAPB's overhead rate is around 70% lower. The results show that applying adaptability to the retransmission probability significantly reduces the overhead in comparison to a scheme that is not able to adapt to local variations. On the other hand, the Speed Adaptive Probabilistic flooding (SPAF) algorithm presented in [73] uses vehicle velocity to dynamically adjust the probability of a node retransmitting a message for high vehicle densities.

In the Receipt Estimation Alarm Routing (REAR) protocol presented in [74], next forwarding nodes are selected based on an estimated message receipt probability. This is evaluated dynamically from knowledge of received signal strength and packet reception rates through the exchange of beacon messages between neighbouring nodes. A node

is favoured through the contention delay as the next relay node if its neighbours have the highest receipt probability. The performance of REAR is compared against a distance contention based scheme. In the simulated scenario REAR is found to outperform the distance based scheme in terms of reachability and the number of broadcast packets. However, REAR was found to have a comparatively long delivery latency.

3.1.2 Clustering Based Approaches

Clustering protocols organise vehicles into a hierarchical structure by grouping vehicles into clusters so as to reduce local data traffic overhead. Clusters can be categorised as being either mobile, where clusters move with vehicles, or fixed to specific locations [12].

The BROADCOM protocol presented in [75] establishes a virtual infrastructure which utilises mobile cell clusters for the dissemination of emergency messages. The highway is partitioned into virtual cells and communication between cells is channelled through cell reflectors. The length of the cells is selected to approximate the communications range achievable by the physical layer to allow optimum transmission and reception. The virtual infrastructure is created though a two step process. Firstly, the virtual cell structure is created by an initiating vehicle (located in cell 1) which broadcasts its location. On reception of the message from the initiating vehicle, vehicles determine the number of the cell they are currently located in given the distance to the initiating vehicle, divided by the cell size. Vehicles exchange hello messages containing speed, position and cell number in order to establish members within the same cell. Secondly, the vehicle(s) located closest to the centre of the cell elect themselves as a reflector and broadcast their identity to their members and members within range in the adjacent cells. Reflector vehicles are maintained through periodic updates and any changes in the reflector vehicle(s) are coordinated within a cell, between its members. The vehicle which originates a warning message broadcasts the message to all the members within its cell. The cell reflector receiving the message then broadcasts the message to other cell reflectors who then broadcast the message to their cell members and the process continues along the highway. The virtual infrastructure is only defined in [75] for one direction of traffic flow. Although BROADCOM can potentially reduce local overhead through the cell reflector acting as the relaying node, there is no guarantee of successful delivery and no stated method for adapting the retransmission to local conditions. In highly dynamic traffic, membership within the cells would be changing faster and the time taken to elect a cell reflector would compromise the timely delivery of the warning message. Moreover, BROADCOM is not able to overcome network partitions.

In [76] and [77] the authors propose alternative architectures for the formation of mobile and stationary clusters, referred to as Local Peer Groups (LPGs). In [76] the authors

aim to coordinate communication within an LPG using intra-LPG communication at the link/MAC layer in order to support 'near instantaneous safety applications' requiring a latency < 100 ms. Coordination between LPGs is proposed through a backbone controlled at the network layer using inter-LPG communication for applications requiring dissemination further along the highway. In the case of stationary LPGs the authors propose to partition the highway using a GPS-based grid which is overlaid onto an onboard mapping database. As a baseline option the authors suggest that dynamic LPGs could consist of neighbouring vehicles for information dissemination without the need for further internal organisation. However, they suggest two further techniques called relative ordering (LPG-RO) and linked equivalent cells (LPG-LEC) for the dynamic formation of LPGs to further improve message dissemination, direction, prioritisation and bandwidth efficiency for message relaying. In LPG-RO vehicles are grouped according to a vector such as traffic direction, vehicles then maintain relative ordering within the LPG through periodic exchange of relative position within the LPG. Messages can then be relayed by vehicles located in the direction specified in the message. The authors state that the exchange of relative ordering messages for dense vehicle networks within an LPG will have a high overhead and consequently propose to partition an LPG into smaller groups called equivalent cells (ECs). ECs consist of vehicles within communication range of each other. Vehicles within an EC maintain relative ordering, however in order to increase bandwidth efficiency only one vehicle is responsible for relaying a message known as an EC header ECH. Each ECH periodically broadcasts its list of linked ECHs in order to direct message dissemination within an LPG.

Although stationary clusters do not require overhead and delay in initialising and organising LPGs, synchronising and updating on board databases will be both a costly and complex procedure. On the other hand the dynamic organisation of LPGs incurs overheads. Although LPG-LEC provides an option to reduce organisational overhead and the number of relaying message broadcasts, the size of an LPG and the periodic exchange of messages between ECHs in terms of timing and update overhead still remain problematic.

In [78], Blum *et al.* present an algorithm called COIN for maintaining more stable clusters in vehicular networks. A cluster is defined by all nodes travelling in the same direction and within radio range. Cluster head election is based on vehicular dynamic and driving intensions instead of relative mobility and ID. COIN also takes into account the oscillatory nature of inter-vehicle distances and specifies a minimum distance for inter-clusterhead distance. COIN maintains longer lived clusters which reduces the overhead incurred by the cluster head election process [78]. Although COIN is shown to offer advantages in terms of messaging overhead when clusters are stable, cluster stability is contingent on a number of detailed observations of relative mobility of neighbours in addition to neighbouring node

intentions e.g length of time and heading for extended periods of time. However, the messaging economy benefit, once stable clusters are achieved, is somewhat offset by the initial cost of establishing clusters.

The authors in [79] propose a dynamic cluster-based method for data dissemination in vehicular networks. They present a Directional Propagation Protocol (DPP) which forms clusters between vehicles moving in the same direction. Each cluster maintains a cluster head and tail which are responsible for communication between adjacent clusters. The authors also consider network fragmentation and propose a store-and-forward technique to overcome such conditions. Message propagation arising from the DPP clustering scheme does not realistically model MAC contention delays, but only models message propagation speed as being bounded arbitrarily between maximum and minimum limits. As such, there are questions regarding the actual predicated cluster stability arising from this algorithm.

3.1.3 Adaptive Power Based Approaches

The adaptation of a node's transmission power is another method investigated by researchers in order to provide reliable and timely dissemination of vehicle hazard information. Power adaptation aims to reduce the interference between nodes leading to fewer packet collisions and hence unnecessary retransmissions. In [80], the authors investigated the effect of varying the transmission range on channel throughput and confirmed that a small transmission range is preferable for hop-to-hop message delivery, in order to reduce the number of packet collisions over longer transmission ranges. In contrast, the study in [81] investigated transmission range effect on fragmentation in the vehicular environment and showed that the fragmentation increases exponentially with a decrease in transmission range. Thus, power adaptation is essentially a compromise between reducing packet collisions whilst maintaining reliable network connectivity.

In [11, 65], the authors propose a distributed power control algorithm called Distributed Fair Power Adjustment for Vehicular environments (D-FPAV), which aims to control the channel load caused by the periodic exchange of beacon messages. D-FPAV controls the transmit power for periodic broadcasts to allow priority for the dissemination of hazard data using the EMDV algorithm, whilst ensuring fairness for beacon messages from different vehicles. D-FPAV algorithm requires a node to collect status information for all nodes within its carrier sensing (CS) range at maximum power. Where the node's in the CS range are outside a nodes transmission range the authors suggest using extended beacon messages where a node aggregates status information of corresponding nodes within its CS range. Based on this information a node computes a maximum common transmit power for all nodes within its CS range such that the beacon load on the channel does

not exceed a maximum threshold, referred to as the Maximum Beaconing Load (MBL). A node broadcasts the maximum computed transmit power to all nodes within its CS range and on reception of computed power from other nodes determines a final transmit power level which is set to the minimum value received from these nodes and its own computed value. Power level is adjusted when channel utilisation increases in order to operate below the MBL value. The value of MBL is not adjusted dynamically in this work; the authors set this value as a maximum channel load.

Node density is used in [82] to adapt node transmit power for reliable dissemination of vehicle safety messages. The power control algorithm presented in [82] calculates transmit power dynamically through the exchange of beacon messages, which includes the power at which the transmitter broadcasts the beacon message. Each node maintains path loss information for each entry in its neighbour table averaged over successive receptions which are also classed according to their path loss value. When a node is required to broadcast a message the power control algorithm will determine the minimum transmit power required to reach a specified "target" number of neighbour nodes. The transmitter will increase its power until the cumulative number of nodes in each class reaches the "target" number. In this way the transmit power will vary according to node density. The authors use a simple restrictive flooding algorithm [82] for the dissemination of safety messages and evaluate message reachability, message redundancy, average power and average delay metrics with fixed transmit power and variable power using their power control algorithm. Message redundancy was found to remain constant as density increases for variable transmit power since the transmit power is reduced, which in turn reduces the number of collisions and hence the number of retransmitting nodes. Although the authors demonstrate that their power control algorithm is adaptive, the evaluation was performed using a simple simulation scenario and unrealistic vehicle movement. Moreover, the determination of the "target" number of neighbour nodes in practice along with the size of the path loss classes requires further investigation in order to make them adaptive to realistic scenarios.

Similarly, in [83, 84], transmission range is adapted dynamically based on an estimation of node density. The Dynamic Transmission-range Assignment (DTRA) algorithm presented in [83, 84] utilises a relationship derived from an analytical traffic flow model in order to derive and estimate local density in conjunction with parameters from the vehicle road traffic simulator, RoadSim. The aim of DTRA is to maintain a high level of connectivity through the estimation of vehicle density in free flowing versus congested traffic. Performance simulations are carried out to determine minimum transmission range and number of partitions for different traffic densities in in-homogeneous traffic. The communication model is based on a simple Euclidean geometry model whereby nodes can communicate with each other if the distance between two vehicles is less than the transmission range.

However, this model omits any interference effects caused by nodes competing to access the communication channel.

In [85], Yang *et al.* propose a one-hop channel adaptive power control algorithm for the broadcasting of vehicle status information (i.e. beacon messages). A vehicle determines channel conditions (i.e. packet collisions and data traffic load) by analysing the overhead sequence numbers in the status message sent by each neighbouring node over a time window. By identifying and counting the successfully received packets, the receiving vehicle can detect failed transmissions and determine network conditions from the average successful and unsuccessful reception rates. The number of transmitting nodes can also be determined from this analysis. The calculated channel conditions are then compared to two thresholds, vehicle traffic load and a target reception rate in order to adapt the transmit power accordingly to utilise the channel efficiently. Yang *et al.* firstly carry out simulations to determine reception rates with the adaptive power mechanism turned off, in order to investigate reception rate versus vehicle density for various transmit powers between the minimum and maximum power. Results from the reception rate simulations provide threshold values for the adaptive power algorithm. The performance of the adaptive power algorithm is compared against a non adaptive algorithm and D-FPAV and the reception rate of beacon messages versus distance is investigated. The adaptive algorithm was found to perform slightly better than D-FPAV within a range of 150 m. Beyond this range D-FPAV outperforms the authors' algorithm as a result of its more accurate channel load information and ability to calculate a more optimal transmit power.

Mittag *at al.* [86] aim to further reduce the overhead generated by D-FPAV in order to maintain a network wide MBL and introduce Distributed Vehicle Density Estimation (DVDE) and Segment-Based Power Adjustment for Vehicular environment (SPAV) protocols. DVDE provides an approximation of surrounding vehicular traffic conditions up to two carrier sense ranges away from each vehicle through partitioning the area up to the maximum transmission range into segments. Each vehicle derives a vehicle density histogram which is based on information received from beacon messages according to the number of vehicles located in each segment. Similar to the extended beacon strategy used in D-FPAV, DVDE piggybacks the vehicle density histogram every 10 beacon messages. Vehicles receiving the extended beacons merge received histograms in order to approximate traffic densities beyond their current transmission range. This approximation is then used by the SPAV protocol to adjust transmission power to ensure the surrounding cumulative load generated by all vehicles is lower than the MBL threshold. The simulation results showed that DVDE/SPAV can be used to reduce the existing D-FPAV overhead. However, D-FPAV still scales linearly with the number of nodes within its CS range.

3.1.4 Congestion and Channel Aware Approaches

The previous sections of this chapter have reviewed data dissemination schemes which predominately aim to reduce network congestion of event based safety messages, by controlling the number of retransmitted messages. However, many of theses schemes are not able to adapt to increased channel congestion since they do not monitor local channel conditions. Moreover, in high density vehicle conditions this can lead to unnecessary retransmissions as a result of increased local network channel usage. Although adaptive transmit power schemes can control the number of neighbours per unit node and hence reduce local network congestion, this occurs at the expense of requiring more retransmissions of event driven messages. In the case of contention-based forwarding schemes, a limited number of schemes are able to adapt retransmission timing dynamically, through the monitoring of vehicle density and/or network loading which helps to reduce unnecessary retransmissions in dense traffic conditions.

However, the predominant source of network congestion for safety related messages is through the periodic exchange of beacon messages. In dense traffic conditions beacon messages may consume the entire channel bandwidth resulting in a saturated or congested channel. In such a congested channel event-driven messages may not be able to access the communication channel. The majority of research which considers the state of the communications channel from the aspect of safety applications focuses on reducing congestion generated by periodic messages to ensure that sufficient channel bandwidth is available for event driven safety messages. These congestion control schemes vary either transmission power or beacon packet generation rate to meet their aim. In order to detect the level of channel congestion different metrics such as channel busy time, delay, node density, message reception probability, message utility, etc., are used in the literature [11, 86, 87, 88, 89, 90, 91, 92, 93]. This section briefly reviews a selection of these schemes.

Both the D-FPAV and SPAV/DVDE schemes previously reviewed in §3.1.3 adjust transmission power in order to maintain a maximum beaconing load. The idea behind defining an MBL is to reserve a portion of the bandwidth for event driven safety messages so that communication of safety messages is not hindered by channel saturation. As discussed in §3.1.1, Torrent-Moreno [10] *et al.* evaluate the performance of their dissemination scheme EMDV for event driven safety message in conjunction with D-FPAV, and prove that control of the periodic messages improves the comparative performance of EMDV with and without the operation of D-FPAV. However, channel knowledge from D-FPAV is not used in the EMDV scheme, which could further improve efficiency and reliability of event driven message delivery.

The authors of [92] propose event-based and measurement-based congestion detection

schemes by controlling congestion via MAC queue manipulation to ensure reliable and timely delivery of safety messages. In the case of event driven detection, congestion control is activated immediately whenever a safety application message is detected, whereas the measurement based technique periodically senses channel usage and activates congestion control when a predefined channel usage threshold is exceeded. The authors propose congestion control approaches based on queue freezing and adaptive QoS-based MAC queue manipulation techniques. Through simulation the authors find that the adaptive QoS congestion control technique is more efficient than freezing MAC transmission queues. However according to [87], the effective transmission of safety messages is not guaranteed because the neighbourhood context and effective bandwidth sharing are not considered.

In [87], the authors propose a cooperative congestion control approach based on priority estimations which considers dynamic factors such as node speed and message validity, as well as message utility, network load and neighbourhood context. Additionally the authors propose an approach for next forwarding node selection which selects the least congested node. The congestion level for each node is maintained in the neighbour list. A node determines its congestion level parameter by evaluating the time taken to send all queued messages over the available bandwidth. A node then informs its neighbouring nodes of its current congestion level by including it in the beacon message.

Although the utility-based congestion control scheme in [91] focuses on non-safety related applications it is worth a brief mention. The authors present a scheme which adapts transmissions to the available bandwidth in a hop-by-hop manner. Priority is evaluated for each packet depending on its utility and size. Nodes with a higher utility are allocated a larger share of the available bandwidth. However, message utility does not take into account local vehicle dynamics such as speed and density, and moreover, is determined by segmenting the road into sections. The segmentation technique cannot be used directly in the context of safety applications because of the additional latency and overhead involved in implementing such a scheme.

Channel Occupancy time is used by He *et al.* in [89] as the congestion threshold mechanism. If the channel occupancy time is detected to exceed the threshold time a MAC blocking mechanism is used for immediate control of periodic beacon messages. In addition He *et al.* [89] propose a cooperative adaptive traffic rate control for congestion avoidance in which a blocking node notifies its neighbours of MAC blocking, who then respond by controlling their traffic rate.

3.2 Overcoming Network Partitions

When a node cannot forward a message beyond its current position because it does not have any neighbouring nodes within its transmission range, network connectivity is considered to be partitioned or fragmented. As a consequence of the limited communication range between nodes and the dynamic nature of vehicular traffic flow, network partitions will occur within the message dissemination region when vehicle traffic density is low, vehicles travel in disconnected clusters or the number of equipped vehicles is low. Therefore, as well as ensuring that the forwarding algorithm efficiently restricts the number of redundant message rebroadcasts and minimises network access contention for all dynamics of vehicular traffic flow (particularly in the case of congested traffic which presents worst case network loading and contention conditions), the scheme must also be able to overcome fragmentation in order to deliver the message reliably to all vehicles within the addressed region.

The general approach used by researchers in the vehicular *ad hoc* networking community to overcome network fragmentation is to use opportunistic broadcasting which is one of a larger class of techniques used in delay-tolerant networks (DTNs) and intermittently connected networks [94]. In the vehicular environment DTNs are generally considered for the dissemination of messages for low priority applications, which are do not consider here. In opportunistic broadcasting a store-and-forward technique is used when a relaying vehicle encounters a network partition [95, 96]. The relay node will store the message until an opportunity arises to forward the message to another vehicle which can further progress the message dissemination using the forwarding algorithm towards the boundary of the addressed region. There are essentially three approaches used in the VANET literature to overcome fragmentation: periodic retransmissions; neighbour knowledge; and infrastructure based.

Periodic Techniques: If the underlying message forwarding scheme does not maintain local neighbourhood information at each node then a relay node will have no knowledge of the local topology so as to detect when network fragmentation occurs. In this case, a relaying node which has not overheard the message being forwarded after n retransmission attempts assumes that a partition has occurred and will store the message and retransmit it periodically [54, 55, 56]. The periodicity of message retransmission, during a partition, is generally decreased in comparison to the retransmission timing used in the forwarding algorithm, in order to reduce the number of redundant retransmissions. For instance, in [55] the retransmission timing during a partition is reduced to a time associated with the time it would take for a node at a maximum speed entering its communication range to reach the location of the node at the time the retransmission was scheduled.

Neighbourhood Techniques: In the case of schemes which maintain neighbourhood information, a relay node is able to detect from its neighbour table when a partition has occurred when there are no entries further forward than its current position in the forwarding direction [40, 57, 64, 97]. The relay node will store the message until it detects from its neighbour table that an entry exists in the required forwarding direction. Thereafter, the relay node will revert back to the forwarding algorithm to continue message dissemination.

Infrastructure Based Techniques: Dissemination schemes which rely on infrastructure such as roadside repeaters to overcome network partitions may be appropriate for urban settings such as at intersections [60]. However, this can be a prohibitively costly technique both in terms of implementation and maintenance, particularly in a highway setting which is the focus scenario for the research in this thesis.

Schemes such as [55] restrict the selection of the relaying node to one direction of traffic flow either when a partition is detected or as a rule of the forwarding protocol. This can lead to unnecessary and longer lived partitions which can increase the end-to-end dissemination time. The opposite traffic flow can be used to overcome the partition and reach the next cluster of vehicles with a shorter delay in comparison to the time that it would take for a node travelling in the same direction as the relay node to enter the communication range. Schemes such as [57] use bidirectional traffic flow to overcome partitions in order to bridge the gap between partitions.

Although periodic retransmission overcomes partitions, it does so at the cost of additional overhead and is, therefore, a balance between protocol overhead efficiency and increased length of end-to-end delay. Schemes such as [56] which increase the length of time between periodic retransmission according to a function which uses static values for maximum transmission and communication range, risk having vehicles not receiving the message in time. This is because, in reality, vehicles invariably exceed the speed limit in a highway scenario. In addition, a reduced communication range could mean that a vehicle entering the distance defined by the maximum communication range will not receive the message and by the time the next retransmission occurs it will have reached or passed the relaying vehicle. On the other hand, detecting and overcoming network partitions using neighbourhood knowledge information is a more efficient and reliable technique of overcoming partitions. However, this is at the expense of an additional overhead through the exchange of beacon messages. Although the maintenance of neighbourhood information has a large overhead on network resources, it is a generally accepted requirement for emergency vehicular applications [65] and can therefore be considered as a shared protocol overhead.

3.3 VANET Security Challenges

The issue of providing secure communications for vehicular *ad hoc* communications is very important since several strict requirements must be met before the deployment of VANET applications can be realised. These requirements include user and data authentication, privacy, non-repudiation and secure communication [98]. Satisfying these requirements in a highly dynamic and mobile vehicular *ad hoc* network is particularly important since a compromised VANET could result in serious or fatal consequences. Although privacy and liability are integral aspects in defining the broader security aspects for VANETs, the focus in this section is the implications on security of data dissemination for geographic based safety applications. To this end, potential threats are considered from adversarial nodes to the dissemination of event driven geocast messages and periodic beacon messages using the threats presented in [99][1] as a baseline for discussion. The following adversarial threats are illustrative of the general problems within the area of vehicle security and are not intended to be a comprehensive review, which is outside the scope of this thesis. The reason for addressing security issues is simply to highlight the need to integrate security into a number of communications layers, including the routing one.

False Position Advertisements: *(ibid) "The attacker claims to be at a different position than its actual one, e.g. by including it in a beacon or data packet."* Broadcasting incorrect location information in a beacon packet will result in neighbour nodes having a false view of local connectivity. This will result in nodes making incorrect forwarding decisions for event driven packets leading to increased forwarding latency causing unnecessary packet retransmissions from intermediate nodes.

Geographic Sybil Attack: *(ibid) "The attacker advertises multiple IDs and/or positions, to mislead other nodes that high number of (non-existent) neighbours exist. Communication across non-existing nodes is in full control of the attacker; e.g. forwarded packets will be lost."* A fictitious node which is expected to act as next forwarding node could lead to increased retransmissions from intermediate nodes. Moreover, if retransmission times are adaptive to local vehicular density then retransmission times will be increased. Additionally, the attacker could rebroadcast an event driven packet pretending to be the fictitious forwarding node with knowledge that there is a partition ahead. This would prevent further dissemination of the event driven message throughout the addressed region.

Packet Alteration: *(ibid) "An attacker changes the content of the header or payload of the packet it forwards."* The attacker could modify location information of beacon packets it receives and then rebroadcast them leading to false updates in neighbour tables. Again this can lead to unnecessary retransmissions when local connectivity is used to make

[1] Italicized text in this section is quoted verbatim from [99]

forwarding decisions. An attacker node could also alter the position of the last forwarding node and rebroadcast the packet. This would lead to receiving nodes making incorrect decisions of their role in the forwarding of the event-driven packet.

Packet Dropping: *(ibid) "Adversaries selected as forwarders can simply drop packets, either all (black-hole attack) or selectively (grey-hole attack)."* A node which is required to relay the message may decide to drop the packet, this would result in additional latency or could cause a network partition.

Replay: *(ibid) "The Attacker re-injects previously received packets into the network."* Retransmitting previously received beacon messages will lead to nodes making false updates and retransmitting event driven messages will increase local congestion.

Packet Injection Attackers could transmit packets at high-rates to consume bandwidth and computation in large parts of the network. This would cause a denial of service (DoS) attack preventing the dissemination of the event driven message and exchange of beacon messages providing an outdated view of local connectivity.

Security architectures and services are and have been addressed by various standardisation bodies and research projects such as the IEEE 1609 working group which developed IEEE 1609.2 and addresses security aspects of DSRC-WAVE, ETSI and various EU research projects such as SeVeCom [100] and on-going in the Car-to-Car Communication Consortium (C2C-CC) [9] and the eSafety Security Working Group [101], amongst others. However, the security aspects of IEEE 1609.2 are restricted to single hop communications and do not address the specific security aspects of geocasting [102]. Such research issues are currently on going within standardisation bodies, industry and academia. The authors of [102] and references therein address geocast security design for geocast communication and consider similar adversarial threats to those discussed above. Moreover, current proposed security solutions for secure beacon exchange often require each message to be signed and carry a certificate to ensure integrity and authenticity which increases the beacons size creating a significant protocol overhead. Research in this area therefore focuses on reducing overheads from certificates and signatures. For instance [103] investigates reducing overhead whilst maintaining a comparable level of security by omitting signatures, certificates or certificate verification in situations where they are not necessarily required. Further information on security challenges, design and architectures can be found in [98, 104, 105, 106].

3.4 Discussion on VANET Protocols

The reviewed protocols are summarised in Table 3.1. The protocols are classified according to mechanisms they use, ability to overcome partitions, mobility model used in simulations and their ability to dynamically adapt to local conditions. Location and position based methods indicate that messages are broadcast based on the geographic area of the transmitting and receiving vehicles in the addressed area. Distance based methods broadcast messages by considering distance or hop count from the transmitting node. In cluster based approaches a message is broadcast within a cluster group. In probabilistic methods a node will broadcast with a given probability, usually based on a back-off timer related to distance. In adaptive power transmission schemes, transmission power is varied to reduce congestion.

As can be see from Table 3.1, the protocols differ in a number of assumptions underpinning their design: some protocols advocate the use of hello beacons to ensure that the presence and location of vehicles on the highway is explicitly declared to their neighbours, whereas others deem these messages as being a waste of bandwidth. Many protocols go to great lengths in order to avoid the broadcast storm problem by restricting the number of nodes that are allowed to rebroadcast the message and in the choice of nodes which forward the message. Typically, they do so by employing contention based schemes incorporating a distance based mechanism which favours the relaying of a message by one of the most distant vehicles within the coverage area of the previous relaying vehicle. This can be attempted deterministically or probabilistically and the corresponding functionality can be incorporated within the MAC or network layer.

A much more subtle set of differences arises due to assumptions made about the accuracy with which vehicles can measure their location, whether the communication radius of vehicles is known *a priori* and is constant for all vehicles, or even at all times for the same vehicle, whether there is perfect capture for the radio channel, whether the radio channel is error-free, how network disconnections/partitions and reconnections are handled or otherwise.

Other differences between protocols include whether they incorporate directional message reception information or vehicular direction of motion information to implement data forwarding more intelligently. Furthermore, protocols may use geographic and directional information to distinguish not only the area relevant to message dissemination but also a zone of forwarding. Finally, protocols may treat the network as static during end-to-end message relaying time intervals, or may choose to exploit the fleeting dynamic nature of vehicle-to-vehicle communication links and in some cases adapt protocol decisions to local variations in vehicular node density.

Many of the contention based dissemination schemes do not consider the dynamic affect of both local traffic generation rates and vehicle density on the deferral retransmission timing. Omitting the actual local access delay from the calculation of the retransmission deferral time can result in packet collisions, unnecessary retransmissions and longer latency times. Although the majority of schemes include a parameter which allows for access delay, this value is generally static and will either provide unnecessary longer retransmission delays or delays which are too short compared with the actual MAC access delays leading to the above mentioned consequences. However, although ODAM incorporates MAC access delay based on received packet delay, its accuracy is questionable since the sampling time window over which MAC access delay is determined will need to be large in order to capture local variations from event driven messages. Additionally, the minimum deferral is unlikely to provide sufficient time discrimination to provide distance ordered retransmission, which will result in unnecessary retransmissions.

In addition many schemes do not ensure that the dissemination of a message proceeds reliably; packet collisions coupled with vehicle movement may cause the dissemination process to fail. Some schemes allow an intermediate and/or a forwarding node to retransmit n times and assume that this will mitigate for such occurrences. In reality, collisions may occur as a result of the hidden terminal problem resulting in dissemination failure. Some schemes allow a previous forwarding node to continue retransmitting repeatedly until it overhears a successful retransmission. Although these methods provide reliability at the expense of additional overhead, they do not consider the case where a forwarding node did not overhear dissemination continuing successfully and therefore continues retransmitting erroneously.

Reducing data congestion in high density scenarios has generally concentrated on lowering overheads generated by beacon messages either by reducing the transmission power or by varying the frequency of beacon messages. These schemes tend to control channel load dynamically through parameters such as traffic density, whilst others monitor received packet rates. Congestion control for event driven messages has not been a focus since beacon messages are considered to incur the biggest overhead, however, unnecessary retransmissions at a local level from event driven messages can severely add to data traffic congestion in a congested network.

In this thesis distance deferral contention-based forwarding is used and the issue of adapting the retransmission deferral time to *both* local vehicle *and* data traffic variations is addressed. The essential difference between the research in this thesis with respect to the previous cited work is that distance deferral resolution is implemented via strict time dilation determined through parameters monitored locally in conjunction with an empirically derived model of MAC channel access delay. The aim of this mechanism is to prevent

unnecessary congestion of event driven messages by reducing the likelihood of unnecessary retransmissions. Additionally, it is ensured that the dissemination process does not fail by allowing a forwarding node to retransmit until it successfully receives an implicit acknowledgement from the forwarding direction. However unlike the cited schemes, the work in this thesis considers the effect that these erroneously retransmitting nodes have on congestion and implement a mechanism which actively suppresses them when they occur. This research also considers variable vehicular network densities in the development of the protocol and, therefore, a a store-and-forward mechanism is implemented which is able to overcome network partitions resulting from low vehicle density in a dissemination region. Moreover, in contrast to the other work cited here, this research considers minimising data congestion whilst maintaining high levels of reliability for all vehicle traffic dynamics for each protocol mechanism.

In earlier work presented in [97] an area based broadcast dissemination protocol was proposed called Distance Deferral Forwarding (DDF) which selects the next relay node based on maximum progress in the forwarding direction. The deferral timing is adaptive to traffic density and data interarrival rate in order to limit redundant retransmissions from intermediate nodes, through ensuring that their retransmission deferral time allows for variations in MAC access delay within the region between the previous and next forwarding node. In order to expedite the dissemination of the warning message the next relay node retransmits the packet without delay. A node implicitly determines that it is the next relay node by comparing its current position with its neighbour nodes positioned between itself and the previous relaying node in the forwarding direction. Each position comparison with a neighbour node is corrected to allow for node movement between neighbour table updates. In order to ensure forwarding reliability an intermediate will retransmit the message n times. On the occasion that a node determines that it does not have any neighbours in the forwarding direction, a partition is assumed and the node will store the message until it detects a node to be further forward in the forwarding direction. The adaptive MAC access delay was determined empirically through extensive simulations of one-hop beacon messages using a calibrated microscopic vehicle simulation tool with varying vehicle density and packet interarrival rates.

The following chapters present, in more detail, the development of the ideas presented in [97]. Chapter 5 provides a detailed description of the mechanisms and operation of the DDF protocol proposed in this thesis and Chapter 7 presents the empirical modelling of MAC access delay.

Protocol Name	Location/Position Based	Distance Based	Probabilistic Based	Cluster Based	Adaptive Trx Power	Neighbour Knowledge	Overcomes Partitions	Propagation Flow	Environment	Mobility Model	Dynamic Adaptability
TRADE (2000)	✓	✓	X	X	X	✓	X	Single	Highway	Traffic simulator	None
DDT (2000)	✓	✓	X	X	X	X	X	Single	Highway	Traffic simulator	None
RBM (2000)	✓	✓	X	X	X	✓	✓	Single	Highway	Traffic model	None
IVG (2003)	✓	✓	X	X	X	X	✓	Single	Highway	Simple	None
ODAM (2004)	✓	✓	X	X	X	X	✓	Single	Highway	Simple	Uses Mean of communication delay in defer time calc
DDF (2005)	✓	✓	X	X	X	✓	✓	Both	Highway	FloWSim traffic simulator	Uses vehicle density and packet interarrival rate along with empirical MAC model to derive channel access delay to adapt defer time calc.
DV-CAST (2007/2010)	✓	✓	✓	X	X	✓	✓	Both	Highway		None
TRRS/ETRRS (2007)	X	✓	X	X	X	X	X	Single	Highway	Simple	None
UMB (2004)	✓	✓	X	X	X	X	X	Single	urban	Simple	None
SB (2006)	✓	✓	X	X	X	X	X	Single	N/S	simple	None
FB (2006)	X	✓	X	X	X	✓	X	Single	N/S	Simple	Adapts trx range parameter in defer calc, from data in received packets
REACT (2007)	X	✓	X	X	X	✓	✓	Single	Highway	Cellular Automata vehicle model	None
D-FPAV (2006)	X	X	X	X	✓	✓	N/A	Both	Highway	DaimlerChrsler traffic simulator trace files	Adapts transmission power to ensure a minimum beacon loading
EMDV (2007)	✓	✓	X	X	X	X	X	Both	Highway	DaimlerChrsler traffic simulator trace files	NONE
MHVB (2006)	✓	✓	X	X	X	✓	X	N/S	N/S	Traffic simulator	Adapts frequency of beacon messages according to vehicle density and velocity
P-IVG (2009)	✓	✓	X	✓	X	✓	✓	Single	Highway	SWANS-ASH	Vehicle density used to adapt probability of number of vehicles starting defer timer
[71](2006)	✓	✓	✓	X	X	X	X	Single	N/S	Stationary	None
OAPB (2005)	✓	X	✓	X	X	✓	X	Single	Highway	Simple model	Rebroadcast probability adapts dynamically to vehicle density
SPAF (2008)	✓	X	✓	X	X	X	X	Single	Highway	VISSIM traffic simulator	Rebroadcast probability adapts dynamically according to vehicle speed
BROADCOM (2005)	N/S	X	X	✓	X	✓	N/S	Single	N/S	None	None
COIN (2003)	N/S	X	X	✓	X	✓	X	N/S	Highway	CORSIM traffic simulator	None
DPP (2005)	✓	X	X	✓	X	✓	✓	Both	Highway	Simple, constant velocity	Cell size adapted to vehicle density

Protocol Name	Location/Position Based	Distance Based	Probabilistic Based	Cluster Based	Adaptive Trx Power	Neighbour Knowledge	Overcomes Partitions	Propagation Flow	Environment	Mobility Model	Dynamic Adaptability
LPG-LEC	✓	X	X	✓	X	✓	X	Single	Highway	N/A	None
[82] (2005)	N/S	X		X	X	✓	✓	X	N/S	None	Transmission range adapted to path loss
DTRA (2005)	N/S	X	X	X	✓	X	X	N/S	Highway	RoadSim	Transmission range adapted to vehicle density estimation
[85] (2008)	X	X	X	X	✓	✓	N/A	Both	Highway	Traffic simulator trace files	Transmission range adapted to channel conditions determined from packet loses
DVDE/SPAV (2008)	X	X	X	X	✓	✓	N/A	Both	Highway	FleetNet movement trace files	Transmission range to vehicle density
REAR (2008)	✓	X	✓	X	X	✓	N/S	Single	Highway	FleetNet movement trace files	Probability of retransmission adapted to channel loss

Table 3.1: Comparison of protocol characteristics (N/S = Not Stated and N/A = Not Applicable)

3.5 Summary

This chapter reviews a number of approaches proposed in the literature for the geocasting of data between vehicles for safety related applications. The review is focused on various types of schemes which aim to reduce the number of retransmissions resulting from the inherent nature of broadcast communication through contention based schemes, cluster based schemes, power reduction schemes and congestion control. Although schemes which reduce the number of retransmitting nodes are considered to be a form of data packet congestion control, they do not, however, attempt to reduce overheads dynamically when vehicle density is high. Therefore, a brief review of congestion control used in the literature for vehicle to vehicle communications is provided. In addition, the security aspects of geocast communication and implications it may have on message dissemination are briefly considered.

CHAPTER 4

THE DISTANCE DEFERRAL FORWARDING PROTOCOL

4.1 Introduction

In this this chapter a data dissemination protocol for safety related IVC applications is presented that is capable of overcoming the challenges and meeting the requirements identified in Chapters 2 and 3. The proposed protocol is loosely based on the notion of distance deferral dissemination and aims to provide reliable, timely and efficient delivery by adapting to local data traffic conditions. This chapter begins by firstly summarising the requirements of the chosen IVC application scenario and then presents the reasoning behind protocol design decisions. The remainder of this chapter presents a detailed description of the operation of the proposed IVC dissemination protocol.

4.2 Application Scenario and its Requirements

Chapter 2 considered different IVC applications, their contrasting data delivery requirements and the challenges of operating in the vehicular environment. Its was decided to develop the proposed data dissemination protocol for the class of safety related applications requiring the dissemination of data within a geographical region in order to forewarn drivers/vehicles. The safety related message may contain data to forewarn the driver of an approaching or pending incident or hazard such as, a traffic hazard, accident or incident, localised weather conditions, e.g. ice, fog, snowstorm, etc, or simply congestion.

In this research it is assumed that vehicles or roadside units (RSUs) [1] are equipped with the relevant sensing technology which is able to detect particular incidents or hazards and

[1] RSUs are assumed to be equivalent to stationary nodes

are therefore the source of the warning message. The application layer is responsible for defining the affected geographical area over which the warning message is to be disseminated. Since the focus of this research is primarily at the network and MAC/PHY layers, the application layer is emulated by randomly varying the size of the geographical area addressed by the warning message.

In §2.4.2 an application specific region to which a message is relevant is referred to as the RZR. Since both directions of traffic flow are used to aid efficient dissemination of safety messages at the network layer there is a need to define the area over which the message is required to be disseminated, which is referred to as the Data Dissemination Area (*DDA*). This distinguishes the dissemination area from the application defined RZR. The RZR defines areas within the *DDA* where vehicles take any necessary action at the application layer according to specific application requirements. For instance, Figure 4.1(a) and Figure 4.1(b) show that $RZR \subset DDA$ and in Figure 4.1(c) the $RZR = DDA$. In Figure 4.1(a) the RZR is relevant to vehicles approaching the accident on both sides of the carriageway, whereas in Figure 4.1(b) the RZR is relevant to the traffic flow approaching the hazard. Vehicles within the RZR may be required to take action whereas the nodes in the *DDA* only, may require a notification of an nearby accident. In the case of Figure 4.1(c), since the *DDA* and RZR are the same all vehicles will take any necessary action defined at the application layer. This research is not concerned with evaluating the performance of individual applications, it is aiming to examine in detail the performance of the dissemination protocol alone, and this thesis shall henceforth only be concentrating on the *DDA*.

In addition to the challenges of implementing IVC over a decentralised network, as discussed in §2.4, safety related applications require messages to be communicated in a reliable and speedy manner. This is necessary to ensure that drivers are able to take appropriate and necessary action.

At the initial introduction stages of IVC applications, the number of equipped vehicles will be low and consequently demand on the communication channel will also be low. The challenging application scenario is when all vehicles are equipped and demand on the communication channel is high. This research focuses on the scenario where all vehicles are equipped with IVC technology, and examines protocol performance under various vehicular traffic flow conditions. This will allow a low level of equipped vehicles to be emulated when network partitions occur regularly during low traffic flow rates.

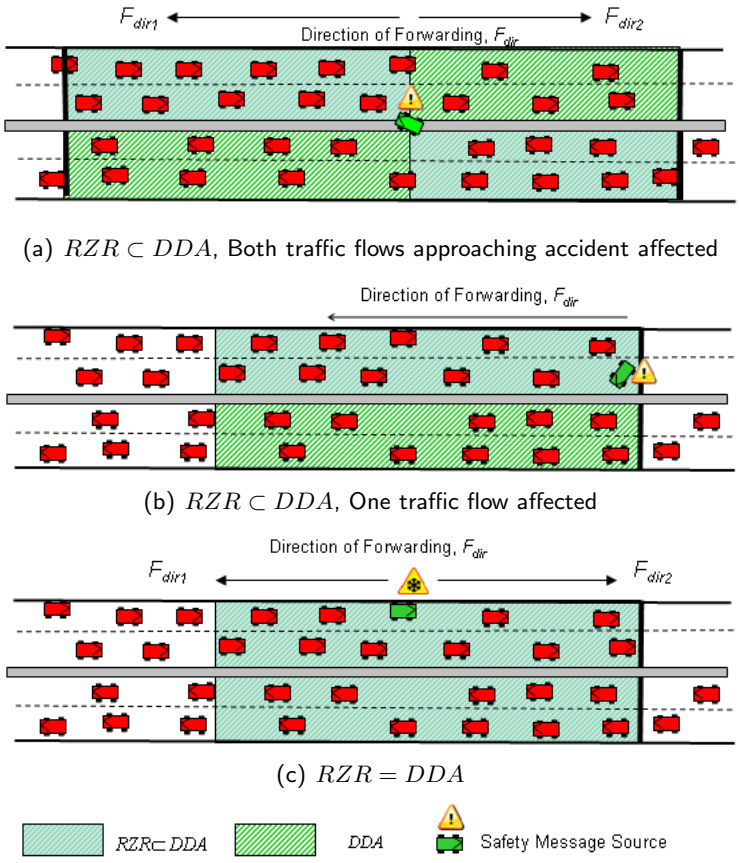


Figure 4.1: Possible variations in RZR in relation to the *DDA*

4.3 Design Decisions

The protocol design decisions taken to satisfy the requirements of the chosen application scenario, which enable the protocol to adapt to the highly dynamic nature of vehicular traffic flow and mitigate against the unreliable nature of the IEEE 802.11 MAC scheme in broadcast mode (see §2.3.3), are covered in the following section.

In order to disseminate the warning message to vehicles or drivers in a region or regions affected by an incident, this research uses a geographic dissemination protocol based on distance deferral techniques called Distance Deferral Forwarding (DDF). This technique has been chosen because it can be used effectively to minimise the number of transmissions whilst providing maximum coverage of over hearing a message in terms of geographic area, for a minimum number of retransmissions.

As can be seen from the literature review in §3.1.1, there are many vehicular based geographic dissemination protocols which utilise the spirit of distance deferral, however, they have different drawbacks in terms of economy of messaging, timeliness of delivery, reliability and scalability. For this reason, this research does not strictly adhere to the con-

ventional notion of distance deferral because of the uncertainties in timing (as a result of channel utilisation) and the propagation environment, which lead to the above mentioned drawbacks. A distance deferral based protocol is implemented where all nodes potentially have a role to play, given knowledge of their local environment. This ensures a balance between reliability, timeliness, and economy of messaging, whilst providing maximum possible coverage throughout the dissemination region for varying dynamics of vehicular traffic flow. The various mechanisms which constitute the DDF protocol in order to meet the previously mentioned aims, are shown in Figure 4.2.

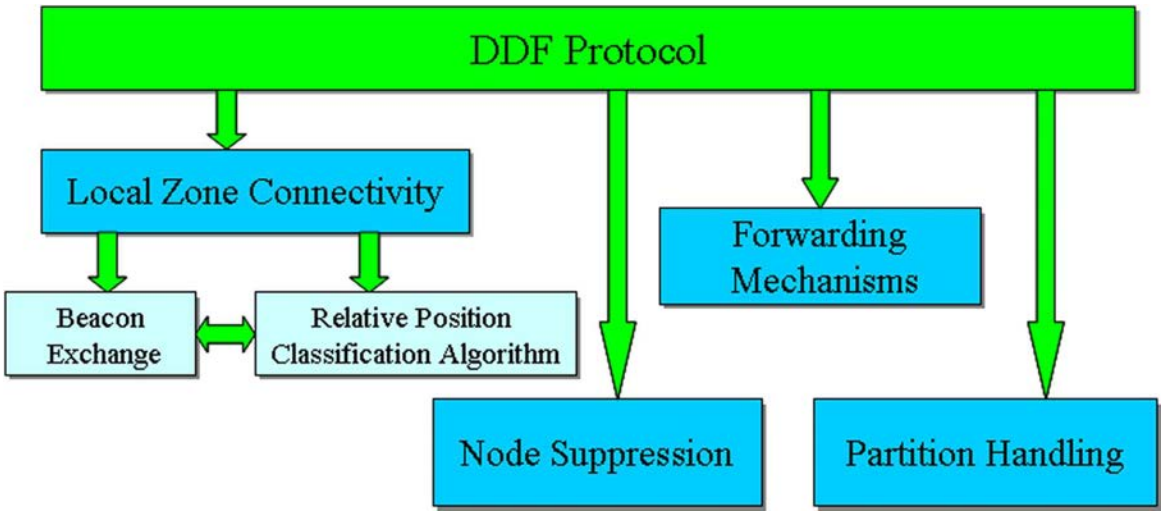


Figure 4.2: DDF protocol mechanisms

The design of the DDF protocol exploits the fact that vehicles are highly constrained to linear highways and that messages relevant to IVC applications need to be disseminated away from the source of the transmission. A key fact that enables the broadcasting scheme within the dissemination area to be simplified, is that the width of the highway and thus the dissemination area is smaller than the transmission radius of each equipped vehicle.

4.3.1 Aims of DDF Protocol

The following section gives a high level description of each mechanism shown in Figure 4.2, and describes how it achieves the aims of the DDF protocol. The specific implementation details for each mechanism and their operation, follows in §4.4.

Local Zone Connectivity

Basic distance deferral protocols, which do not maintain knowledge of local connectivity information, need to defer retransmission based on distance away from the source of the

transmitting node, in conjunction with a distribution which approximates MAC access delay, in order to prevent nodes accessing the communication medium at the same time. The distribution of MAC access delay is determined from an empirical study documented in Chapter 6. Without knowledge of the actual MAC access delay, this type of protocol is not adaptive and will not function efficiently during low channel activity and similarly when the network is highly congested, the actual time to access the communications channel could exceed the network deferral time, leading to packet collisions and unnecessary retransmissions. It was therefore decided to maintain local connectivity information to provide a dynamic picture of the one hop nodes within a node's communication range in order for the protocol to be able to make decisions adaptively, according to relative position and the density of neighbouring nodes.

Local connectivity knowledge enables a node to make forwarding decisions implicitly based on knowledge of neighbour node positions and position of the transmission source in relation to its own position. This means that timeliness of delivery is more reliable through allowing nodes to retransmit (according to their relative position) without delay, potentially leading to a faster geographical coverage time. For those nodes which defer retransmission, their deferral time will be calculated based on actual channel activity which leads to fewer packet collisions and hence an economy in messaging. Moreover, knowledge of local connectivity allows a node to detect network partitions which enable it to react accordingly until network connectivity changes and dissemination can resume.

Local zone connectivity is achieved through the exchange of beacon messages with one hop neighbours. Additionally, a relative position classification algorithm is implemented which classifies neighbour nodes into sub areas in order to increase the efficiency of the implicit decision making employed in the DDF protocol. The local zone connectivity is a supporting mechanism within the DDF protocol which ensures that the forwarding, partition handling and suppression mechanisms operate efficiently and is therefore considered to be a shared overhead. Although the exchange of beacon messages increases the protocols overheads, it is considered to be a necessary compromise in achieving economy of messaging (in terms of message retransmissions) and ensuring reliability and adaptability to different vehicular traffic conditions.

Forwarding

The fundamental forwarding concept of distance deferral is to achieve maximum coverage by allowing a node or selection of nodes farthest away within range of the transmitting node to retransmit the message first. In order to overcome the unreliable nature of the 802.11 MAC scheme in broadcast mode, discussed in §2.3.3, basic distance deferral schemes

implement reliability at the network layer. This is achieved by allowing a subset of nodes to retransmit a message according to their distance away from the transmitting node, if after a deferral time they have not overheard the message being forwarded. The selection of the transmitting node(s) vary between schemes from selection being explicitly defined by the transmitting node given neighbourhood knowledge [10], to decisions being made without local neighbour knowledge based on distance away from the transmitting node. The balance between reliability, economy of messaging and timeliness of delivery ensuring adaptability to varying vehicular and data congestion has not been fully addressed in the literature.

In the proposed DDF protocol, the concept of a forwarding chain is used in this research to represent the message forwarding process which extends from from the source of the transmission as it propagates outwards towards the boundaries of the dissemination region. In theory, each link within the forwarding chain is formed by a forwarding node, which is determined implicitly based on local zone connectivity information. Intermediate nodes located in the region between the previous and the current forwarding nodes are responsible for ensuring forwarding chain continuity by retransmitting the message after a deferral time, if they have not overheard the forwarding chain propagating successfully. Each transmission also acts as an acknowledgement to the nodes in the previous link and essentially forms an acknowledgement chain which extends from the current forwarding node back to the source of the transmission. The concept of the forwarding and acknowledgment chain is shown in Figure 4.3.

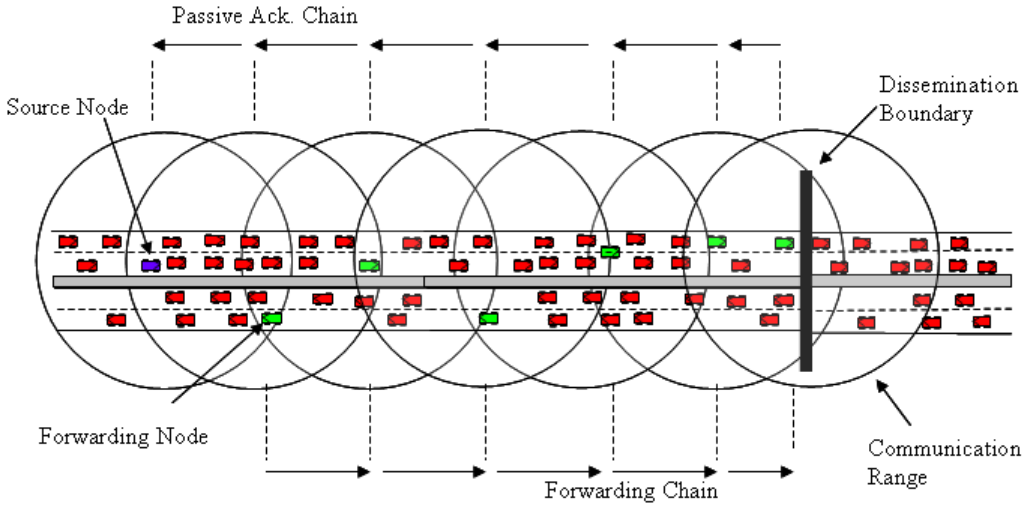


Figure 4.3: Forwarding and acknowledgement chain

In comparison to similar schemes (e.g. [55, 56]) where the forwarding nodes delay retransmission, the research in this thesis allows the forwarding node to retransmit without delay. This aims to minimise any delay and hence expedite message delivery within the

dissemination area. Any delay which the forwarding node may encounter in accessing the communications channel is taken into account by the intermediate nodes in the calculation of the retransmission deferral time so as to avoid intermediate nodes gaining access to the channel prior to the forwarding nodes.

The DDF retransmission deferral timing aims to control channel contention at the network layer in order to provide a strict distance-ordered retransmission timing in order to avoid both the hidden terminal and broadcast storms problems, while at the same time avoiding excessive delays at the MAC layer. This is achieved by optimising the conventional distance deferral technique so that it is able to adapt the retransmission delay according to both local node density and offered traffic. The aim of implementing adaptive deferral timing is to provide economy of messaging and increased reliability through minimising the likelihood of nodes accessing the communication channel simultaneously, which will lead to coverage delays and unnecessary retransmissions. Moreover, in the event of high channel activity, adapting the deferral time to allow intermediate nodes sufficient time to overhear communication from forwarding nodes reduces unnecessary retransmissions in comparison to schemes that are not truly adaptive to local conditions.

Although intermediate nodes provide a reliability mechanism which ensures that the forwarding chain continues successfully, this is done at the expense of increased protocol overheads. Some schemes balance reliability over economy of messaging and allow intermediate nodes to retransmit until they overhear the forwarding chain progressing. Packet collisions could occur around the intermediate node at the time the forwarding node retransmits preventing it from overhearing the forwarding chain progress successfully. In this situation the intermediate node will retransmit the message repeatedly unless a mechanism is included which detects and prevents unnecessary retransmitting nodes. Conversely, other schemes prefer economy of messaging and restrict the number of times intermediate nodes are allowed to retransmit the same message. However, such restrictions could lead to the forwarding chain collapsing during high channel activity.

In the DDF protocol intermediate nodes are allowed to retransmit the message but restrict retransmissions to a minimum. Additionally, it is ensured that the forwarding chain does not collapse by allowing forwarding nodes to retransmit on the occasion that the intermediate nodes do not successfully repair the forwarding chain. The forwarding node is only allowed to relinquish its role when it has detected that the forwarding chain has progressed successfully. In order to prevent potential erroneous retransmitting nodes, a suppression mechanism is implemented as outlined below. Therefore, the aim is to provide a balance between reliability and economy of messaging through additional protocol complexity, not as a messaging overhead.

Similar schemes (e.g. [54, 55, 45, 62, 57]) allow nodes to retransmit the message without taking into account previous reception and forwarding history. Essentially a node will act upon a received message regardless of whether it previously received the message. Although this method ensures that the message is forwarded within the dissemination area, it may be at the expense of a greater transmission overhead. Nodes receiving a message that they have previously seen and for which the forwarding chain has progressed successfully beyond their current location, will forward the same message again, if a node further back along the chain retransmits the message because local packet collisions prevented it from overhearing the forwarding chain progressing successfully. Conversely, not allowing a node to retransmit the message, again, could lead to the forwarding chain collapsing. This could be the case if such a node overtakes the forwarding chain and is required to act as a forwarding node. Again a balance between reliability and economy of messaging is required. Potential retransmission overhead, as mentioned above, is avoided, and reliability is ensured in the DDF protocol by allowing a node having previously seen a message to make an informed decision as to wether it should act as a forwarding node again. This is achieved through maintaining information on previously received messages and position history of the forwarding chain for the lifetime of a message. The lifetime of a message refers to the time that a threat or hazard remains relevant, and is determined by the message originator.

Node Suppression

As mentioned above, local packet collisions can prevent nodes from overhearing the forwarding chain progressing successfully and as a consequence retransmit the message unnecessarily. This research aims to improve protocol efficiency by incorporating suppression mechanisms into the DDF protocol. There are essentially two classes of suppression techniques used in the DDF protocol. The first class includes functionality which occurs when a node is in an active state in relation to the forwarding of a message. The second class includes functionality which suppresses erroneously retransmitting nodes when the forwarding chain has passed beyond the position of the retransmitting node, using an algorithm which detects, tracks and actively suppresses nodes deemed to be erroneous.

To the best of our knowledge, no similar protocol employing distance deferral techniques in the vehicular environment has implemented the two active suppression techniques employed in the DDF protocol.

Partition Handling

Partitions in network connectivity can occur within the data dissemination area, particularly in the case of free flowing, sparsely connected vehicular traffic. When a partition is encountered by a forwarding node no further dissemination of the message can be made beyond this point until more nodes enter the communication range of the forwarding node.

In order for protocols to operate in the vehicular environment over sparsely connected dissemination areas, various techniques have been employed in the literature which overcome network partitions. Some schemes which do not maintain neighbourhood information overcome partitions by allowing the forwarding node to periodically broadcast the message. When a node eventually moves into communication range it will continue the forwarding process and hence overcome the partition. Other schemes which maintain neighbourhood information use a store and foreword technique whereby the forwarding node, having detected that there are no neighbouring nodes within communication range, will store the message whilst moving towards the boundary of the dissemination area. When the node detects that it has neighbours within its communication range it will rebroadcast the message, thus overcoming the partition.

This research overcomes partitions using the store and forward mechanism which is similar to [107]. However, the partition handling mechanism employed in the DDF protocol differs in a number of ways: Firstly, the selection of the forwarding node at the head of the partition is not limited to nodes travelling in the direction of the dissemination boundary only. This is because the largest connected chain of vehicles capable of broadcasting a message in the shortest possible time could be composed of those moving away from the dissemination boundary. Secondly, the DDF protocol doesn't just wait for a new neighbour to enter the communication range of the node at the head of a partition, it uses local zone connectivity information in conjunction with the vehicle position class data to determine if a current neighbour node has become better placed to take over the forwarding role.

Under high network activity conditions a forwarding node, which has neighbouring nodes in the required location, could be prevented from continuing the forwarding chain successfully. Under such conditions the DDF protocol only allows the forwarding node to transmit a specific number of times. Thereafter, a "soft-partition" has been detected and in order to avoid adding to the local channel activity, the node temporarily stores the message until a change in network connectivity is detected. Although this scenario is not anticipated to occur frequently, it does however ensure forwarding persistence by reacting to local conditions and not compounding them.

4.4 DDF Protocol Description

The following section provides a detailed description of the functionality of the DDF protocol in order to meet application requirements discussed in §4.3 and originally introduced in Chapter 2.

4.4.1 Notation

Prior to describing the model of the DDF protocol both the notation and the terminology utilised in the formal description need to be defined first. There are two types of *spaces* which need to be defined in order to model the operation of the DDF protocol; a Euclidean space and a Discrete (enumerable) space of node identities (ID). This necessitates that a function is defined in order to map node IDs into positions in the Euclidean space and an inverse function to map ranges of positions into node IDs. There is a need to define separate spaces since objects are defined that are areas belonging to the continuum space (Euclidean plane) such as the *DDA*, *LZ* and *FZ* (introduced later in this section) and then objects/attributes are also defined of nodes positioned in these areas, in discrete space.

Space on the Euclidean plane by is denoted by Ψ. It is assumed that each node knows its own physical position within Ψ which allows a finite number of points denoted as P_i to be defined, where $i = 1, 2 \ldots, n$, such that $P_i \in \Psi$. Given two arbitrary points P_u and P_v such that $(P_u, P_v) \in \Psi$, then $[P_u, P_v]$ denotes the *line segment* connecting the two points, and $\parallel P_u P_v \parallel$ denotes the *Euclidean distance* between P_u and P_v. Two nodes are able to communicate with each other if $\parallel P_{x_u} P_{x_v} \parallel \leq R$, where R is the maximum transmission range of a node (i.e. the unit disc graph (UDG) model is adopted). In reality R is a stochastic quantity characterised by a PDF that can be derived from propagation modelling and physical layer parameters such as modulation and coding schemes. However, the probabilistic nature of the DDF protocol does not assume that a well-defined coverage radius exists, can cope with uncertainty in R and is able to function efficiently without knowledge of the probability distribution of R, $P(R)$. A further finite set of node labels is defined as x_i, such that x_i belongs to the set of all node labels N, where $N = \{x_i | x_i \in N, i = 1, 2...n\}$ in discrete space.

In order to describe relationships between objects in the two different spaces a mapping function is defined which maps the physical location of a node in Ψ to its address in the discrete space given by equation (4.1). Conversely an inverse function is defined which maps the address of the node x_i at position P_{x_i} which is given by equation (4.2). Time has been omitted from equations (4.1) and (4.2) since this research is only interested in the state of the network or a node at a specific time instant t. This now provides the

basis from which both relationships and memberships between sub-spaces and nodes can be defined.

$$f_p : x_i \longrightarrow P_{x_i} \tag{4.1}$$

$$f_p^{-1} : P_{x_i} \longrightarrow x_i \tag{4.2}$$

The first sub-space within Ψ that needs to be defined is the Data Dissemination Area (DDA). This is the area over which a packet is disseminated such that $DDA \subseteq \Psi$, defined by some application running in some node (mobile or static) and included in packet headers. The node which defines the DDA is referred to as the source, or originator, node and is denoted by S. An additional sub-space is then defined within Ψ called the Local Zone, (LZ) where $LZ \subset \Psi$, which is defined as the area covered by the radio transmission coverage of a node x_v. The LZ_{x_v} of node x_v is defined by equation (4.3), where P_{x_i} denotes a general position vector in Ψ. The local zone is defined as the area covering the one-hop neighbours of a node. The LZ is a notional area and, depending on local propagation conditions, this representation may not yield the correct LZ_{x_v} neighbourhood membership, since in reality R is not a constant. However, for the purposes of the formal notation, uniform coverage within R is assumed.

$$LZ_{x_v} = \{P_{x_i} | \parallel P_{x_v} P_{x_i} \parallel < R\} \tag{4.3}$$

Given the above definition of a node's LZ it is now possible to map the set of 1-hop neighbouring nodes of x_v denoted by LN_{x_v} using the mapping function defined in equation (4.2).

$$LN_{x_v} = f_p^{-1}(LZ_{x_v}) \tag{4.4}$$

The final region within Ψ that needs to be defined is the forwarding area of a general node, x_i, which is denoted by FZ_{x_i}. The forwarding area is a sub-set of the LZ_{x_i}, defined dynamically and existing strictly on the condition that $LZ_{x_i} \subset DDA$. This is in contrast to the property of the local zone LZ_{x_i} which can exist anywhere within Ψ. For a particular node x_v, the location of FZ_{x_v} within LZ_{x_v} is determined by $\overrightarrow{F_{dir}} = \overrightarrow{P_S P_{F_b}}$, which is the vector between the position of the source node S denoted by P_S at the time the DDA was defined, and the edge of the DDA referred to as the forwarding boundary. The forwarding boundary is denoted by F_b and the forwarding direction, $\overrightarrow{F_{dir}}$, is defined by the rule which restricts message propagation to the direction from P_S towards F_b. Given $\overrightarrow{F_{dir}}$, FZ_{x_v} is then defined as the area in LZ_{x_v} where all points P_i within LZ_{x_v} are closer to P_{F_b} than

the nodes current position P_{x_v}. Thus FZ_{x_v} can be described by the property defined in equation (4.5). Finally, given the set of node positions FZ_{x_v}, it is now possible to map them to their identities in discrete space, denoted by FN_{x_v}, using the mapping function in equation (4.6).

$$FZ_{x_v} = LZ_{x_v} \cap \{P_{x_i} | (\overrightarrow{P_i} - \overrightarrow{P_{x_v}}) \cdot \overrightarrow{P_S P_{F_b}} > 0\} \tag{4.5}$$

$$FN_{x_v} = f_p^{-1}(FZ_{x_v}) \tag{4.6}$$

Subsets of a node's forwarding area can be further defined if restrictions are applied to those nodes that are allowed to forward messages, e.g. limiting those nodes that are allowed to retransmit to the flow of vehicles heading away from S towards F_b. Such concepts are discussed in more detail in the following section.

4.4.2 Storage Mechanisms

Various dynamic data storage mechanisms are used by the DDF protocol in order to enable functionality such as the discovery of local connectivity information, forwarding, partition handling and node suppression. The following section provides a brief overview of the various data structures, lists and tables maintained by the DDF protocol.

The Forwarding Table

The forwarding table (ForTable) is defined in each equipped node in the network and is used to store information about data dissemination messages (M_{DDF}) *each time a node participates in the forwarding process.* The goal of this information is to provide each node with the capability to retransmit a message if it has not detected that M_{DDF} is being forwarded in the required direction by another eligible node during a retransmission deferral period. Each sForTable is a data structure within the forwarding table containing a copy of M_{DDF} originated or received (indexed by source node and sequence number), retransmission deferral timer interrupt, the forwarding direction (FZ_v), the forwarding node status and position data at the time M_{DDF} was received . The sForTable structure also contains a pointer to a deferral cache and on the occasion that a node is the farthest forward and no neighbouring nodes are detected in the forwarding direction, a link to the neighbour waiting queue, which is described below. Each sForTable remains in the forwarding table for the period of time it takes for the forwarding requirements of the stored M_{DDF} to have been satisfied.

The Deferral Cache

The deferral cache (DefCache) is created by the same process which creates the sForTable structure since each entry in the forwarding table contains a pointer to its own deferral cache. The aim of the deferral cache is to store the details of all M_{DDF} packets it overhears, which match both the originator address and sequence number of the corresponding entry in the forwarding table. Each time a node overhears a packet which meets the above mentioned criteria in sForTable, a structure sDefCache is entered into the DefCache containing details about the received packet.

When the deferral timer maintained in sForTable expires, the linked DefCache is analysed to determine if the M_{DDF} has been forwarded in the required direction. As in the case of the sForTable structure, the memory allocated to the deferral cache is deallocated once the forwarding requirements have been met for the corresponding entry in the ForTable.

The Message Seen List

The message seen list (MsgSeenList) contains information about all DDF messages it has received, indexed by originator and sequence number. Each time the forwarding requirements of an M_{DDF} packet have been realised at a node, its details are then entered into a sMsgSeen structure which is inserted into the message seen list. The sMsgSeen structure contains data collected from M_{DDF} such as the packet TTL, originator address ID, sequence number, packet creation time and the position of the receiving node at the time of receiving M_{DDF}. The data stored in the message seen table is used to prevent a node forwarding a message that it has previously received and for which the forwarding rules have been met. However, there are exceptions to this rule which are discussed further in §4.4.5. For every M_{DDF} that a node receives for which there is a matching entry in the message seen list, if the position of the transmitting node is closer to the forwarding boundary than the position stored in the message seen list then the position data and transmitting node address is updated. Maintenance of position history in the Message Seen list enables erroneous forwarding nodes to be suppressed as described in detail in §4.4.7. An entry remains in the Message Seen list for as long as the TTL of a packet remains valid. The sMsgSeen structure also contains pointer to a suppression list, which is described below.

The Suppression List

The suppression list (SuppList) is created and linked to an entry in the message seen list when an erroneous forwarding node is detected to be disseminating a matching originator and sequence number pair stored in the message seen table. The sForSupp structure is populated with data containing information about the erroneous transmitting node along with temporal information and the number of times this node has retransmitted

the packet in error. The sForSupp is indexed by transmitting source and time. The suppression list contains an entry for each erroneous transmission source disseminating the same message. An entry remains in the suppression list until a control packet is sent to stop the erroneously retransmitting node, or the TTL of the entry stored in the Message Seen List expires. In this case the linked Suppression List entry is deallocated.

The Neighbour Waiting Queue

On the occasion that a forwarding node detects that there are no neighbouring nodes in the required forwarding direction it will enter the details of M_{DDF} into the Neighbour Waiting Queue (NbrWaitQ). The Neighbour Waiting Queue is a FIFO (first-in-first-out) queue. The details of each packet waiting to be forwarded are stored in the sNbrWait structure, along with information about the forwarding zone and packet TTL. An entry remains in the Neighbour waiting Table until a neighbouring node is detected to be in the required forwarding zone, the node is no longer inside the *DDA* or the TTL of the packet has expired. The NbrWaitQ is serviced from events resulting from the local zone connectivity tracking functionality, which is detailed in §4.4.4

Local Zone Table

The Local Zone Table (LzTable) contains an entry for each neighbouring node from which it has received an M_{beacon} message. For each new neighbour node detected a sNbr structure is created which contains information on neighbour ID, current and previous position coordinates, the position class (the functionality of which is described in §4.4.4) to which the neighbour node belongs, the time the next beacon message is expected to be received and the number of missed messages.

Position Class Structure

The position class structure (sPosClass) maintains a count of the number of neighbouring nodes belonging to each position class. Each time a neighbour node changes position class or is removed from the Local zone Table the relevant position class is updated. Any changes in the PosClass will cause a NbrWaitQ service event if it contains entries and is not currently being serviced. The maintenance of this structure is discussed in detail in §4.4.4.

Packet Count List

In order to provide statistics on the dynamic nature of both local data traffic and density of vehicular traffic, each packet type received by a node is entered into sCntList structure maintained in the packet count list (PcktCntList). Entries PcktCntList are used to estimate local offered traffic intensity in the vicinity of each node, using a temporal window, in order to provide input variables for the calculation of retransmission deferral delays.

Entries are removed from the PcktCntList when they are no longer inside the temporal window.

4.4.3 Packet Types

There are three different types of packets which are used by the DDF protocol. Brief descriptions of each of these packets are given below:

M_{DDF} This packet is originated by a source node, S, in order to alert vehicles and or drivers of an incident, hazardous driving conditions, traffic congestion, etc., within a determined area which is anticipated to be affected by the event. The packet fields include a unique ID, originator address, sequence number, transmission source, time-stamp, position of the originator node, position of the forwarding boundary, position of the transmission source, time-to-live, and road identification field. This packet is broadcast within the DDA by the forwarding chain towards F_b. Each node receiving M_{DDF} that determines it has a role to play in the forwarding process, will update the transmission source, transmission position and network layer transmission time fields of M_{DDF} prior to rebroadcasting. Each rebroadcast of M_{DDF} by a node also acts as an acknowledgement to nodes positioned in the previous link of the forwarding chain, that M_{DDF} is being forwarded successfully towards F_b.

M_{beacon} The beacon packet is broadcast periodically by each node to all one-hop neighbours. The exchange of M_{beacon} between neighbouring nodes allows a vehicle to determine connectivity within its local zone and to track local connectivity changes. Moreover, relative positioning and movement can be tracked within a nodes LZ through consecutive exchange of M_{beacon}. The exchange of M_{beacon} facilitates the construction of local zone connectivity tracking and is fundamental in being able to make near-stateless forwarding role decisions implicitly, based on the LZ tracking knowledge. The information fields included in the M_{beacon} packet are, originator ID, transmission source ID, position of originator and transmission source, network layer transmission time, and road ID. The exchange of M_{beacon} can also be used by the application layer to detect hazards within LZ of a vehicle.

M_{Supp} This packet is sent by a node that has identified an erroneous forwarding node within its LZ. The detecting node addresses M_{supp} to the erroneous node and sends it as a unicast transmission. The acknowledgement process for this transmission takes place at the MAC layer. The M_{Supp} packet contains, source, sequence number, position of F_b, transmission source, transmission position and time-stamp fields.

4.4.4 Local Zone Connectivity Tracking

Local zone connectivity tracking enables a node, x_v, to construct a dynamic picture of the set of nodes, LN_{x_v}, located within its LZ_{x_v}, through the periodic exchange of M_{beacon}, with its one-hop neighbour nodes, x_i. Each node maintains a LzTable, where, for each M_{beacon} that x_v receives from x_i, it inserts sNbr_{x_i}, into its LzTable, such that $\text{Nbr}_{x_i} \in$ LzTable. sNbr_{x_i} contains data fields relating to temporal and position information which are extracted from the successive reception of M_{beacon} from x_i. The data fields maintained in sNbr_{x_i} for each x_i are shown in Table 4.1. Future references to any of the fields within sNbr_{x_i}, will be denoted as sNbr_{x_i}.*field name*. Assuming a reliable beaconing scheme, then $LN_{x_v} \equiv$ LzTable. However, the situation where $x_i \in LN_{x_v} \wedge x_i \notin$ LzTable could occur as a result of local radio channel packet collisions, or a variation in transmission range. The occurrence of the above mentioned condition, its consequences and the mechanisms to mitigate potential effects on the efficiency of the DDF protocol have been incorporated into the design of the DDF protocol.

Name	Description
ID	Address of neighbour node
T_{found}	Time NBR first detected
x	x coordinate of neighbour node
y	y coordinate of neighbour node
T_n	Time M_{beacon} received
$x\prime$	Previous x coordinate on neighbour node
$y\prime$	Previous y coordinate on neighbour node
$T_n\prime$	Time previous M_{beacon} received
$MISSES$	Number of times M_{beacon} missed
$PosClass$	Relative location of neighbour
$Count$	Number of M_{beacon} received
$T_{nbr_{update}}$	scheduled event if Next M_{beacon} not received

Table 4.1: Data fields maintained in structure sNbr_{x_i}

Recent trajectory history is maintained for each neighbour entry in the LzTable for up to a maximum of two previous beacon messages. This allows a node to derive approximate relative position, heading and velocity information in order to construct the dynamic picture of each neighbouring vehicle within its *LZ*. The maintenance of both temporal and position history also allows for movement correction between the reception of successive M_{beacon}. The ability to correct for node movement when making forwarding decisions increases the efficiency of the DDF protocol, as discussed in §4.4.5.

Beacon Exchange

Each node periodically broadcasts M_{beacon} according to the beacon transmission interval T_{beacon}, containing its own address, position, time of generation and road identifier. In

order to avoid synchronisation resulting in collisions of the transmission of M_{beacon} between neighbouring nodes [108], the transmission of M_{beacon} is jittered by 50% of the beacon interval period, T_{beacon}. On reception of M_{beacon} from node x_u, x_v will create and insert a new record called sNbr_{x_u} for node x_u into its LzTable on the condition that an entry for x_u does not already exist, (i.e. $\text{sNbr}_{x_u} \notin$ LzTable). Otherwise, if node x_v already had an entry in its LzTable for node x_u such that $\text{sNbr}_{x_u} \in$ LzTable, then the fields within sNbr_{x_u} are updated.

Each time a node inserts a new entry into its LzTable, it uses the Relative Position Class (RPC) algorithm, described below, to determine the position class to which the neighbour node belongs and stores the result in $\text{sNbr}_{x_i}.PosClass$. The corresponding position class count in sPosClass is incremented. sPosClass is maintained by each node and contains a counter for each of the six position classes listed in Table 4.2. For each successive M_{beacon} a node receives from a neighbour, it uses the RPC algorithm to constantly reassess the relative position of each neighbour node x_i and updates sNbr_{x_i} and sPosClass to provide the DDF protocol with an up to date view of its dynamically changing LZ.

A change in any of the position class counts in sPosClass can result in a queue service interrupt being generated for the NbrWaitQ. The NbrWaitQ is a queue that is maintained by the DDF protocol where data packets are stored on the occasion that a forwarding node does not have any neighbouring nodes in the required forwarding direction, (this functionality is described in detail in §4.4.5). The conditions under which the generation of a NbrWaitQ service interrupt can occur, are described as follows: If the LzTable_{x_v} of node x_v on receiving M_{beacon} from node x_u had the property LzTable = $\{\emptyset\}$ and node x_v has packets waiting to be serviced in its NbrWaitQ, an interrupt event is scheduled to initiate the servicing of the NbrWaitQ. On the other-hand, if node x_v was in the process of updating an entry for node x_u such that $\text{sNbr}_{x_u} \in$ LzTable and the position class for node x_u has now changed, it will decrement the previous position class count and increment the new position class count within sPosClass for node x_u. If node x_v also has packets waiting to be serviced in its NbrWaitQ, it schedules a NbrWaitQ service event, since node x_u may meet the forwarding requirements of any number of entries in the NbrWaitQ.

In order to reflect the dynamic nature of a node's LZ in addition to eliminating stale neighbourhood information, a node will delete an entry from its LzTable if it does not receive M_{beacon} from a neighbouring node for a period longer than $Max_{nbr\_miss} \times T_{update}$, where, T_{update} is the expected period between successive M_{beacon} and Max_{nbr_miss} is the maximum number of allowed successive missed $M_{beacons}$ from a neighbour node. Each time an entry is updated or added to the LzTable, a NbrUpdateInt event is scheduled to occur at $T_{update} = T_{current} + (T_{beacon} \times \delta_{nbr})$, where δ_{nbr} is a factor which allows for potential channel access time delays. On the occasion that a NbrUpdateInt event occurs for entry

sNbr_{x_u} such that $\text{sNbr}_{x_u} \in$ LzTable, then the value stored in $\text{sNbr}_{x_u}.miss$ is incremented. If $\text{sNbr}_{x_u}.miss > Max_{nbr\_miss}$ then the record is removed from the LzTable and the position class counter maintained in sPosClass to which the neighbour node belonged, is decremented. Otherwise, LzTable.*misses* is incremented and the next NbrUpdateInt event is scheduled according to T_{update}. It is assumed that an entry is removed from the LzTable for one of three reasons: firstly, $\parallel P_{x_u}P_{x_v} \parallel > R$; secondly local packet collisions are preventing communication between x_i and x_v over extended periods of time; thirdly, node x_i has failed.

Relative Position Classification

In order to increase the efficiency of the DDF protocol and further reduce LZ_{x_v} into sub-areas which restrict a node's forwarding decision to specific vehicular traffic flow, each x_i of a node x_v is classified according to its relative position and direction using a Relative Position Classification (RPC) algorithm. The RPC algorithm classifies each x_i according to its heading and relative position to x_v and assigns one of six position classes listed in Table 4.2.

Class Label	Class Description
INF	x_i infront
SDINF	x_i infront moving in same direction
OPDINF	x_i infront moving in opposite direction
BHND	x_i behind
SDBHND	x_i behind moving in same direction
OPDBHND	x_i behind moving in opposite direction

Table 4.2: Position class labels applied to neighbour vehicle, x_i, by the RPC algorithm relative to P_{x_v}

The RPC algorithm classifies relative position and relative motion between vehicles on a motorway using position vectors and their recent time evolution history. Figure 4.4 illustrates how current and past (primed) position vectors for a vehicle x_v and its neighbouring vehicle x_u are depicted for the position class INF. Given the position of x_u relative to vehicle x_v and its direction of motion, the angle α between the trajectory of x_v and x_u falls within the range $-\frac{\pi}{2} < \alpha < \frac{\pi}{2}$ this implies that $cos\alpha > 0$ and vehicle x_u is therefore classed by the RPC algorithm according to the conditions defined in equation (4.7a).

Taking into account the use of position vectors in determining the position class for the scenario depicted in Figure 4.4, it is possible to build upon these basic concepts in order to construct the conditions which define each relative position class listed in Table 4.2. The prime symbol is used to denote a past position. Essentially, vector difference terms containing $\overrightarrow{P}_x - \overrightarrow{P}'_x$ with the same subscript correspond, or are proportional to, the direction of motion, or velocity. Whereas vector terms that are both primed or both

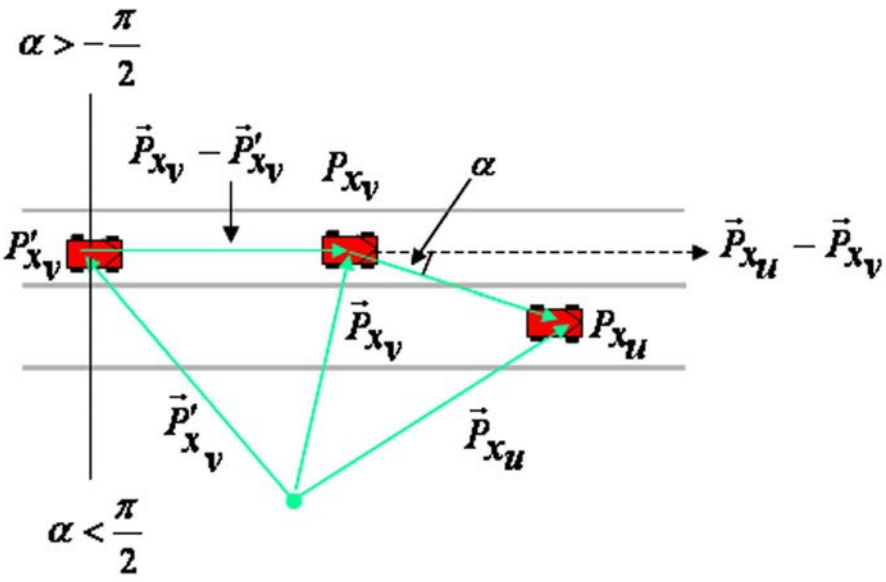


Figure 4.4: Vehicle position vectors

unprimed with different subscripts correspond to relative position vectors of other vehicles. The conditional terms which govern the assignment of relative position class are listed in equations (4.7a) to (4.7f). As can be seen from equation (4.7d) for position class OPDINF the first term is a Boolean expression that holds if the direction of motion for vehicle x_v and x_u are pointing away from each other and the second term says that the relative position is pointing towards the direction of vehicle x_v. This translates to vehicle x_u moving in the opposite direction to, and being positioned ahead of, vehicle x_v, given its current heading.

$$INF = (\overrightarrow{P_{x_u}} - \overrightarrow{P_{x_v}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) > 0 \tag{4.7a}$$

$$BHND = (\overrightarrow{P_{x_u}} - \overrightarrow{P_{x_v}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) < 0 \tag{4.7b}$$

$$SDINF = (\overrightarrow{P_{x_u}} - \overrightarrow{P'_{x_u}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) > 0 \wedge (\overrightarrow{P_{x_u}} - \overrightarrow{P_{x_v}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) > 0 \tag{4.7c}$$

$$OPDINF = (\overrightarrow{P_{x_u}} - \overrightarrow{P'_{x_u}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) < 0 \wedge (\overrightarrow{P_{x_u}} - \overrightarrow{P_{x_v}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) > 0 \tag{4.7d}$$

$$SDBHND = (\overrightarrow{P_{x_u}} - \overrightarrow{P'_{x_u}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) > 0 \wedge (\overrightarrow{P_{x_u}} - \overrightarrow{P_{x_v}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) > 0 \tag{4.7e}$$

$$OPDBHND = (\overrightarrow{P_{x_u}} - \overrightarrow{P'_{x_u}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) < 0 \wedge (\overrightarrow{P_{x_u}} - \overrightarrow{P_{x_v}}) \cdot (\overrightarrow{P_{x_v}} - \overrightarrow{P'_{x_v}}) < 0 \tag{4.7f}$$

Provided that T_{beacon} and R are such that all $\overrightarrow{\delta P}$ in equation (4.7) are (much) smaller than the radius of curvature of a motorway bend, equations (4.7) are reasonably accurate for non-straight roads.

In the instance when node, x_v, adds a new entry to its LzTable the assignment of a relative position class for a newly detected neighbour node, x_{x_i}, is restricted to classes

INF or BHND since position history, $\overrightarrow{P_{x_i}} - \overrightarrow{P'_{x_i}}$, does not yet exist.

The resulting position classification from the application of equations (4.7a) to (4.7f) to the vehicles shown in the example of Figure 4.5, relative to vehicle x_7, is shown in the Table of Figure 4.5. Vehicle x_{11} has just entered the transmission radius of x_7 and can only be classed according to its relative position to x_7, which is BHND. Only when position history exists and the node has moved since the last M_{beacon} was received, can the newly detected node be classed according to both relative motion and position.

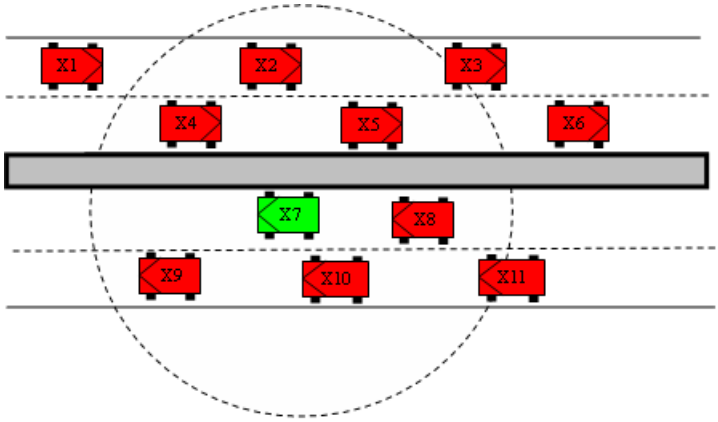
On the occasion that node x_v detects that a previously mobile neighbouring node, x_u, has not moved between successive M_{beacon}, then relative motion information is frozen. Position information is only updated for each successive M_{beacon} received from x_u until x_v detects that it is moving again.

Vehicles that are found to be parallel to x_v such that $\alpha = \frac{\pi}{2} \wedge \alpha = -\frac{\pi}{2}$ are assigned as NC which means "not classed". These vehicles will not be included as potential forwarding candidates in any of the decisions made by the DDF protocol. Such classification is transitory and depends on the precise timing of the received M_{beacon}. Position history is frozen in this state to allow relative positioning to resume as soon as this condition has lapsed. A more accurate technique for a real-world implementation on detecting vehicles which fall into this category could be employed by using an angular capture range related to the length of the vehicles. However, given the potential inaccuracies of position information the proposed technique is fit for purpose.

In this work vehicles are classified on the main carriageways of motorways only, junctions or intersections are not considered at this stage. The current classification technique will operate on roads that have bends up to a maximum $\alpha \pm 90°$. Since motorways do not normally bend that sharply this is considered to be adequate for the purposes of research in this thesis. In order to consider more complicated road layouts, classification would need to be implemented using fine grain sectorisation techniques coupled with road ID information (included in the header field of M_{beacon}) and an onboard mapping system. However, the classifications used in this work are considered to be adequate enough in order to demonstrate the concepts of the DDF protocol and the chosen application environment.

4.4.5 Message Forwarding Mechanisms

As previously explained warning messages are disseminated within a region (or regions) defined by the originating node S. The forwarding mechanism encompasses all functionality which aids the propagation of the warning message by the DDF protocol towards the forwarding boundary of the *DDA* defined by S in M_{DDF}. The DDF protocol ensures that



Neighbour Address	Position Class
X_2	OPDINF
X_3	OPDBHND
X_4	OPPDINF
X_5	OPDBHND
X_8	SDBHND
X_9	SDINF
X_{10}	SDBHND
X_{11}	BHND

Figure 4.5: Vehicle location and classification relative to x_7

data is disseminated in a reliable and speedy manner through forwarding decisions made implicitly by each node receiving the data packet. Depending on the status of the receiving node, decisions are taken as to whether the node will contribute to the forwarding process and in what way. The following section details cases where the current status of the receiving node is such that it may contribute towards the forwarding of the M_{DDF}. If the status of the receiving node prevents it from operating the forwarding mechanism, it will proceed to determine whether it is required to run either the suppression or partition handling mechanisms described in §4.4.7 and 4.4.6, respectively.

Message Received For the First Time

This section covers the actions and mechanisms that a node will implement on receiving a DDF message for the first time, given its current location in relation to the transmitting node, the dynamic nature of its LZ, and the relative location of both the forwarding boundary and the location of S. The following actions are summarised in the workflow diagram shown in Figure 4.6 and described below in detail.

A warning message is only of interest to a node positioned within the geographical area addressed by the source node. Therefore, on reception of M_{DDF} from a transmitting node x_u, node x_v will proceed to determine if its current position is located within the DDA

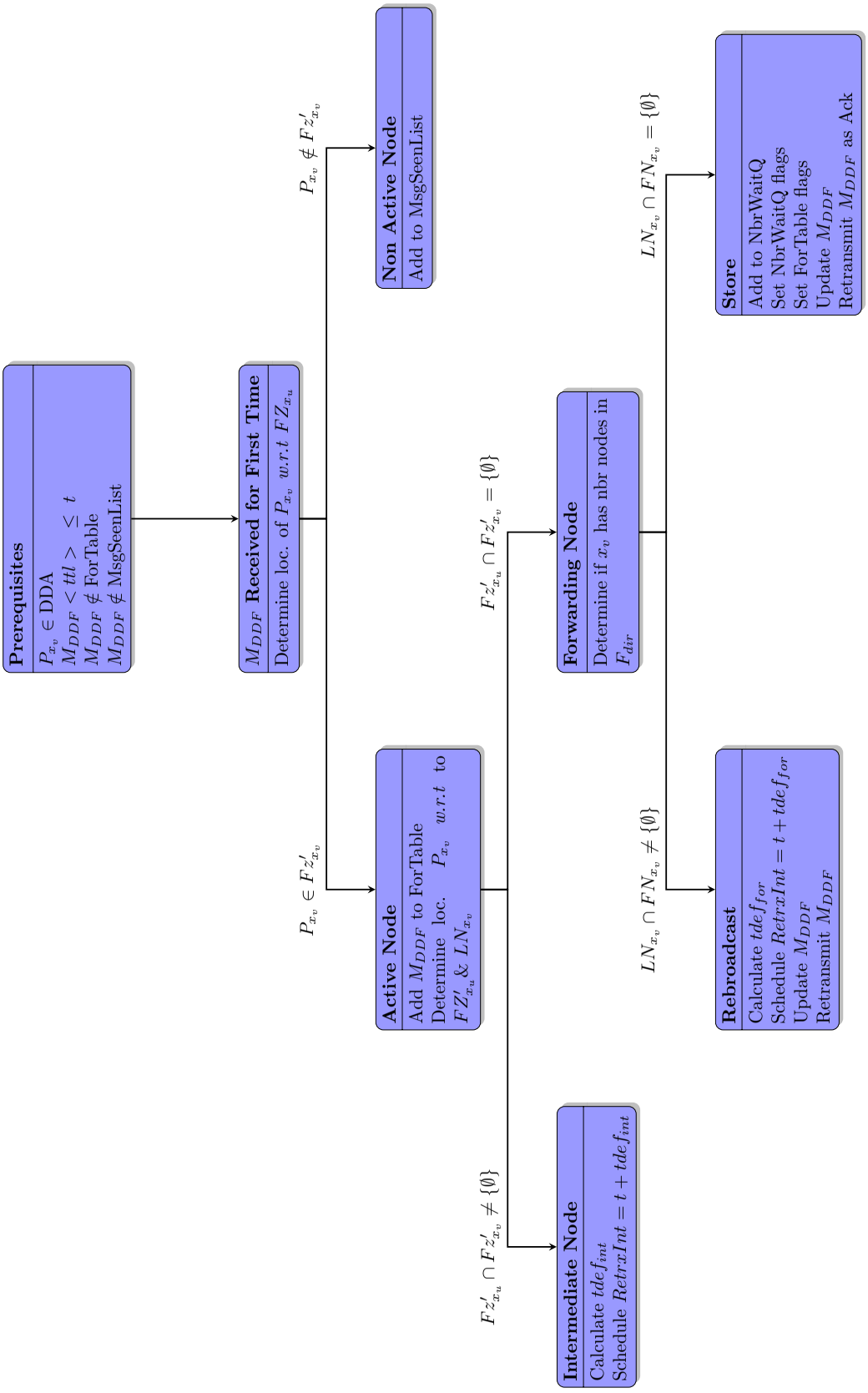


Figure 4.6: Workflow diagram of forwarding actions when message is received for the first time

specified by S, from the data contained in the header fields of the received message. If node $P_{x_v} \in$ DDA, then it will continue to process the packet further. Otherwise, if $P_{x_v} \notin$ DDA node x_v will drop the packet and discontinue further processing of M_{DDF}.

Node x_v will proceed to execute the forwarding mechanism if it determines that it has not previously received or is currently in the process of servicing a copy of M_{DDF}. Having established that M_{DDF} is received for the first time, since $M_{DDF} \notin \text{ForTable}_{x_v} \wedge M_{DDF} \notin \text{MsgSeenList}_{x_v}$, node x_v proceeds to determine its role in the forwarding of M_{DDF}. The forwarding role is determined implicitly based on the relative position of node x_v with respect to x_u, S and F_b and the membership of its LzTable positioned within the FZ of the transmitting node, x_u.

In order to ensure M_{DDF} makes progress towards F_b with respect to P_{x_u}, only those nodes positioned closer to P_{F_b} than P_{x_u} such that, $P_{x_i} \in FZ_{x_u}$, proceed to determine if they have an active role to play in the forwarding of M_{DDF}. Therefore, node x_v proceeds to check conformity with this rule by firstly determining whether it is located within the forwarding zone of node x_u which is denoted as FZ'_{x_u}. FZ'_{x_u} is determined using equation (4.5), where $\overrightarrow{P_S P_{F_b}}$ is determined from fields in M_{DDF}, and $\overrightarrow{Px_u}$ is determined from the metadata maintained in the LzTable for node x_u.

If node x_v determines that it is positioned such that $P_{x_v} \notin FZ'_{x_u}$, it has previously failed to receive M_{DDF} and therefore does not have an active role to play in the forwarding process. This situation could have resulted as a direct consequence of local packet collisions or local variations in radio propagation conditions preventing a node from receiving the packet. Node x_v enters metadata extracted from M_{DDF} along with the current position of node x_v into its MsgSeenList.

However, if $P_{x_v} \in FZ'_{x_u}$, node x_v will have an active role in the forwarding process and will proceed to determine its relative distance within FZ'_{x_u}, in comparison to its neighbour nodes. In order to expedite delivery and provide maximum coverage for each transmission of M_{DDF} towards the F_b, the node farthest away within its FZ assumes the *forwarding* role and retransmits M_{DDF} without delay. All other nodes within FZ_{x_v} assume the role of *intermediate* node. Each node makes this decision implicitly based on its current view of the relative position and heading of neighbouring nodes maintained in its LzTable which are within the forwarding zone of the transmitting node with respect to its current position.

$$dist_{x_u x_v} = \| P_{x_u} P_{x_v} \| \tag{4.8}$$

$$dist_{x_u x^i_{nbr}} = \mid \overrightarrow{P_{x^i_{nbr}}} + \frac{\Delta t^i_{corr}}{\Delta t^i_{nbr}} \cdot (\overrightarrow{P_{x^i_{nbr}}} - \overrightarrow{P'_{x^i_{nbr}}}) - \overrightarrow{P_{x_u}} \mid \tag{4.9}$$

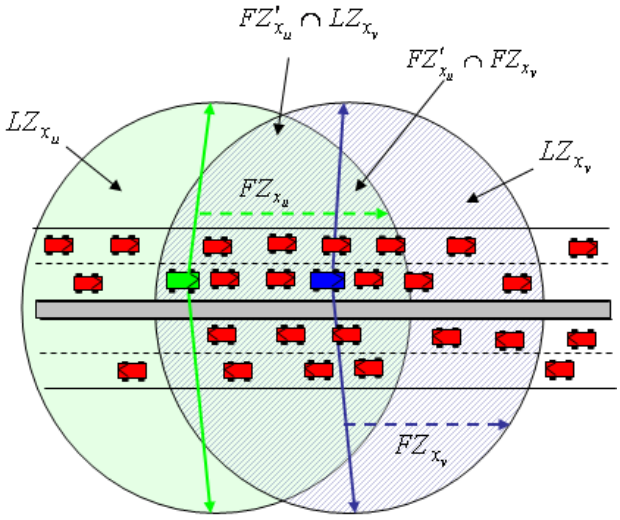
Having determined that $P_{x_v} \in FZ'_{x_u}$, node x_v proceeds to calculate $dist_{x_u x_v}$, the Euclidean distance between itself and P_{x_u} using equation (4.8), and $dist_{x_u x^i_{nbr}}$, the distance between neighbour nodes in LzTable that are within FZ_{x_u}, such that $\forall x_{nbr_i} \mid FN_{x_v} \subset LN_{x_v}$. Given that a node elects its forwarding role implicitly, based on a snapshot in time of local connectivity data stored in its LzTable, the calculation of $dist_{x_u x^i_{nbr}}$ must allow for nodal movement between successive updates $\forall x_{nbr_i} \mid FN_{x_v} \subset LN_{x_v}$ in order to reflect physical positions at a specific instance in time. Omitting to incorporate nodal mobility into the calculation of distance could lead to a node erroneously assuming the role of forwarding node, if, in the intervening time between updates neighbouring nodes move such that they are the most appropriate forwarding node. Therefore, in order to reduce positioning errors the DDF protocol corrects for vehicle movement within the window between the last update and current time using position vectors extracted from the position history data maintained in the LzTable.

The parameter $dist_{x_u x^i_{nbr}}$ is determined from equation (4.9), where $\overrightarrow{P_{x^i_{nbr}}}$ and $\overrightarrow{P'_{x^i_{nbr}}}$ are the position vectors of the last beacon update and the previous beacon update respectively for a neighbour entry stored in the LzTable. Δt^i_{nbr} is the time window between successive updates for the ith entry in the LzTable and is derived from $t_1 - t_2$ where t_1 is the last time and t_2 the previous time that an entry in the LzTable was updated. Δt^i_{corr} is the time window over which the correction factor is applied and is determined from $t_n - t_1$, where t_n is the current time. $\overrightarrow{P_{x_u}}$ is the position vector of the transmitting node obtained from M_{DDF}.

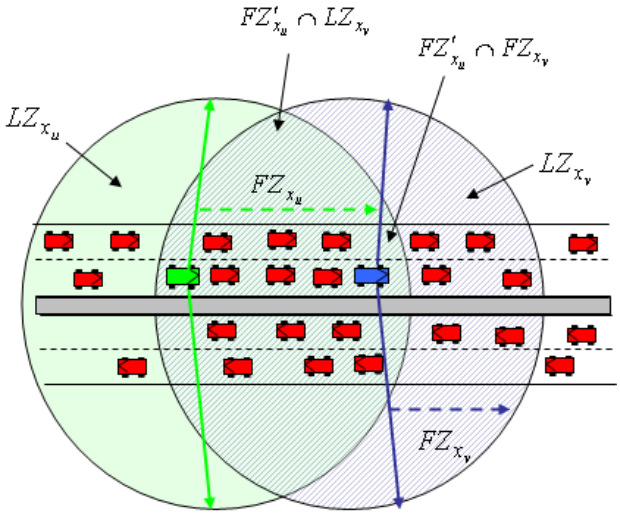
If node x_v determines that it is positioned farthest away within FZ'_u such that $FN'_{x_u} \cap FN'_{x_v} = \{\emptyset\}$ then it will assume the role of the *forwarding* node, as shown pictorially in Figure 4.7(b). Otherwise, if $FN'_{x_u} \cap FN'_{x_v} \neq \{\emptyset\}$ then node x_v will assume the role of *intermediate* node as shown in Figure 4.7(a). Node x_v inserts a copy of M_{DDF} into ForTable prior to executing the functionality associated with the role of either forwarding or intermediate node, as detailed below.

Forwarding Node

The role of the forwarding node is to establish the forward/acknowledgement chain, which extends from the source node to F_b, by forwarding M_{DDF} straight away, without deferring retransmission. The forwarding node does not relinquish its role until it has established that the forwarding chain is continuing successfully. This is achieved by allowing a forwarding node to retransmit M_{DDF} if it has not overheard a retransmission from any of



(a) x_v assumes *intermediate* role



(b) x_v assumes forwarding role

Figure 4.7: In Figure (a) $FZ'_{x_u} \cap FZ_{x_v} \neq \{\emptyset\} \longrightarrow x_v =$ *intermediate* node. Whereas in Figure (b) $FZ'_{x_u} \cap FZ_{x_v} = \{\emptyset\} \longrightarrow x_v =$ *forwarding* node.

the nodes within its FZ. The forwarding node is effectively at the head of the required coverage area in the DDA as it progresses towards F_b. In the event that the forwarding node detects a partition, which means that no further forwarding progress can be made towards F_b, then the role of the forwarding node is to store M_{DDF} until another node is detected closer to F_b.

As a forwarding node, x_v proceeds to determine from its LzTable if $P_{x_{nbr_i}} \in FZ_{x_v}$. If

node x_v determines that $LN_{x_v} \subset FN_{x_v} = \{\emptyset\}$ it has no neighbour nodes within its communication range, closer to F_b, than its current position. Node x_v, has detected that it is currently at the head of a network partition which it cannot overcome until $LN_{x_v} \subset FN_{x_v} \neq \{\emptyset\}$. Node x_v adds an entry for M_{DDF} into its NbrWaitQ and links the entry to the ForTable. The neighbour queue flag, NbrQFlag is set to provide an indication to the local connectivity tracking algorithm to start servicing the NbrWaitQ upon detection of a new node within LZ_{x_v} or on the occasion that $x_{nbr_i} \in LZ_{x_v}$ has changed position class. The ForTable.*nbrwaitflag* field is set to indicate that x_v is in the 'waiting for neighbour' state for M_{DDF}. Finally, node x_v updates the position, ID and time fields of M_{DDF} and rebroadcasts it in order to satisfy the requirements of the passive acknowledgement chain.

In the case that $LN_{x_v} \subset FN_{x_v} \neq \emptyset$, node x_v has neighbour nodes within its FZ_{x_v} and can therefore broadcast the message and expect to receive a passive acknowledgement when the next forwarding node, within the forwarding chain towards F_b, broadcasts M_{DDF}. Node x_v calculates a retransmission deferral time t_{def}^{for} using equation (4.25) and schedules a retransmission interrupt event to occur at $T_{current} + t_{def}^{for}$. Prior to retransmitting the message, Node x_v updates position, ID and time fields within M_{DDF} and rebroadcasts M_{DDF} to further progress the forward and passive/acknowledgment chain towards F_b.

Intermediate Node

The role of the intermediate node ensures that the forward/acknowledgement chain continues to progress towards F_b on the occasion that it has not overheard M_{DDF} being forwarded prior to its deferral timer expiring. If the deferral time has elapsed and the above requirement has not been met, then the intermediate node rebroadcasts the packet. The timing of the deferral event for the intermediate node is critical to ensuring that the DDF protocol provides efficient, reliable and speedy delivery to the *DDA* addressed by the source node. Rather than waiting for the forwarding node to retransmit the message, the intermediate nodes are allowed to retransmit since they are in a position to make greater forwarding progress towards F_b. However, to eliminate intermediate nodes competing for access at the MAC layer, which will introduce additional forwarding delays, the retransmission deferral time is calculated to allow nodes closer to F_b to retransmit the message first.

As an intermediate node, x_v will proceed to calculate its retransmission deferral time t_{def}^{int} using equation (4.25). The method used to calculate t_{def}^{int} is covered in detail in §4.4.9. Node x_v completes its actions for M_{DFF} by scheduling a retransmission deferral interrupt event to occur at $T_{current} + t_{def}^{int}$.

The functionality which follows a scheduled retransmission deferral event for both the forward and intermediate node, is described later in this section.

Message Previously Received

The course of action that a receiving node, x_v, will take when it determines that is has previously received a duplicate copy of M_{DDF} from a transmitting node, x_u, depends upon whether an entry for M_{DDF} exists in either the ForTable or the MsgSeenTable. In either case, any proceeding action depends on the state of node x_v in relation to M_{DDF}, its relative location to x_u, and LZ_{x_v} connectivity. The possible states and associated actions which node x_v will take depend on whether $M_{DDF} \in (\text{ForTable}_{x_v} \wedge \text{MsgSeenTable}_{x_v})$, are summarised in the workflow diagram shown in Figure 4.8 and described in detail below.

Entry in Forward Table

If $M_{DDF} \in$ ForTable node x_v can be in either one of two states; the 'deferral' state waiting to overhear a passive acknowledgement for M_{DDF} from F_{dir} or in the 'waiting for neighbour' state at the head of a partition holding M_{DDF} in its NbrWaitQ.

- **Deferral State:** If node x_v detects that it is in the deferral state it will enter meta-data for M_{DDF} into its DefCache. The DefCache will be inspected upon expiration of the scheduled RetrxInt event in order to establish whether coverage requirements have been met and M_{DDF} has been forwarded towards F_b successfully. The possible actions which node x_v will take when the RetrxInt occurs are discussed later in this section.

- **Waiting For Neighbour:** If ForTable.*NbrWaitFlag* is set, then node x_v is at the head of the forwarding chain and is currently storing the message until local Zone connectivity tracking detects a new node within FZ_{x_v}, such that $FN_{x_v} \neq \{\emptyset\}$, before it can further progress M_{DDF} towards F_b. At the time node x_v, made the decision to insert M_{DDF} into NbrWaitQ, which is denoted as t', it had determined from its LzTable_{x_v} that $FN_{x_v} = \{\emptyset\}$. Node x_v could quite simply drop the received packet without any further processing at this stage, in the knowledge that it is at the head of a partition and the forwarding chain is progressing towards F_b. However, there are two cases where a forwarding node in this state will need to take action in order to prevent unnecessary congestion of the bandwidth, which are explained as follows:

  - **Case 1:** As previously mentioned in §4.4.4 it is possible that $LN_{x_v} \not\equiv$ LzTable, if node x_v elected itself as forwarding node under this condition at t', it may not be the *real* forwarding head. This condition could result under circumstances where, for example, at t' a node, x_i, positioned closer to F_b than x_v, received

Figure 4.8: Workflow diagram of actions on reception of a duplicate copy of M_{DDF}

M_{DDF} and node x_v had no knowledge of x_i such that $x_i \notin \text{LzTable}_{x_v}$. This situation could occur under the following circumstances:

* x_i had recently been removed from LzTable as a result of local collisions or a variation in communication range preventing x_v from receiving $M_{beacon_{x_i}}$.
* x_i had entered LZ_{x_v} and received M_{DDF} a t' before x_v had received $M_{beacon_{x_i}}$ and updated its LzTable.

– **Case 2:** Another condition node x_v must act upon in this state is to suppress unnecessary retransmissions of M_{DDF} from nodes that are positioned such that $P_{x_i} \notin (FZ \subset LZ)$. Retransmissions could occur from nodes x_i that failed to overhear the passive acknowledgement from x_v at t', having determined that the passive acknowledgement chain had been broken for M_{DDF}, when the $RetrxInt$ event occurs.

The above mentioned cases are mitigated against in the DDF protocol as described below:

On reception of M_{DDF}, if node x_v establishes that $Px_u \in \text{ForTable}.FZ_{x_v}$, it removes the corresponding entry for M_{DDF} from both the ForTable and the NbrWaitQ, since it has determined that it is no longer the lead forwarding node at the head of the forwarding partition, and inserts an entry for M_{DDF} into the MsgSeenTable. If the NbrWaitQ $= \{\emptyset\}$ then the NbrQFlag is reset.

Otherwise, if on reception of M_{DDF} node x_v determines that $Px_u \notin \text{ForTable}.FZ_{x_v}$, it infers that M_{DDF} has been retransmitted from a node further away from F_b than Px_v. Node x_v, updates M_{DDF} with the current time, its current position and ID and rebroadcasts the updated message to act as an passive acknowledgement to suppress any further retransmissions.

Entry in Message Seen Table

If on reception of M_{DDF} node x_v determines that $M_{DDF} \in$ MsgSeenTable, then it has previously received and completed all forwarding requirements for M_{DDF}. However, due to the dynamic nature of vehicular traffic and the fact that the DDF protocol does not restrict retransmitting nodes to one direction of traffic flow, node x_v could find itself closer to F_b than the transmitting node x_u, such that $P_{x_v} \in FZ_{x_u}$. In this case node x_v could be positioned such that it will be implicitly elected by $x_{nbr_i} \in FZ_{x_u}$ as the forwarding node. In order to ensure that the forwarding chain continues towards F_b, node x_v must therefore check its status in relation to M_{DDF}, P_{x_u} and LZ_{x_v} to determine whether it will have an active forwarding role to play.

As previously discussed in §4.4.2, the function of the MsgSeenTable is not only to provide an indication that the warning message has previously been received and sent to the applic-

ation layer, but it also used to aid the suppression of erroneous transmitting nodes. When a node enters details for M_{DDF} into its MsgSeenTable, it sets the MsgSeenTable.$PosHist$ field with the position of the node at the time it received the message. Thereafter, each time a node receives a copy of M_{DDF} it will check whether it will need to suppress an erroneous node, update MsgSeenTable.$PosHist$, or act as a forwarding node.

The actions that node x_v will take on detecting whether it will be required to suppress erroneous forwarding node, update MsgSeenTable.$PosHist$, or act as a forwarding node is discussed below.

Forwarding Node

The DDF protocol allows a node which has previously received and successfully overheard the forwarding chain progressing towards F_b to make an informed decision as to whether it should take part in the forwarding process again. This is possible since an entry is only inserted into the MsgSeenTable on completion of all forwarding requirements for M_{DDF} and through the maintenance of position history information on the last known received position of M_{DDF} closest to the F_b. This decision is based on whether a node previously overheard the message being forwarded at a position closer to F_b than the current position of the transmitting node as explained below.

If node x_v determines that it is positioned according to the condition given in equation (4.10) then it will be elected as the forwarding node for M_{DDF} within FZ_{x_u}. However, node x_v will only function as a forwarding node if the last time it received a copy of M_{DDF} it was positioned further away from P_{F_b} than the current position of the transmitting node x_u. Therefore, if node x_v proceeds to determine that the condition given in equation (4.11) is true, it will assume the role of forwarding node.

$$P_{x_v} \in FZ_{x_u} \wedge (FZ_{x_u} \cap FZ_{x_v} = \{\emptyset\}) \tag{4.10}$$

$$\| P_{x_u}P_{F_b} \| < \| \text{MsgSeenTable.}PosHistP_{F_b} \| \tag{4.11}$$

Node x_v having established that it meets the required conditions to operate as a forwarding node, proceeds to remove the entry for M_{DDF} from the MsgSeenTable and continues by processing M_{DDF}, transmitted from x_u, according to the functionality previously described for a forwarding node receiving M_{DDF} for the first time.

Updating Position History

Updating the MsgSeenTable.$PosHist$ field each time a node receives a copy of M_{DDF} for which an entry exists in the MsgSeenTable, provided the position of the transmitting node is closer to F_b than MsgSeenTable.$PosHis$, is the underlying functionality to ensuring that

the selection of the forwarding node and suppression of erroneous nodes operates correctly.

The MsgSeenTable allows informed decisions to be made since it maintains position information on the closest point a node has been within the forwarding chain towards F_b, which allows it to determine its role when it next receives a copy of M_{DDF} for which an entry exists in the MsgSeenTable. If a node receives M_{DDF} at a position closer to F_b than MsgSeenTable.$PosHist$, then it must update this field to reflect this condition. Updating the MsgSeenTable.$PosHist$ field each time a node receives a copy of M_{DDF} for which an entry exists in the MsgSeenTable, provided specific conditions are met, is imperative to ensuring that the selection of the forwarding node and suppression of erroneous nodes operates correctly. The specific conditions for updating MsgSeenTable.$PosHist$ are discussed below.

If node x_v determines it is not a forwarding node within FZ'_{x_u} such that the condition given in equation (4.12) is true and the position of Px_u is closer to F_b than the current position recorded in MsgSeenList$< PosHist >$, equation (4.11), then it will proceed to update MsgSeenList$< PosHist >$ with P_{x_u}.

$$P_{x_v} \notin FZ'_{x_u} \vee [(P_{x_v} \in FZ'_{x_u}) \wedge (FZ'_{x_u} \cap FZ_{x_v} \neq \{\emptyset\})] \tag{4.12}$$

Detection of Erroneous Node

As previously mentioned local packet collisions and/or variations in a node's transmission range could lead to forwarding loops and/or more than one forwarding chain in the scenario where the forwarding nodes passive acknowledgment requirements have not been met. The data maintained in the MsgSeenTable along with the specific conditions described below, can be used to aid the detection of a suspected erroneous retransmitting node.

If node x_v detects that the conditions given in equation (4.13) are true, then it has detected a potentially erroneous transmitting node since it has determined that its current position and position history are closer to F_b than the position of the transmitting node.

$$(P_{x_v} \in FZ'_{x_u}) \wedge (\text{MsgSeenTable}.PosHist \in FZ'_{x_u}) \tag{4.13}$$

Having detected a potentially erroneous transmitting node, x_v proceeds to execute the node suppression functionality detailed in §4.4.7.

Retransmission Forwarding Interrupt

In the intervening time between the scheduling and expiration of a RetrxInt event (deferral time), a node caches any copies of M_{DDF} it receives from x_{nbr_i} in the RetrxCache associated with the matching entry in the ForTable (an entry in the ForTable for which the RetrxInt event has occurred, is referred to as ForTable$_{R_{int}}$). This allows a node to determine from its RetrxCache if the forwarding chain has been successfully progressed towards F_b with respect to the FZ defined at the time RetrxInt was scheduled. The actions that a node, referred to as x_v, will take when the RetrxInt event occurs, are summarised in the workflow diagram shown in Figure 4.9 and described in detail below.

Since it is most likely that node x_v has moved during the deferral time, node x_v must first establish whether it is still inside the DDA from the metadata contained in the ForTable$_{R_int}$. If node x_v determines that is now outside the boundaries of the DDA, it can no longer participate in the forwarding process and will complete its actions by removing ForTable$_{R_{int}}$ from the ForTable.

However, if x_v determines that $P_{x_v} \in DDA$, it will proceed to search through the RetrxCache associated with ForTable$_{R_{int}}$ to determine if it contains any entries for which it overheard M_{DDF} being transmitted from x_{nbr_i} during the deferral time. If RetrxCache $\neq \{\emptyset\}$ and x_v determines that $\Omega \neq \{\emptyset\}$ in equation (4.14), then the coverage requirements have been successfully met. Having established that the forwarding chain has progressed successfully towards F_b, node x_v proceeds to enter details from ForTable$_{R_{int}}$ into the MsgSeenTable. MsgSeenTable.$PosHist$ is initialised with the position of RetrxCache Px_{nbr_i} closest to F_b. This enables the suppression tracking functionality to operate correctly, as discussed in §4.4.7. Finally node x_v completes its actions by removing entry ForTable$_{R_{int}}$.

$$\Omega = \forall Px_{nbr_i} \in \text{RetrxCache} \cap \text{ForTable}_{R_{int}}.FZ_{x_v} \tag{4.14}$$

Conversely, if node x_v determines that $\Omega = \{\emptyset\}$ in equation (4.14), the coverage requirements have not been met. Since node x_v has determined that it has not overheard the forwarding chain progressing, it will proceed to retransmit M_{DDF} in order to ensure that the forwarding chain continues successfully towards F_b. The actions that node x_v will proceed to take prior to retransmitting M_{DDF} vary depending on its previously assigned forwarding role. However, irrespective of whether node x_v is currently classed as a forwarding or intermediate node, it must firstly determine whether it has any neighbour nodes positioned in the FZ, based on its current position, for the entry in ForTable$_{R_{int}}$.The conditions and resulting functionality are discussed below:

Figure 4.9: Workflow diagram for retransmission interrupt event

If node x_v determines that it does not have any neighbouring nodes in the forwarding direction such that, $FN_{x_v} \cap LN_{x_v} = \{\emptyset\}$, then it is currently at the head of a partition in the network and must store M_{DDF} until it detects that $FN_{x_v} \cap LN_{x_v} \neq \{\emptyset\}$. Node x_v proceeds to enter the details of M_{DDF} into its NbrWaitQ and sets both the NbrQFlag and ForTable.$NbrWaitFlag$, if not already set. A node that previously acted as a intermediate node for M_{DDF} will now assume a forwarding role since it has detected that it is currently at the head of the forward chain closet to F_b. Node x_v completes its current actions for M_{DDF} by broadcasting a copy of M_{DDF}, which acts as an acknowledgement in order to suppress any further retransmissions from x_{nbr_i} located in the area defined by $Px_i \mid Px_i \notin FZ_{x_v} \wedge Px_i \in LZ_{x_v}$.

Otherwise, if Node x_v detects that $FN_{x_v} \cap LN_{x_v} \neq \{\emptyset\}$, then it has neighbour nodes closer to F_b than its current position, and is therefore in a position to participate as an active retransmitting node. Node x_v will increment the retransmission count maintained in ForTable.$RetrxCount$. Thereafter, the proceeding action depends on whether the state of the previously determined forwarding role status, stored in ForTable$_{R_{int}}$.$Role$, is either "forward" or "intermediate".

Forwarding node: A forwarding node does not relinquish its role until it either overhears the forwarding chain progressing successfully, or it is suppressed as an erroneous transmitting node. This is because localised network activity surrounding the nodes within the current link of a forwarding chain could prevent it from progressing beyond the current forwarding node. Therefore, a mechanism is needed which ensures both persistence and reliability under congested bandwidth conditions within the LZ of a forwarding node. Although intermediate nodes retransmit M_{DDF} up to IntRetrxMax times, as a method of providing reliability, the value of IntMaxRetrx is limited to prevent network congestion within the LZ of a node, since more than one intermediate node could potentially retransmit M_{DDF}.

The DDF protocol ensures persistence and additional reliability which ensures that the forwarding chain is unlikely to collapse by allowing the forwarding node to retransmit after a deferral time, from the time it first forwarded the message, up to ForRetrxMax times. Moreover, by the time a node has reached ForRetrxMax it would have expected to overhear the forwarding chain progressing, since the node has established that it has neighbours present within its FZ. This research defines such an occurrence as a "soft partition", whereby no transmissions are making further progress towards F_b. In order to ensure the forward chain progresses and avoids adding to the current LZ communication overhead, an extended back-off time is introduced prior to attempting the retransmission process again, by entering it into NbrWaitQ until network connectivity changes within FZ. If the forwarding chain had made further progress without being detected by a previous

forwarding node, either because it moved outside of communication range of the next forwarding node or a local collision prevented it from overhearing progress, then it would be suppressed as an erroneous node, prior to reaching ForRetrxMax, through the suppression mechanism discussed in §4.4.7. Although the "soft partition" concept proposed in this research could potentially be at the expense of introducing an extra delay to the forwarding of M_{DDF} towards F_b, it does ensure forwarding persistence and high message successful reception rates.

As a forwarding node, x_v will firstly check the number of times it has rebroadcast M_{DDF}. If node x_v determines that ForTable.$RetrxCount \leq$ ForRetrxMax, then it will calculate a retransmission deferral time, t_{def}^{for}, using equation (4.25), and schedule the next retransmission event to occur at $T_{current} + t_{def}^{for}$, check that the RetrxCache is clear and finally rebroadcast an updated copy of M_{DDF}. However, if node x_v determines that ForTable.$RetrxCount >$ ForRetrxMax, a soft partition has occurred, since the maximum allowed retransmission events been reached. Node x_v completes its current action for ForTable$_{R_{int}}$ by adding metadata for M_{DDF} to its NbrWaitQ.

Intermediate node: As an intermediate node x_v will firstly check the number of retransmissions it has previously broadcast. In the case where ForTable.$RetrxCount <$ IntRetrxMax, node x_v will schedule the next retransmission event to occur at $T_{current} + t_{def}^{for}$, check that the $RetrxCache$ is clear and finally rebroadcast an updated copy of M_{DDF}. In the case where ForTable.$RetrxCount =$ ForRetrxMax, node x_v, will rebroadcast an updated copy of M_{DDF} for the final time, remove the entry for which the interrupt occurred from the ForTable, and finally enter metadata for M_{DDF} into the MsgSeenTable. MsgSeenTable. $PosHist$ is initialised with the position of P_{x_v} since node x_v has not overheard M_{DDF} being broadcast from a position closer to P_{F_b}.

4.4.6 Overcoming Network Partitions

Partitions in network connectivity can occur within the *DDA* when forwarding a warning message from the source node towards the F_b. This situation is detected by a forwarding node, x_{for}, when it determines from LZ connectivity information that there are no neighbouring nodes closer to F_b than its current position, such that, $FN_{x_{for}} \cap LN_{x_{for}} = \{\emptyset\}$.

Inserting Entry into NbrWaitQ

In the event of a network partition, the message forwarding remains the responsibility of x_{for}, which enters details of M_{DDF} into the NbrWaitQ. Each entry in the NbrWaitQ is linked to its corresponding entry in the ForTable, allowing access to metadata for M_{DDF}.

For each instance that a node inserts an entry into NbrWaitQ, it broadcasts a copy of M_{DDF} in order to satisfy the acknowledgment requirements for the previous link in the forwarding chain. The exception to this rule is when a soft partition is detected, in this case, x_{for} will have previously retransmitted M_{DDF} up to $ForRetrxMax$ times, which will satisfy the requirements of the acknowledgement chain. If not already set, x_{for} will set the NbrQueFlag. This provides an indication to the local zone connectivity tracking program to start servicing the NbrWaitQ, if not already doing so, when there is either a change in position class or a new neighbour node is detected. The ForTable.$NbrWaitFlag$ is also set to indicate that node x_{for} is currently storing M_{DDF}.

Although the instances when an entry is inserted into the NbrWaitQ have been discussed previously in §4.4.5, a summary of these events are listed below for ease of reference:

- x_v receives M_{DDF} from x_u for the first time and determines that it is a forwarding node, $x_v \rightarrow x_{for}$, such that:
  $FZ'_{x_u} \cap FZ_{x_v} = \{\emptyset\} \wedge FN_{x_v} \subseteq LN_{x_v} = \{\emptyset\}$.
- RetrxEvent occurs for x_{for} such that:
  DefCache = $\{\emptyset\}$, ForTable.$RetrxCnt \leq MaxRetrx$ and $FN_{x_{for}} \subseteq LN_{x_{for}} = \{\emptyset\}$
- RetrxEvent occurs for x_{for} such that:
  DefCache = $\{\emptyset\}$, ForTable.$RetrxCnt > MaxRetrx$ and $FN_{x_{for}} \subseteq LN_{x_{for}} = \{\emptyset\}$
- x_v receives M_{DDF} from x_u and $M_{DDF} \in$ MsgSeenTable and determines it is a forwarding node, $x_v \rightarrow x_{for}$, such that:
  $FZ'_{x_u} \cap FZ_{x_v} = \{\emptyset\} \wedge FN_{x_v} \subseteq LN_{x_v} = \{\emptyset\}$.

Servicing of NbrWaitQ

The servicing of the NbrWaitQ is initially triggered by the local connectivity tracking mechanism when either a new entry is added to the LzTable or a change in the vehicle position class membership occurs. If the NbrQSerFlag is set then the NbrWaitQ is currently being serviced, any change in network connectivity will be taken into account during this process, and no further action is taken. However, if NbrQSerFlag is not set, its status is changed and the servicing of NbrWaitQ is triggered.

The NbrWaitQ servicing functionality cycles through the entries in NbrWaitQ until an entry is found which meets the following condition given in equation (4.15). On finding an entry which meets the above criteria the linked entry pointed to in the ForTable is accessed and the ForTable.*RetrxCnt* field incremented, the ForTable.*NbrWait* field is reset, a retransmission deferral time calculated using equation (4.25) for a "forwarding" node, the entry in the NbrWaitQ is removed and an updated copy of M_{DDF} is broadcast. Finally,

if the NbrWaitQ $\neq \{\emptyset\}$, then node x_{for} will schedule a queue service event, NbrQEv, to occur at $T_{current} + t_{QSR}$, where t_{QSR} is the queue service rate. If after servicing an entry, NbrWaitQ $= \{\emptyset\}$, then the status of both the NbrQFlag and the NbrQSerFlag are reset. However, if no entries in the NbrWaitQ can be serviced, no further queue service events are scheduled since the nodes current connectivity does not satisfy the forwarding requirements for any of the entries in NbrWaitQ and the NbrQSerFlag is reset.

$$(TTL < t) \wedge (Px_{for} \in DDA) \wedge (LN_{x_{for}} \subseteq FNx_{for} \neq \emptyset) \tag{4.15}$$

4.4.7 Node Suppression

As mentioned previously in §4.3 the DDF protocol incorporate suppression mechanisms in order to improve the efficiency of the DDF protocol. This is achieved by reducing the number of retransmissions which can occur when local collisions have prevented successful reception of the passive acknowledgement, or where a node implicity elected itself as a forwarding node based on local connectivity from LzTable, which did not reflect physical connectivity at the instant the decision was made. Additionally, in the case where a forwarding node has not overheard M_{DDF} being forwarded towards F_b, it will continue to retransmit M_{DDF} up to ForRetrxMax, enter details of M_{DDF} into NbrWaitQ where the retransmission cycle will be repeated after a change in connectivity occurs. The additional functionality given to the role of forwarding node allows it to retransmit M_{DDF} after a deferral time which provides assurance that the forwarding chain will not collapse. In order to ensure that the additional functionality assigned to the forwarding node operates efficiently, a node suppression technique is employed. This suppression technique reduces the number of instances erroneous retransmissions occur, in the scenario where the forwarding node did not hear M_{DDF} making forward progress.

The suppression techniques used in the DDF protocol can effectively be placed into two different classes. The first class includes functionality which occurs when a node is in an active state in relation to the forwarding of M_{DDF} (i.e. it is either acting as a forwarding or intermediate node) and the second class includes functionality for the suppression of erroneous forwarding nodes when a node has previously taken part in the forwarding process of copy of M_{DDF}.

Class I: Techniques which suppress or limit unnecessary retransmissions by either a forwarding or intermediate node actively participating in the forwarding process of M_{DDF}, have been built into the functionality of both roles. This functionality has previously been discussed in §4.4.5, but the particular instances are summarised here for clarity:

- Forwarding/passive acknowledgement chain acts as a means of suppressing nodes from retransmitting.
- The retransmission deferral time, t_{def} has been constructed so as to limit retransmissions from the previous link in the forwarding chain.
- Limiting intermediate nodes to a maximum number of retransmissions.
- Limiting forwarding nodes to a maximum number of retransmissions before inserting into $NbrWaitQ$ (case of soft partition) and retrying after a time delay.
- Node x_v receives M_{DDF} from x_u: The state of x_v is such that ($M_{DDF} \in$ ForTable) $\wedge$ (ForTable.NbrWaitFlag = TRUE). The location of node x_u is such that, $P_{x_u} \in FZ_{x_v}$. Node x_v is currently acting as a store and forward node (having previously detected a partition). If node x_u is closer to F_b, in order to prevent x_v from retransmitting M_{DDF} when the NbrWaitQ is next serviced, x_v removes the entry for M_{DDF} from the ForTable and NbrWaitQ since the forwarding requirements of node x_v have been satisfied.
- Node x_v receives M_{DDF} from node x_u: State of x_v such that ($M_{DDF} \notin$ ForTable) $\vee$ ($M_{DDF} \notin$ MsgSeenList). The location of node x_u is such that, $P_{x_u} \in FZ_{x_v}$. Node x_v previously failed to receive M_{DDF} during previous forwarding chain links. Since x_v knows that x_u is closer to F_b than its current position, it does not retransmit.
- Node x_v received M_{DDF} from x_u: State of x_v such that ($M_{DDF} \in$ ForTable) $\wedge$ (ForTable.NbrWaitFlag = TRUE). The location of node x_u is such that, $P_{x_u} \notin FZ_{x_v}$. Node x_v acting as a forwarding node at the head of a partition has received M_{DDF} from node x_u which is further away from F_b than its current position. x_v broadcasts a copy of M_{DDF} to act as passive acknowledgment to node x_u to prevent it from retransmitting further, since x_v knows it is currently at the head of a partition closer to F_b.

Class II: The functionality which suppresses erroneous retransmitting nodes consists of three processes. Firstly, the detection of a suspected erroneous node; secondly tracking to ascertain that the node is indeed erroneous and thirdly suppression of the erroneous node. This functionality is only carried out by nodes that have previously received a copy of M_{DDF} ($M_{DDF} \in$ MsgSeenList), and are not actively involved in the forwarding process, since erroneous retransmitting nodes will be located in previous links of the forwarding chain. The three phases of suppression are discussed below:

Detection: A suspected erroneous forwarding node is detected (previously discussed on page 84) if node x_v on receiving M_{DDF} from node x_u determines that the conditions given in equations (4.10) and (4.13) are TRUE. Action will only be taken by node x_v if it

determines that it is positioned as the most forward node to F_b within FZ_{x_u} in relation to its neighbour nodes receiving M_{DDF} from x_u. Having determined that $FZ_{x_v} \cap FZ'_{x_u} = \{\emptyset\}$, node x_v will proceed to enter the details of node x_u into its SuppList which is linked to the corresponding entry for M_{DDF} in the MsgSeenList. The variable SuppList.*Cnt* is incremented to provide a count of the number of times M_{DDF} has been received from the transmitting node. Each entry in SuppList is indexed by the transmitting node. Node x_v schedules an interrupt at $t_{SuppClear}$, determined from equation (4.16), in order to remove an entry from SuppList on the occasion that x_u stops transmitting before supplist.$cnt = SuppMax$ or node x_v moves out of the transmission range of node x_u. The value of $t_{SuppClear}$ allows for forwarding transmissions up to ForRetrxMax to have occurred prior to clearing the entry from the SuppList. In the case where $FZ_{x_v} \cap FZ'_{x_u} \neq \{\emptyset\}$, node x_v drops the packet since it is not positioned closest to F_b within the transmission range of x_u and therefore not eligible to start the suppression process.

$$t_{SuppClear} = t_{def}^{for} \cdot ForRetrxMax \tag{4.16}$$

Tracking: Prior to taking action to suppress a suspected erroneous node, it is firstly tracked in order to establish that it is indeed an erroneous transmitting node. An erroneous node is tracked until the number of transmissions received reaches *SuppMax*. The value of *SuppMax* is assigned to allow retransmissions from intermediate nodes to be limited by IntRetrxMax and for erroneous forwarding nodes to be suppressed before *ForRetrxMax* is reached. Therefore, the value of SuppMax is set so that $IntRetrxMax < SuppMax < ForRetrxMax$. Moreover, the tracking process allows for the acknowledgement/forwarding chain to carry out suppression without taking any extra measures. After the detection phase, for each copy of M_{DDF} that x_v receives from transmitting node x_u, such that the conditions in equations (4.10) and (4.12) are TRUE, independent of whether or not node x_v is the closest node to F_b within FZ_u, node x_v will update the SuppList.*time* field and increment SuppList.*SuppCnt*. When SuppList.$Cnt = MaxCnt$, the condition has been met which confirms that x_u is an erroneous node and node x_v will now actively proceed to suppress the erroneous forwarding node.

Suppression: Node x_v will suppress node x_u from retransmitting M_{DDF} further by addressing a *SuppMsg* packet to node x_u. Node x_v proceeds to create a M_{Supp} packet which contains information on S, DDA, sequence number, x_v, and t, current time. Node x_v clears the entry for node x_u from the SuppList and then sends M_{Supp} as a unicast packet addressed to node x_u. On reception of the M_{Supp} packet node x_u searches its ForTable to find a matching entry for M_{DDF} from the data included in the header of the M_{Supp}. Node x_u will cancel any scheduled retransmission events and in the case where

node x_u has added M_{DDF} to its NbrWaitQ, it will remove this entry. If NbrWaitQ is now empty node x_u will cancel any queue serving events and resets any flags associated with the NbrWaitQ. Finally node x_u will remove its entry for M_{DDF} from the ForTable and enter metadata for M_{DDF} into the MsgSeenList.

Node x_v detected and started the tracking process according to the rule that it must be the closest node to F_b within FZ_u. This rule is chosen since it provides a criteria for the selection of a node to start the process, thereafter, the position on reception by this node is not relevant since an erroneous node has been detected. If after, x_v had detected and started tracking x_u another node, x_{other}, also started tracking node x_u, node x_v will have suppressed x_u before x_{other} had reached SuppList.$Cnt = MaxCnt$. In this case x_{other} removes the entry for M_{DDF} when the SuppClear event occurs. However, in the case where x_v moved out communication range with x_u, x_{other} will assume responsibility for the suppression of node x_u and node x_v would remove the entry in the SuppList when SuppClearInt interrupt event occurs.

4.4.8 Forwarding Chain Termination at F_b

The propagation of M_{DDF} is constrained within the limits of the DDA through the termination of the forwarding chain at F_b. This is achieved by checking the proximity of F_b each time a node enters a condition where it is required to retransmit or forward a copy of M_{DDF}. If a node detects that $P_{F_b} \in FZ_{x_i}$, the resulting action that the node will take to terminate further forwarding of M_{DDF} is determined by its current forwarding role and status in relation to M_{DDF}. Any node that is currently designated or has determined that it is a forwarding node will be responsible for terminating the forwarding chain. The resulting divergence in functionality when $P_{F_b} \in FZ_{x_i}$, in comparison to that taken for retransmission actions mentioned previously, is listed below.

Condition: $\boldsymbol{P_{F_b} \in FZ_{x_u}}$

$\boldsymbol{M_{DDF} \notin}$ (ForTable $\wedge$ MsgSeenTable) : In the case that node x_v has determined that it is a forwarding node it will enter details for M_{DDF} into MsgSeenTable, update and rebroadcast M_{DDF}. If x_v has determined that it is an intermediate node it will add the details of M_{DDF} to its MsgSeenTable.

($\boldsymbol{M_{DDF}} \in$ MsgSeenTable) $\wedge$ ($\boldsymbol{FZ_{x_v} \cap FZ'_{x_u} = \{\emptyset\}}$) : Update details in MsgSeenTable, update and broadcast M_{DDF}.

RetrxEvent: When a retransmission event occurs for a forwarding node it will copy M_{DDF} from the ForTable, update the required fields, enter details for M_{DDF} into

its MsgSeenTable, remove the entry from ForTable and finally update and broadcast M_{DDF}. In the case of an intermediate node, the details for M_{DDF} will be entered into its MsgSeenTable and the entry removed from the ForTable.

NbrQEvent: When a NbrQEvent occurs and node determines that it is in proximity of F_b for current packet being serviced, it will copy M_{DDF} from the ForTable, update the required fields, enter details for M_{DDF} into its MsgSeenTable, remove entry from ForTable and finally update and broadcast M_{DDF}.

4.4.9 Calculation of Retransmission Deferral Time

In order to overcome the shortcomings of a simplistic implementation of distance deferral techniques (discussed in §4.3) the DDF retransmission deferral time aims to control channel contention at the network layer and provide strict distance-ordered retransmission timing, whilst avoiding excessive channel access delays at the MAC layer. This is achieved by enhancing the conventional distance deferral technique so that it is able to adapt the retransmission delay algorithm according to node density and offered traffic within a node's LZ in conjunction with MAC delay estimation factors determined from the simulation analysis of the distribution of MAC access delay, documented in §6.4.

In order to construct the equation used to calculate the retransmission deferral delay, the following sections proceeds by considering each element that needs to be factored into the equation in order to meet the requirements mentioned above. This section begins by considering deferral timing based on distance and discusses how to adapt and evolve this to cope with the intrinsic lack of reliability and predictability of broadcasts in an uncoordinated radio environment.

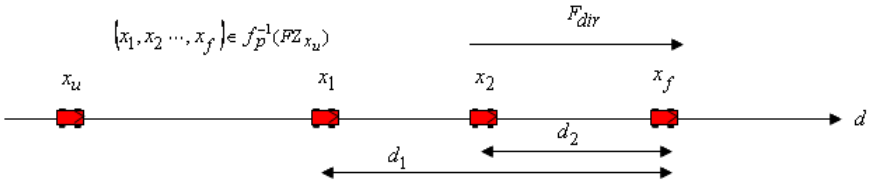


Figure 4.10: Retransmission ordering

Figure 4.10 shows a link in the forwarding chain from node x_u towards F_b. Assuming an ideal radio environment in the scenario depicted in Figure 4.10 where there are no MAC/PHY layer imperfections to overcome, the most distant node x_f in the forwarding zone of x_u will attempt to relay the packet M_{DDF}, broadcast by x_u, first, without delay. Failing this, nodes x_2 and x_1 should attempt to relay the packet in lieu of x_f in the temporal order $t_{def}^{x_2} < t_{def}^{x_1}$. On receiving the transmission by x_u, each node, x_i (where $i = 1, 2 \ldots, x_f$) positioned between x_u and x_f will set up a deferral time,

$$t_{def}^{(x_i)} = 2d_i/c + \tau_{proc} \tag{4.17}$$

where the first term on the right hand side accounts for the signal propagation delay node x_i expects to experience in overhearing a successful retransmission by x_f across d_i (see Figure 4.10) and the second term represents the packet processing time of the same packet by node x_f. In other words, the intermediate node closest to the boundary of the forwarding zone of x_u, in this case x_2, will undertake to retransmit M_{DDF} in the event that it has failed to receive the packet transmission by x_f successfully. In the case where x_1 fails to overhear a transmission from $x_i \mid x_i \in f_p^{-1}[FZ_{x_1} \cap FZ_{x_u}]$ it will be responsible for the retransmission of the packet.

The first problem that arises is the issue of accurate time-interval measurement by nodes x_i since $2d_i/c \ll \tau_{proc}$. This immediately implies that there should be a marked difference between $2d_i/c + \tau_{proc}$ and τ_{proc} for up to some desirable minimum distance d_i to avoid retransmission reversal between a node x_i closest to x_f and node(s) x_i positioned closer to x_u. This problem is due to the fact that the speed of propagation of radio waves is so high that distances of a few 10 m to 100 m correspond to propagation delays in the region of 100 ns, whereas typical packet processing and transmission times are of the order of a few 100 μs. The latter figure is dominated by the amount of time it takes to transmit a MAC frame over the radio channel.

The second and more significant timing issue to resolve arises due to the nature of random access contention over a wireless channel. As previously described in §2.3.3, in reality, once a packet is output to a wireless network interface card, it is temporarily buffered until the MAC layer protocol senses the broadcast medium and determines that no other transmission by another node is currently taking place, or else waits for a current transmission to terminate. This delay, referred to as the channel access delay, is non-deterministic in the presence of (unpredictable) channel contention by other nodes and is, therefore, a stochastic variable. In order to arrive at a variable which models the stochastic phenomenon of the channel access delay, ensemble averaging is carried out over the number of neighbouring nodes and their positions as well as their exact packet transmission rates and timings. This stochastic variable can only be estimated by each node x_i on the basis of observations of the recent history of the radio channel within its LZ. The expected time interval over which node x_i will overhear a successful retransmission from node(s) $x_i \mid x_i \in f_p^{-1}[FZ_{x_1} \cap FZ_{x_u}]$, is thus given by,

$$\Delta t^{(x_i)} = 2d_i/c + \tau_{proc} + \tau_{MAC} \tag{4.18}$$

where τ_{MAC} is a stochastic variable describing the channel access delay experienced by x_f. Given that τ_{MAC} is a random variable, it is evident that node x_i cannot compute this time interval as a deferral time, since τ_{MAC} is described by some distribution that this research characterises empirically in some detail in §6.4. A solution to estimating $t_{def}^{(x_i)}$ from $\Delta t^{(x_i)}$ is to replace the random variable τ_{MAC} by its average value $\bar{\tau}_{MAC}$ which each node, x_i, positioned within FZ_{x_u} can estimate, to give,

$$t_{def}^{(x_i)} = 2d_i/c + \tau_{proc} + \bar{\tau}_{MAC} \tag{4.19}$$

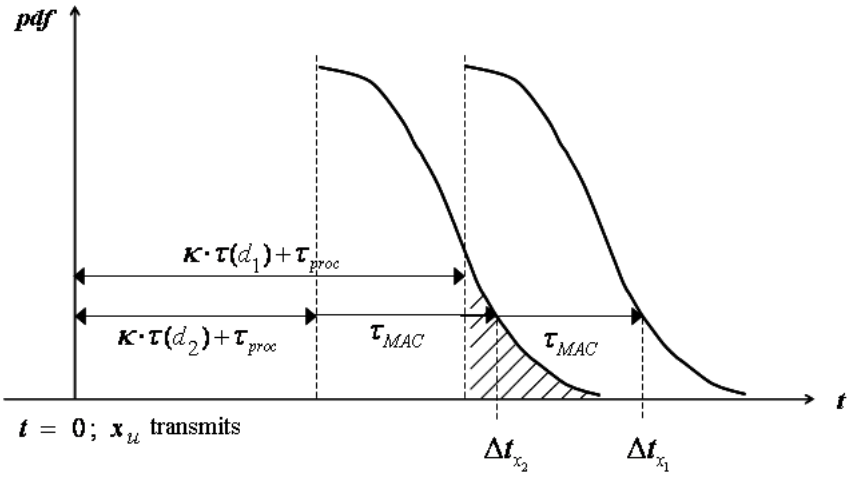


Figure 4.11: Temporal ordering using separation $p(\tau_{MAC})$

However, since each node $x_i \mid xi \in FZ_{x_u}$ determines $\bar{\tau}_{MAC}$ based on channel activity in their *LZ*, it will only be approximately similar for each node. Hence $\bar{\tau}_{mac}$ alone is not enough to provide the required timing discrimination between nodes to achieve temporal ordering. This issue can be demonstrated more clearly by considering the range of $\Delta t^{(x_i)}$ schematically in Figure. 4.11 for nodes x_1 and x_2, which shows the probability density function for the stochastic variable τ_{MAC}. Note that one instance of the random variable τ_{MAC} is plotted for both x_1 and x_2 in Figure 4.11 as this corresponds to a single realisation of the random channel access delay experienced by node x_f. In §6.4.2, this research shows that the probability density distribution $p(\tau_{MAC})$, can be approximated *adequately* by an one-sided Gaussian function,

$$p(\tau_{MAC}) = \frac{1}{\sigma_{MAC}}\sqrt{\frac{2}{\pi}}\exp(-\frac{\tau_{MAC}^2}{2\sigma_{MAC}^2}) \tag{4.20}$$

where σ_{MAC} is found empirically to depend on the level of offered data traffic intensity within a node's local zone. The mean and standard deviation of this distribution can be readily computed to be approximately $0.798\sigma_{MAC}$ and $0.603\sigma_{MAC}$, respectively (see

Appendix B). As will be seen in §6.4, the channel access delay can be as small as τ_{proc} and as large as a few ms, which can become the dominant time-scale which needs to be considered.

The issue of accurate time-interval measurement identified above is now much more problematic, as the difference in the deterministic part of $t_{def}^{(x_i)}$ for the two intermediate nodes needs to be large enough so as to ensure that the two one-sided Gaussian curves depicted in Figure. 4.11 are sufficiently spaced apart in time for values of $d_1 - d_2 > d_{min}$ where d_{min} is some minimum separation threshold distance. Failure to do this may result in retransmission reversal between competing nodes x_i and thus give rise to inefficiencies in the geographic data dissemination protocol. To do this a 'time dilation' factor κ is introduced which is adopted to ensure that $2\kappa d_i/c$ is comparable or greater than typical τ_{MAC} values, where the latter are computed adaptively based on how busy the local broadcast radio channel is. This implies that,

$$t_{def}^{(x_1)} - t_{def}^{(x_2)} > \kappa \frac{2d_{min}}{c} \gg \tau_{MAC} \tag{4.21}$$

The shaded region of the first probability density curve in Figure 4.11 gives the probability that there will be an unwanted reversal for two nodes x_1 and x_2 separated exactly by a distance d_{min}. To minimise this probability the shaded region has been chosen to correspond to $\tau_{MAC} \geq 4\sigma_{MAC}$ (the reason for making this choice is quantified below). Thus,

$$t_{def}^{(1)} - t_{def}^{(2)} > \kappa \frac{2d_{min}}{c} = 4\sigma_{MAC} \tag{4.22}$$

Therefore, κ can be solved for to give,

$$\kappa = \frac{2c\sigma_{MAC}}{d_{min}} \tag{4.23}$$

In the evaluation simulations of the DDF protocol in Chapter 7, a threshold value of $d_{min} = 20$ m is adopted. It is a trivial matter to compute the probability of role reversal at various vehicle separations: At $d_{min} = 20$ m this is $\text{erfc}(2\sqrt{2}) = 6.3 \cdot 10^{-5}$; at 10 m, $\text{erfc}(\sqrt{2}) = 4.6\%$; and at 5 m, $\text{erfc}(1/\sqrt{2}) = 31.7\%$.

The time-dilation factor κ is therefore included in the calculation of $t_{def}^{x_i}$ and a random jitter time τ_{jit} is also introduced, which is drawn from the empirical MAC delay probability density distribution to avoid introducing unnecessary packet collision between nodes that are less than d_{min} apart. The value of $\tau_{jit} \ll \kappa \frac{2d_{min}}{c}$ and therefore, does not affect the ordering of nodes where the separation distance is greater than d_{min} apart.

The time dilation factor and random jitter are therefore adapted to give,

$$t_{def}^{(x_i)} = 2\kappa d_i/c + \tau_{proc} + \bar{\tau}_{MAC} + \tau_{jit} \tag{4.24}$$

Finally, the value of d_i depends on the role that node x_i implicitly assumes on receipt of M_{DDF} since a forwarding node determines d_i on the basis of the maximum transmission range and an intermediate node on the separation distance between itself and the next forwarding node. The term $2d_i/c$ is substituted for using τ_{dist} resulting in a final modified distance deferral time given by,

$$t_{def}^{(x_i)} = \kappa\tau_{dist}^{i} + \tau_{proc} + \bar{\tau}_{MAC} + \tau_{jit} \tag{4.25}$$

where τ_{dist} depends on the role of x_i given by,

$$\tau_{dist}^{i} = \begin{cases} \frac{2(d_{x_u,x_f}^{x_i} - d_{x_i,x_f})}{c} & \text{if } x_i \text{ intermediate node} \\ \frac{2D_{Rmax}}{c} & \text{if } x_i \text{ forwarding node} \end{cases} \tag{4.26}$$

where $d_{x_u,x_f}^{x_i}$ is the Euclidean distance between the position of the source of the transmission, P_{x_u} and the position of the node implicitly determined by node x_i to be the next forwarding node, P_{x_f}. d_{x_i,x_f} is the Euclidean distance between the position of node x_i and the position of the node implicitly determined by node x_i to be the next forwarding node, P_{x_f}. Finally, c is the speed of light, $3 \times 10^8\ ms^{-1}$ and D_{Rmax} is the maximum transmission range.

The above equation provides a method of calculating the total deferral time which is adaptive to local data traffic intensity and provides spatio-temporal ordered forwarding in the presence of an imperfect and unreliable underlying PHY/MAC channel.

Local Data traffic Intensity ($G_{off}^{x_i}$)

As mentioned previously the parameters that are used to estimate $\bar{\tau}_{MAC}$ (or σ_{MAC} in the equation for $t_{def}^{(x_i)}$ are derived from the empirical distributions, detailed in §6.4.3, as a function of the local offered traffic intensity, $G_{off}^{x_i}$. The following describes the method used within the *DDF* protocol to determine $G_{off}^{x_i}$.

The *DDF* protocol monitors channel activity at the network layer to enable local traffic intensity parameters to be determined in order to adapt the calculation of $t_{def}^{(x_i)}$ to real-time channel access delays. This is achieved by entering meta-data for each packet that node x_i overhears into the $PcktCntList$. For each instance that x_i is required to calculate $t_{def}^{(x_i)}$,

the required MAC delay parameters are derived from $G_{LZ}^{x_i}$ which is determined from the summation of all packet types received within a sliding window in time within LZ_{x_i}. The offered traffic, $G_{LZ}^{x_i}$ is given by,

$$G_{LZ}^{x_i} = \bar{\eta} \cdot \Phi_{M_{beacon}} + \frac{\Gamma_{M_{DDF}}}{\Upsilon} + \frac{\Gamma_{M_{Supp}}}{\Upsilon} + \frac{\Gamma_{M_{other}}}{\Upsilon} \tag{4.27}$$

where Υ is the length of the sliding window in seconds. $\bar{\eta}$ is the mean number of neighbours observed within Υ, and $\Phi_{M_{beacon}}$ is the inter-arrival frequency of M_{beacon} given by $1/BeaconRate$. The interarrival rate of M_{beacon} can be determined from $\Phi_{M_{beacon}}/\Upsilon$, however, this does not account for any collisions that may have occurred during Υ. This research accounts for any collisions that may have occurred within Υ to provide a more accurate approximation of the interarrival time of M_{beacon}, by multiplying $\Phi_{M_{beacon}}$ by $\bar{\eta}$. $\Gamma_{M_{DDF}}$ is the sum or all M_{DDF} packets received within Υ, $\Gamma_{M_{Supp}}$ is the sum or all M_{Supp} received within Υ and $\Gamma_{M_{other}}$ is the sum or all other packet types received within Υ.

4.5 Summary

A novel data dissemination protocol called data dissemination forwarding (DDF) has been presented in this chapter. The Chapter firstly introduced and justified design decisions taken in specifying the proposed data dissemination protocol. The remainder of this chapter provided a detailed description of the operation the DDF protocol and the method used in developing the calculation which allows the DDF protocol to adapt to local MAC access delay variations.

CHAPTER 5

SIMULATION ENVIRONMENT

The aim of this thesis is to quantify the performance and identify the limitations of reliable data dissemination algorithms for vehicle-to-vehicle communication. In this chapter, the choice of the evaluation methodology, aims and the constituent parts of the evaluation environment along with the requirements of each entity are presented.

5.1 Evaluation Methods

Evaluating the performance of data dissemination protocols on a large-scale VANET, experimentally, would require a large number of instrumented vehicles, at both considerable cost and complexity. Moreover, the logistical and safety-related issues which need to be taken into account during such measurements can prove to be quite cumbersome. Although projects such as FleetNet [109], C2C-CC [18], PATH [110], SAFESPOT [111] and PReVENT [112] (to name but a few), have carried out real-world performance measurements, the number of equipped vehicles to-date, has been rather limited. As a consequence, an investigation of performance characteristics such as the scalability of the protocol under study during varying vehicular traffic flow conditions, becomes problematic.

Simulation has been the most widely used research tool in order to gain a better understanding of both the performance and limitations of VANET protocols [113]. Although simulation can provide more degrees of freedom than physical experiments in terms of metric variability and number of participating vehicles, it can be limited as memory and processor requirements become prohibitive when the realism of the simulation is increased. However, simulation allows for numerous network configurations and scenarios to be evaluated repeatedly whilst varying parameters of interest. Thus, it enables the benefits of the proposed scheme to be both qualitatively and quantitatively analysed, when compared with other candidate protocols under equivalent conditions. This leads to a deeper under-

standing of how certain parameters of interest can affect the performance of the algorithm allowing optimisations to be made [114]. Such an analysis would be extremely difficult to implement experimentally. However, real world experiments in the target environment are still critical to the understanding of protocol performance, since the characteristics of the physical environment (hardware, propagation effects, etc.) may severely affect the performance of the algorithm. Additionally, experimental data enables validation of the simulation model and the assumptions made. Since simulation can provide such performance flexibility and scenario reparability, protocol performance in this thesis has been performed through simulation.

5.1.1 Accuracy of Simulation

Although simulation provides a flexible and inexpensive approach to researching the performance of network protocols, consideration of the accuracy of such an approach also needs to be taken into account. The results are as reliable and accurate as the abstraction of the system being implemented. Therefore, care needs to be taken in documenting both the abstractions and assumptions implemented in simulation model in order to correctly interpret the results [113]. Similarly, the validation of the model at each stage is an important process in order to substantiate that the model behaves with consistent and satisfactory accuracy that is in agreement with stated simulation goals [115]. This research aims to take into account the points raised in [113, 116, 117], which address the common pitfalls to be avoided in order to provide credible simulation output data.

5.2 Simulation Aims

The simulation platform, detailed in this chapter, has been implemented to enable two separate studies to be performed. The first study requires the distribution of the MAC delay to be characterised and the second study evaluates the performance of the DDF protocol. Both studies require the mobility of the nodes to approximate realistic traffic flow patterns, from free flowing to highly congested vehicular traffic networks.

The study of the MAC delay distribution provides the calibration data which is required to construct a deferral timing calculation that enables the DDF protocol to adapt to LZ variations in both vehicle density and data traffic intensity. The distribution of MAC delay within a node's LZ is determined through the periodic exchange of beacon messages between one-hop neighbouring nodes.

The protocol evaluation study uses performance metrics to analyse the efficiency and reliability of the DDF protocol against two other protocols.

Although both studies use the same simulation tools and elements of the same network model, the methodology and analysis requirements are different. For this reason, the characterisation of the MAC delay distribution and the protocol evaluation studies are presented separately in Chapters 6 and 7 respectively. The proceeding chapter presents the entities which form the simulation platform used in both of the studies.

5.3 Simulation Platform

Figure 5.1 shows the entities of the simulation platform. The first entity in the performance evaluation is the traffic simulator which generates the vehicle mobility traces that are then utilised by the second entity, the network simulator, during the main simulation of the vehicle communication platform. The final entity is the processing of the output data from the network simulator to provide calibration data that is fed back into the network simulator and spatio-temporal data filtering in the case of the protocol evaluation study.

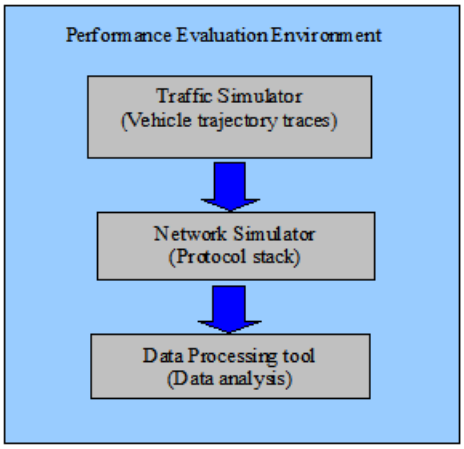


Figure 5.1: Diagram of the performance evaluation environment

5.4 Traffic Simulator

In §2.4.1it has already been mentioned that the mobility of the communicating nodes during simulation plays an central role in achieving a greater fidelity to the target application environment. However, in some instances the performance evaluation of IVC communication protocols has occurred using random mobility models and uniformly spread vehicular traffic, which does not reflect vehicular traffic dynamics, nor the constraints of the motorway. Additionally, evaluation of VANET protocols has also been performed using traffic models with limited realistic mobility, where vehicles are spread homogeneously along a

stretch of road without including lane changing and clustering, which does not reflect realistic traffic dynamics. This section investigates different types of vehicle traffic simulators and their suitability for providing realistic mobility patterns for use in the simulation platform used in this research.

5.4.1 Vehicle Traffic Simulators

Since the seminal paper by Lighthill [118] identifying the analogy between the behaviour of vehicle traffic flow and particle movement in a fluid, the study of traffic flow modelling has been an active field of research for the past 50 years. This has resulted in a plethora of models which can be characterised depending on the level of detail describing different aspects of traffic flow. Vehicle traffic flow models generally fall into three different classes, namely microscopic, mesoscopic and macroscopic [119]. Microscopic models consider the movement of individual vehicles and their interaction with other vehicles within their proximity, allowing individual vehicle trajectories to be traced. Microscopic models incorporate lane changing and car following behaviour models to achieve a high level of realism at the individual vehicle level. In contrast to microscopic models, mesoscopic models consider the overall behaviour of the drivers without individual vehicle detail. These replicate discrete movement but cannot distinguish between individual vehicles. Macroscopic models focus on the collective nature of traffic flow without distinguishing between its constituent parts, and thus do not model vehicles as discrete entities [119].

The chosen network simulation tool models packet flow between individually defined entities and hence models behaviour at the microscopic level. Thus, the appropriate class of vehicle traffic simulator is the microscopic model. This is because it models individual vehicle behaviour along with the interaction of vehicles in close proximity to one another, providing individual vehicle trace files that can be used to model vehicle mobility in the network simulation tool. The mesoscopic and macroscopic modelling tools cannot generate individual vehicle trace files since they do not distinguish individual vehicle behaviour, and, as a consequence are not appropriate.

5.4.2 Microscopic Simulation Tools

There are numerous microscopic traffic simulation tools allowing traffic flow simulation to be carried out [120]. Simulators such as CORSIM [121], PARAMICS [122] and VISSIM [123] are widely used on a commercial basis for the modelling of transportation networks. However, since the traffic flow patterns are required to provide realistic movement traces that will be used by another simulator there is no need to incur the expense of a commercial simulator that also provides a graphical user interface (GUI) visualization tool. For this

reason the output from a research-based microscopic traffic simulation tool developed by the Transportation Research Group (TRG) [124] at Southampton University, called FLOWSIM, is used in this research.

5.4.3 FLOWSIM

FLOWSIM is a microscopic simulation modelling tool which was originally developed for motorway driving scenarios [1]. The modelling within a microscopic traffic simulator occurs at the individual vehicle level. Each vehicle makes its own decisions on speed and lane changing. The central components in emulating realistic driving behaviour are generally the lane changing and car following models. There are different approaches taken in the implementation of such models [125], however, in FLOWSIM both these models are based on fuzzy logic. The lane changing and car following model within FLOWSIM has been validated and calibrated using data collected from field tests on UK motorways, using an instrumented vehicle. It can be seen from [126] that the simulation data replicates closely the field test data.

5.4.4 Traffic Simulator Set-up

The network simulation tool which is discussed in §5.5 requires that all simulation entities are defined prior to the start of the simulation. However, the microscopic traffic simulator defines entities which enter and leave the simulation space (domain) throughout the duration of the simulation. In order to overcome this issue a closed loop traffic network is used where vehicles are gradually injected into the simulation space until a predefined traffic flow rate has been achieved. When the traffic circulating around the closed loop track has reached a steady-state distribution, the trajectory information for each vehicle is updated and recorded to a vehicle trace file for each individual vehicle every 250 ms.

This research investigates the performance of the proposed DDF dissemination protocol using hypothetical scenarios under motorway driving conditions. The motorway environment was chosen for two reasons: firstly the dynamic nature of traffic flow in this environment presents a test of protocol adaptability in the face of varying speeds and clustering; secondly, pragmatic initial deployment of a VANET application is likely to be for motorway driving. Within this environment various characteristics of traffic can be observed from free flowing traffic through to peak hour traffic conditions. Unlike work presented in [55] where the placement and movement of the vehicles/nodes assumes a uniform and

[1] FLOWSIM is now available on a commercial basis and has since been developed further to include urban traffic environments.

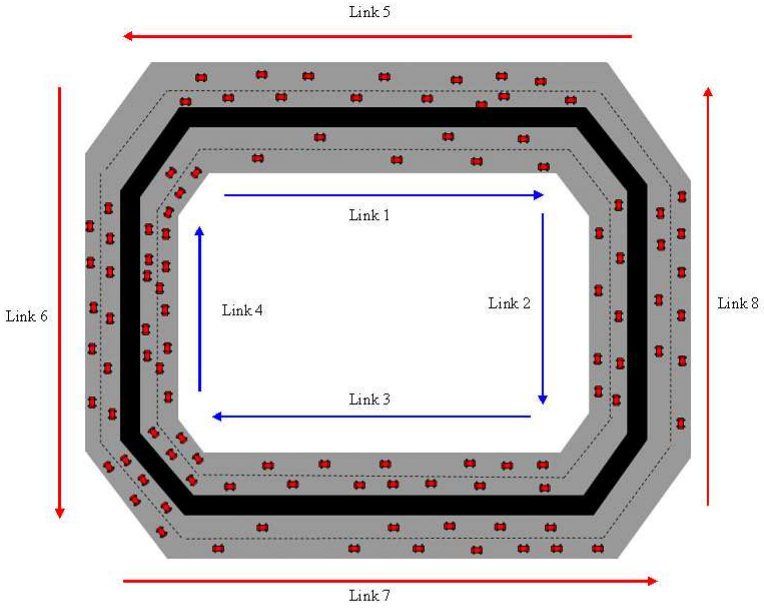


Figure 5.2: Configuration of road traffic network used to generate the vehicle mobility trace files

fully connected distribution around the simulated traffic network, this research uses an inhomogeneous distribution of vehicles, which during free-flowing conditions will introduce partitions in the network. This allows the performance of the proposed protocol to be investigated when the network is not fully connected and partitions occur due to vehicles travelling in convoys or when bunching occurs. Additionally, mixed traffic categories are used in the simulations. The vehicular movement trace files generated from FLOWSIM corresponds to traffic moving on a closed loop bi-directional network with 2 lanes in each direction, and having varying traffic flow densities emulating free-flowing traffic through to dense peak-time traffic with traffic jams caused by fluctuating speed movements. Between these two extremes there are 3 further intermediate traffic flow density simulations, low-medium, medium and medium-high. Vehicle traffic characteristics at motorway junctions are not modelled in this work, it is therefore recommend that the interaction between vehicles entering and leaving the motorway and the effect it has on the main carriageway is an area for further work.

Figure 5.2 shows a pictorial representation of the configuration of the closed loop traffic network defined in FLOWSIM and used to generate the vehicle trace files under varying traffic flow rates. The configuration values used to define the closed loop network along with the various simulation settings are summarised in Table 5.1. Links 1 - 4 represent the clockwise flow of traffic (CLW) around the network and Links 5 - 8 the anti-clockwise flow (ACW). For each of the 5 flow rates simulated, the vehicles circulating around the closed loop consist of a mix of passenger vehicles and heavy goods vehicles (HGV). Flow rates 1 - 3 consist of 33% HGV's and 66% passenger vehicles whilst flow rates 4 and 5

consist of 95% passenger vehicles and 5% HGV's. The HGV's travel at a lower maximum speed than passenger vehicles.

Description	Value
Number of lanes per direction	2
Divided highway	yes
Overtaking and lane changing	Yes
Length of simulated road (circumference)	16 km (5 X 3 km)
Lane width	4 m
Width of central reservation	0.8 m
Maximum speed	130 km/h
Traffic flow rates	600, 800, 1100, 1500, 1700 veh/lane/hr
Position update frequency	250 ms

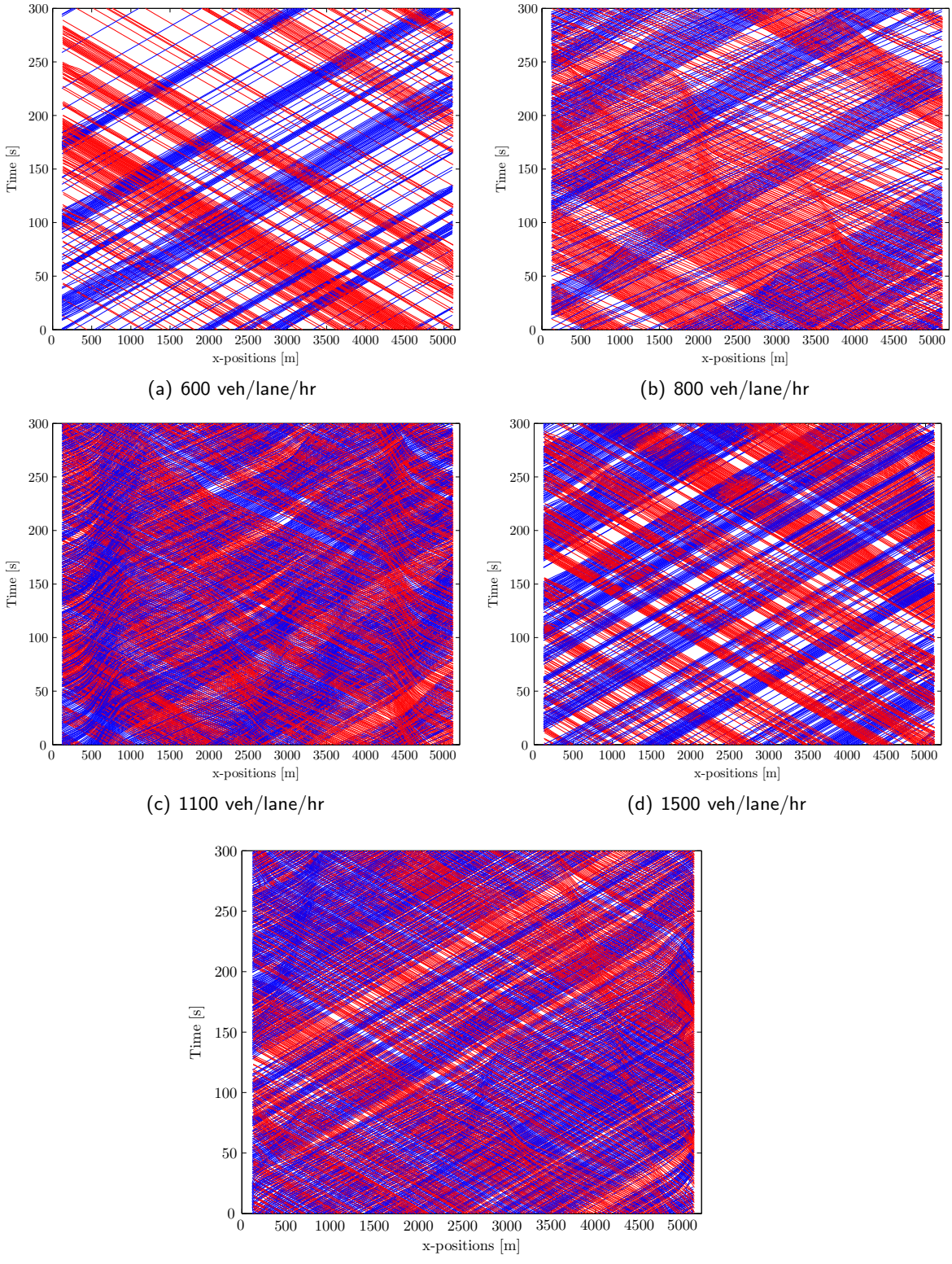
Table 5.1: Parameters used to model the road traffic network

The resulting characteristics of the simulated traffic flow for the varying traffic rates can be visualised with the aid of a space-time diagram. A space-time plot shows the position of each vehicle as a function of time and aids the understanding of traffic flow. Figure 5.3 shows the space-time diagram for links 1 and 5 with both directions of traffic shown on each figure for each flow rate. The blue and red lines denote the vehicles moving around the traffic network in an anti-clockwise and clockwise direction, respectively. Each line on the space-time plot corresponds to a vehicle trajectory and the separation between these lines gives the headway between following vehicles. A vehicle moving at a constant speed will appear as a diagonal straight line (having constant slope equal to the reciprocal of its velocity), whilst a stationary vehicle appears as a horizontal line, overtaking can be seen when two lines cross over one another and curved sections of the trajectories represent vehicles undergoing speed changes such as deceleration. Space-time plots for the separate directions of traffic flow for each road link and traffic flow rate can be found in Appendix C.

The space-time diagrams shown in Appendix C have been used in this research to help place vehicles originating event-based messages, in the network simulator, at different times and locations around the simulated road geometry. This allows varying traffic flow characteristics such as clustering, congestion and partitions to be captured within the DDA used in the protocol evaluation simulations documented in Chapter 7. Partitions within the network can be identified in the space-time plots when the headway distance between vehicles is greater than the transmission range e.g. Figure 5.3(a). In addition the space-time plots were referred to during the analysis of the simulation results in Chapter 7 to confirm congested locations and the occurrence of partitions in network connectivity.

5.5 Network Simulator

Most knowledge currently available in the literature on the performance of wireless *ad hoc* networks and indeed VANETs has been derived through computer simulation. The vast



(a) 600 veh/lane/hr

(b) 800 veh/lane/hr

(c) 1100 veh/lane/hr

(d) 1500 veh/lane/hr

(e) 1700 veh/lane/hr

Figure 5.3: Combined space-time plot of vehicle positions on links 1 (CLW) and link 5 (ACW) for each flow rate.

majority of this work has been developed using three main simulation tools [113] namely OPNET Modeler [127], NS-2 [128] and GloMoSim [129] . However, other simulation tools such as QualNet[2] [130], JiST/SWANS [131], OMNeT++ [132] and CSIM [133] are starting to feature more within the literature. At the time when the choice of simulation tool was made to carry out the work in this thesis, the selection of simulation tools available for wireless *ad hoc* networks and supporting a suitable implementation of a wireless access scheme was essentially limited to the former mentioned group of simulators. All of these simulation tools provide environments which support the testing and development of network protocols and also provide libraries of standard communication protocols as well as contributed models from the simulation community.

OPNET Modeler version 9.1A was chosen to simulate all aspects of the VANET performance. This network simulator was chosen since at the time of making this decision OPNET Modeler was the most highly developed simulator supporting the appropriate functionality. Moreover, OPNET Modeler incorporated a well developed and more realistic model of the radio channel through its radio pipeline stage when compared to both NS-2 and GloMoSim. Additionally, since OPNET Modeler is available on both an academic and commercial basis, full user support was provided when problems were encountered, unlike NS-2 and GloMoSim, where support was and is via discussion groups.

5.5.1 OPNET Modeler Simulation Methodology

OPNET Modeler is a discrete event simulator. It provides a broad and detailed framework for the modelling of both wired and wireless networks. The OPNET Modeler uses an object-oriented approach to model development. A model can be defined as a *CLASS*, which can be reused any number of times within the simulation by creating multiple instances of the model. OPNET Modeler provides a library of basic, standardized and vendor models of network entities, all of which can be customized according to the users requirements. The simulation environment consists of a hierarchy of three main levels of modelling, the simulation scenario or network model, the node model and the process models. Each level is defined via dedicated editors that allow the user to define and manipulate objects and attributes which constitute the model. The wireless channel is modelled in OPNET Modeler using a modular framework known as the transceiver pipeline.

[2]QualNet is a further developed and commercial implementation of GloMoSim. GloMoSim is no longer under active development.

5.5.2 Network Model

The network model, also known as the simulation scenario is the highest level in the simulation hierarchy and defines the topology of the network. The instances of the model class referred to as a node are defined at this level. The attributes associated with a model class are promoted from lower levels within the model hierarchy, allowing the user to configure each node instance at a global or individual level by selecting or specifying attribute values within limits defined by the lower levels.

5.5.3 Node Model

The node model is the second level in the modelling hierarchy and is defined using the node editor. The node editor allows the system being modelled to be depicted as functional blocks called "modules" along with the flow of data between the various modules. There are a number of module types which can be selected to build the node model, each with differing capabilities and levels of programming flexibility. The transmitter and receiver modules do not allow the user to change any functionality, only to set predefined parameters. Whereas the processor, queue and external system modules allow module behaviour to be programmed by the user by defining a custom process model. Such modules are assigned process models to achieve the required functionality. The various interactions between modules are defined using either a "packet stream" or a statistic wire. Packet streams are used to transfer packets between modules, whereas a statistic wire can be used to convey control information directly between modules and are typically used when one module needs to monitor the state of another.

5.5.4 Process Model

The process model is the lowest level of the model hierarchy and is defined using OPNET Modeler's process editor to describe the behaviour of processor and queue modules. The tasks that these modules execute are called processes and communication between each process is supported by interrupts. Process models are expressed in a language called Proto-C, which consists of state-transition diagrams (STDs), an OPNET Modeler defined library of kernel procedures and the standard C/C++ programming language. The process editor uses a finite state machine (FSM) approach to support the specification of any type of protocol or algorithm. States and transitions define graphically the progression of a process in response to events. The conditions that specify what happens within each state when an event occurs are specified using Proto-C.

5.5.5 Radio Pipeline Model

The radio transceiver pipeline model is used by the wireless module to model the transmission of packets between nodes. The pipe line consists of 14 stages, which can be seen in Figure 5.4; a brief summary of the functionality of each stage can be found in Appendix D and further detailed information can be found in [134]. The functionality governing each stage of the pipeline is defined using the Proto-C language in external files which are selected as an attribute in the wireless module for each stage. OPNET Modeler provides a library of standard pipeline models. However, the user has the flexibility of amending these files or defining their own in order to meet specific requirements. The pertinent settings of both the transmitter/receiver and physical channel are covered in §5.7.3.

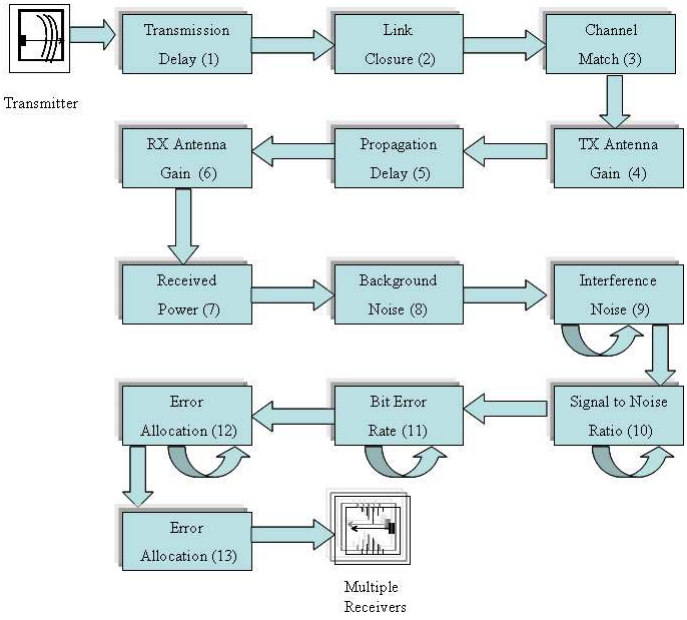


Figure 5.4: OPNET Modeler radio transceiver pipeline

5.6 Data Analysis

Although OPNET Modeler provides functionality for recording user-defined statistics, all performance data was written to external files to allow post-processing to be carried out. This was necessary since the protocol performance data was required to be filtered based on a spatio-temporally resolved criteria, which could only be realised as a post-processing exercise. The need for spatio-temporal filtering arises since a packet is addressed to different regions within the road traffic network. Therefore, the data of interest lies within this region between the times that message is originated, and the time for the message to propagate within the boundaries of the region. This can vary between regions

in accordance with vehicle traffic dynamics.

Furthermore, by saving data to external output files, a permanent record of complete simulation traces also facilitated model validation to be performed. This was particularly important so as to be able to validate the performance and accuracy of the protocol during the development stage. Additionally, error-checking code was developed and used in the algorithm to ensure that each process invoked by a particular event within the process model followed the expected flow of events, ensuring that all conclusions drawn from the same data set are mutually consistent.

In order to model the distribution of the MAC access delay, all performance data was again written to external output files in order to carry out the mathematical analysis, as a post-processing exercise.

MATLAB [135] was used to carry out all post-processing and plotting of results for the protocol performance evaluation and the mathematical analysis of the distribution of the MAC access delay. In order to ensure that the position information obtained from the OPNET Modeler simulations was correct, the vehicle trace files generated from FLOWSIM were also used as an input to the post-processing exercise. This enabled vehicle position data from the OPNET Modeler simulations to be cross-correlated with the FLOWSIM data to ensure that the vehicle positions were correct at the time the data was recorded.

5.7 OPNET Modeler Model Implementation

The model created in OPNET Modeler provides a framework for the simulation and evaluation of protocols operating in a VANET environment. This framework provides a generic hierarchical structure of models (as discussed in §5.5) consisting of a network, node and process models. Rather than creating separate models for the different protocols presented in this work, the VANET framework incorporates all such protocols, allowing certain functionality to be reused. Protocol selection is determined by the user as a simulation setting. The following section describes the hierarchical entities which constitute the simulation framework.

5.7.1 Network Model

Since the number of nodes constituting the network topologies vary from 600 to just under 1700 for the different traffic flow rates, creating and assigning attributes manually would be extremely laborious. Therefore, the *XML topology export* function was used from within OPNET Modelers network editor to export the settings and attributes of a

node to an output file. The resulting XML file was then manipulated programmatically[3] outside the OPNET Modeler environment in order to create the topology files and assign values to specific attributes. The required number of node definitions were created and the node ID, MAC address, mobility trace file and initial coordinates were assigned to the appropriate attributes programmatically for each traffic flow rate. These files were then imported back into OPNET Modeler node editor to create the different network topologies corresponding to the different traffic flow rates. The validation of both the node count and assignment of the attribute values was performed by outputting the assigned vehicle trace file, along with the node and MAC address and initial position coordinates to an output file during the model initialisation stage for each simulation run. The address values were manually cross-checked to ensure that they had been assigned correctly and the position coordinates were cross-correlated programmatically with the corresponding file allocated to the trajectory file attribute.

5.7.2 Node Model

As shown in Figure 5.5 the node model has been loosely split into sections to resemble the Open Systems Interconnection (OSI) stack. However, the layers shown in the node model equate to those layers necessary for the implementation and evaluation of the proposed DDF protocol.

The different modules which constitute the node model along with a brief description of their functionality are given as follows:

***Source* module:** This module generates packets according to a specific packet size and inter-arrival distributions. The process model behind this module is the *basic source* model which is supplied with the OPNET Modeler library. Changes were made to this model at the initialisation stage to ensure that the first packet generation interrupt for each node occurred randomly. This reduces the probability of neighbouring nodes transmitting a packet at the same time, which would cause an artificial broadcast storm problem to occur at the initialisation stage and for each beacon packet generated thereafter (previously discussed in §4.4.4).

***Application* module:** On receipt of a packet from the upper layer the Application Manager preforms the following functions: If the node is designated as a source node it will allow the packet to be sent to the lower layers; otherwise it will be deleted and the current event terminated for this node. The protocol selected by the user, which is an input attribute to this module and set as a simulation attribute, is communicated to the

[3]The term *programmatically* means using a program to accomplish a task.

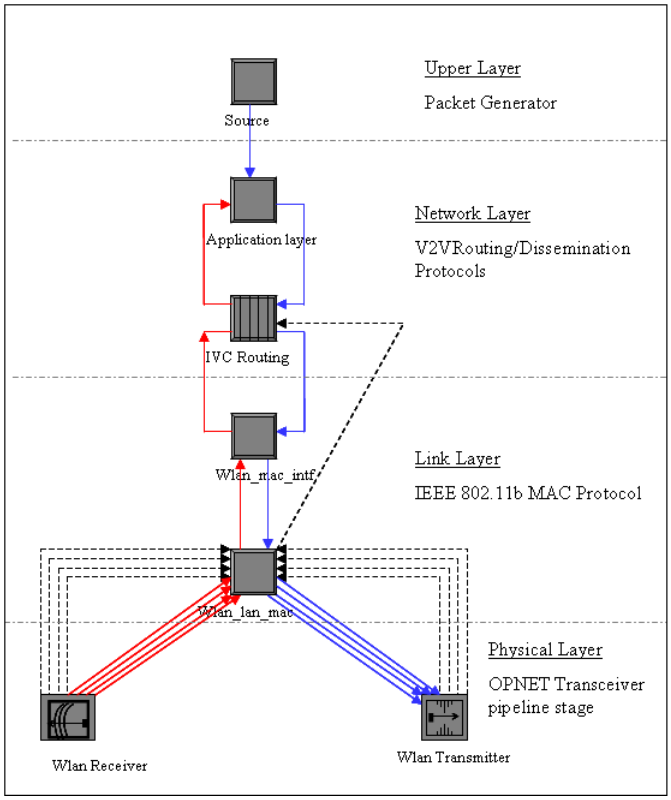


Figure 5.5: IVC node model

IVC module using an Interface Control Information (ICI)[4]. The packet received from the upper layer along with the ICI are communicated to the lower layer.

***IVC routing* module:** The routing module receives both the packet and the ICI from the upper layer. Depending on the protocol type indicated in the ICI this module will execute the appropriate functionality for the selected protocol (DDF or the comparison protocols mentioned in §7.3). Beacon messages are generated by this module since neighbour knowledge information is used by the DDF protocol. For any packet type being communicated from this module to the lower layer, an ICI is generated to communicate the destination node of the packet[5]. If this module receives a message from the lower layer (wlan_mac_interface), then depending on the packet type (defined in the packet header field) the appropriate functionality will be followed.

***Wlan_mac_intf* module:** This module is provided by OPNET as part of the WLAN model and provides an interface between the MAC and upper layers. It sends and receives packets to both the MAC and IVC modules.

[4]An ICI is a structured collection of data that is transferred between processes, as a form of inter-process communication allowing the exchange of user-defined data.

[5]If the packet is to be broadcast then the ICI contains a null character, which indicates to the MAC module that the data packet will be broadcast at the selected channel data rate.

***MAC* module:** This module contains the scheme which governs the method a node accesses the communication medium. The model is supplied in the OPNET Modeler library, and is an implementation of the Medium Access Control (MAC) scheme based on the IEEE 802.11b standard [136]. The IEEE 802.11b physical layer settings of the OPNET Modeler implementation of this model was based on direct sequence spread spectrum (DSSS). The IEEE 802.11b process model supplied by OPNET Modeler was modified so that the media access delay is communicated to the MAC layer on each occasion that a packet is transmitted to the communications medium.

***Wlan_Transmitter and wlan_receiver* modules:** These modules define the physical parameters which govern the characteristics of the simulation channel.

Further details regrading attribute settings for the MAC module and WLAN transmitter and receiver modules are discussed in §5.7.3 and listed in Table 5.2.

5.7.3 Physical Layer Characteristics

The salient physical layer settings are discussed in this section only. The operation and flow of events for the OPNET radio pipeline, are included in Appendix D.

Transmitter and Receiver Settings

At the time of making the decision on the radio channel settings[6] a transmit range (T_r of 300 m was being discussed within the V2V communications community. It was therefore decided to adopt this transmission range as a basis for the simulations in this thesis. The transmit power was chosen to give a successful packet reception at the receiver positioned 300 m from the source of the transmission (with 2-ray path loss model) with a receiver threshold Rx_{thresh} value of -101 dBm.

Directional antennas increase spatial re-use, increasing the local throughput at a node whilst only addressing certain locations and thus reducing the locally seen density of nodes. Since worst case local neighbourhood vehicle density conditions are being investigated, as many as possible of the potential 1-hop neighbours should be reached. An omni-directional antenna is therefore used in the simulations to provide for worst case local neighbourhood vehicle density. Additionally the use of an omni-directional antenna facilitates the requirement within the proposed protocol to exploit the propagation of messages along a carriageway using the opposite direction of traffic flow. The proposed

[6]All the decisions regarding the physical and MAC layer settings used in the simulations were made prior to the emergence of IEEE 802.11p. A comparison between the main IEEE 802.11p PHY and MAC settings and those used for IEEE 802.11b in the simulations used in this research, is given in Appendix E

antenna patterns for WAVE devices [14] are essentially peanut shaped, centered around the vehicle, the omni-directional antenna patterns provides a good enough approximation given that the motorway provides a locally linear topology over a transmission range of 300m.

Propagation Model

The default model used in the radio pipeline to determine the level of signal attenuation between the transmitter and receiver uses the Friss path loss equation which models propagation in free-space assuming a line-of-sight path between the transmitter and receiver. The default model has been modified to include the flat earth two ray model that assumes a direct and a reflected ray at the receiver. In this model, there are two regions with different slopes separated by a breakpoint beyond which the path loss follows $1/R^4$ law or 6dB per octave. In the first region, however, the received signal oscillates due to destructive and constructive addition of the two rays. This phenomena is known as Fresnel zone clearance, where the breakpoint can be viewed as the distance d_{brk} at which the ground just begins to obstruct the first Fresnel zone and the signal changes from following free-space propagation to being influenced by the ground reflection. The break-point distance is determined by equation (5.1), where h_t and h_r are the height of the transmit and receive antennas respectively and λ the wavelength of the centre frequency of the communication channel. The average received power P_r is determined depending on the separation distance, d between the transmitter and receiver. For transmitter separation distance for which $d \leq d_{brk}$ holds, P_r is determined from equation (5.2) and for $d > d_{brk}$ equation (5.3) is used. G_t and G_r are the antenna gains of the transmit and receive antennas respectively and P_t is the transmit power.

$$d_{brk} = \frac{4\pi h_t h_r}{\lambda} \tag{5.1}$$

$$P_r = \frac{P_t G_r G_t \lambda^2}{4\pi d^2} \qquad when \quad d \leq d_{brk} \tag{5.2}$$

$$P_r = \frac{P_t G_r G_t h_t^2 h_r^2}{d^4} \qquad when \quad d > d_{brk} \tag{5.3}$$

Although the 2 ray model is a simplification of the path loss mechanisms that would be realistically encountered in a motorway environment, motorways are typically locally flat over the 300 m maximum transmission range. Deviations from the 2-ray model are therefore going to arise, generally, due to shadowing by other vehicles [137] and as a

consequence this makes the results reported in this thesis overly conservative. Therefore, as part of further work, it is recommend that a more realistic path loss model is used to model fading characteristics.

In VANET simulations, and indeed in this thesis, the nodes in the network simulator are dimensionless and therefore have no effect on signal propagation at the physical layer. However, in reality the metal structures of vehicles, in particular HGVs, will have an effect on signal propagation. Two characteristic effects which can arise are: Firstly, the line-of-sight (LOS) path between communicating vehicles is often obstructed by other vehicles which can cause multipath components to be dominant at the receiver resulting in variations in the transmission range; secondly, vehicle structures can also act as reflectors causing transmissions to extend beyond the maximum transmission range. In both of the above cases, the effect is compounded in the case of HGVs. Although the above mentioned effects resulting from vehicle structures are not included in the propagation model, the message persistence functionality of the DDF protocol, proposed by this thesis, is able to cope with the resulting variations in the transmission range.

Attribute	Setting
Frequency	2.402 GHz
Transmit power	1 mW
Reception threshold	$7.33e^{-14}$ W (-130 dbm)
Antenna type	Omnidirectional
Antenna gain	0 dBi
Antenna height	1m
Path-loss model	2-ray

Table 5.2: Physical layer and radio channel settings

MAC Layer

The MAC layer parameter settings were set in accordance with IEEE 802.11b, as shown in Table 5.3.

Attribute	Setting
Frequency	2.402 GHz
Data rate	11 Mbps
Basic data rate	1 Mbps
Slot time	$20\mu s$
SIFS time	$10\mu s$
PLCP overhead (preamble)	$192\mu s$

Table 5.3: IEEE 802.11b MAC settings

5.7.4 IVC Process Model

The DDF protocol specified in §4.4 and the comparison protocols (ODAM and flooding) specified in §7.3, are implemented in the IVC process model shown in Figure 5.6. The simulation settings for the one-hop neighbour MAC delay and protocol simulations are given in §6.3 and §7.5.4. This section provides a general description of the functionality of each state within the IVC process model.

***init* state:** This state performs the initialisation stage of the process model. User-defined attributes and free variables are loaded and initialised (see §7.5.4 for a list of the free variables for each protocol and §6.3 for the MAC analysis variable settings). Memory storage entities for the DDF protocol detailed in §4.4.2 and the comparison protocols detailed §7.3 (ODAM and flooding) are also created and initialised. The external output files to which all performance data is written are also created at this point. In the case of functionality requiring events to occur at frequent intervals, such as the broadcasting of beacon messages, bandwidth usage and positioning mechanisms, self interrupt events are scheduled to initiate the first occurrence of these events. On completion of the initialisation stage the process transitions to the *idle* state.

***src_arrvl* state:** When a packet arrives at the upper layer input stream of the IVC module, the process transitions into this state. The data is extracted from the ICI which was sent with the data packet from the upper layer. The data contained in the ICI determines the protocol type and the functionality that the process will follow thereafter. The packet from the upper layer will be encapsulated in a newly created packet (depending on the protocol type) and the packet header fields populated. The packet is then sent on the output stream to the lower layer. The process will then transition back to the *idle* state.

***hello_beacon* state:** This state is responsible for the periodic generation of beacon packets, M_{beacon}. The time at which a node first transitions into this state is scheduled to occur randomly from the *init* state at $T_{beacon\_int} = R[0, T_{beacon}]$. This avoids synchronisation and hence collisions between neighbouring nodes. The beacon message is created and the header fields populated. The next beacon generation event is scheduled to occur at $T_{beacon} + T_{current}$. Finally, M_{beacon} is sent on the output stream of the IVC module to the lower layers for broadcast by the MAC layer, prior to transitioning back to the *idle* state

***bw_util* state:** Bandwidth utilisation characteristics are determined at regular intervals throughout the simulation. The first event is scheduled in the *init* state, thereafter bandwidth utilisation events are scheduled from within the *bw_util* state. The process transitions to the idle state upon completion.

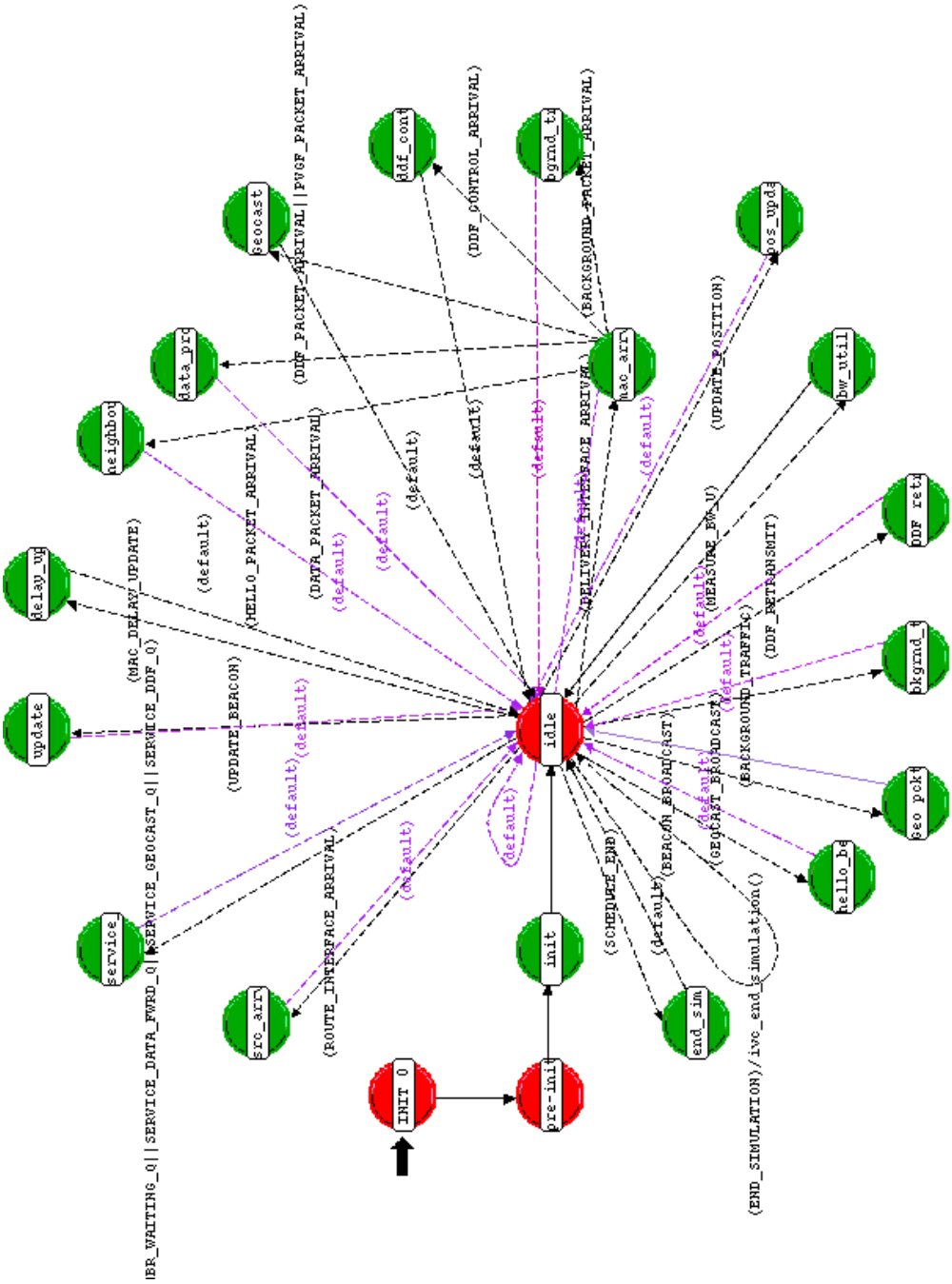


Figure 5.6: IVC routing process model

***position_update*:** This state is responsible for periodically updating a node's position history for the ODAM comparison protocol discussed in §7.3. The time at which a node first transitions into this state is scheduled to occur from the *init* state at $T_{PosHist}$. The PosHistory cache is updated with the current coordinates of the node. Prior to transiting back to the *idle* state, a position update event is scheduled to occur at $T_{current} + T_{PosHist}$.

***delay_update* state:** When a node gains access to the communication medium for which it has had to defer transmission, it communicates the MAC access delay, τ_{MAC}, to the network layer. The MAC access delay is communicated via the statistic wire connecting the MAC and IVC Routing modules, as shown in Figure 5.5. The interrupt from the statistic wire causes the process to enter the *delay_update* state. The MAC access delay along with other network layer information (node ID, number of neighbours, $T_{current}$, etc.) is written to an output file for post processing. On completion the process hands control back to the MAC layer. The data collected from this state is used for the analysis of the distribution of MAC delay study only, and for this reason, the statistic wire is only operational during this study.

***mac_arrvl*:**The process transitions into this state on receipt of a packet arriving at the IVC module from the lower layer input stream. If the packet type is DDF, the details of the packet along with data from the LzTable are added to the DDFPcktCntList. This data is used to determine bandwidth usage statistics and parameters for the calculation of τ_{def}. In the case of an ODAM, the packet details are added to the ODAMPcktCntList. Depending on the packet type, the process then transitions into the corresponding state and proceeds to run the associated protocol functionality depending on current status relative to the packet.

***neighbour_discovery*:** If the packet type is a beacon message the process follows a forced transition into this state from the *mac_arrvl* state. The packet will be added to the LzTable if no entry is previously found from the node originating the packet. However, if the node already appears in the LzTable, then the recorded information will be updated. If the relative position of the neighbour node has changed with respect to the node holding the current process the change in position class count will be updated in sPosClass. If a class change occurs and data packets exist in the NbrWaitQ, an interrupt will be scheduled to begin servicing the NbrWaitQ. After the process completes its operation it will transition to the idle state.

***geocast_packet*:** The process follows a forced transition into this state if the packet received by the *mac_arrvl* state is either a PVGF (for future development) or DDF date packet. Depending on the protocol type, the appropriate functions are called to process the packet. Upon completion, the process will transition to the *idle* state.

***ddf_control*:** The process transitions into this state from the *mac_arrvl* state on reception of M_{Supp}. This type of packet is unicast to the receiving node in order to suppress it from operating as an erroneous retransmitting node. The status of the node is changed and any scheduled retransmission interrupts are cancelled and removed from the execution list. Upon completion, the process transitions to the *idle* state.

***DDF_retrx* state:** This state occurs when the DDF retransmission deferral timer expires. Prior to retransmitting an updated version of M_{DDF}, the entry in the ForTable is updated. Any packets received by the node prior to transitioning into this state will be cached in DefCache. When this state is entered, if the process finds no packets in the DefCache from the required forwarding direction, then the packet pointed to in the ForTable (for which the event occurred) will be either retransmitted and another retransmit event scheduled, or dropped, depending on the forwarding status of the node and the value of RetrxCnt. This state is also used as a transition for ODAM (comparison protocol, detailed in §7.3.1) functionality when its retransmission timer expires.

***Update state*:** This state occurs if a node has not received M_{beacon} from a neighbour node within the expected time window. If the number of missed messages exceeds the maximum allowed limit, then the process removes the entry from the LzTable and adjusts the vehicle position class count in sPosClass that the node belonged to. The process checks to see if the current node holding the process has any entries in its LzTable. If no neighbouring nodes exist then any events scheduled to service the NbrWaitQ are cancelled. If the maximum number of allowed missed has not been exceeded, the process then schedules another update event for this entry in the LzTable and then transitions to the idle state.

***service_queue*:** This state can occur for two reasons. Firstly, when changes in the membership of the vehicle position *Classes* occur (as a result of either a vehicle changing its relative position with the current node or a new neighbour being detected) and the NbrWaitQ contains packets, but is not currently being serviced. Secondly, a queue service event is scheduled to occur at regular intervals until the end of the queue has been reached and neighbouring nodes exist in the required forwarding direction.

***end_sim* state:** Termination of a simulation should occur when the region over which the message was addressed spreads out from the source vehicle and reaches its boundaries. However, this time depends on temporal events and can vary, particularly when partitions occur for the network topology within the region addressed by the message. Therefore, specifying a long simulation time could in some cases extend simulation time unnecessarily when the network is fully connected and conversely during partitions, the simulation may terminate prematurely. For this reason, functionality exists for each algorithm to schedule a simulation termination event when the boundary condition(s) have

been met. When a simulation termination event occurs the process generates a simulation termination interrupt.

5.8 Summary

This chapter looked at different methods in which VANETs are modelled and evaluated. Simulation was used in this case due to cost implications of implementing an experimentally testbed coupled with the difficulty of performing repeatable measurements whilst varying test parameters of interest. The entities of the evaluation platform used to model and analyse the proposed data dissemination protocol were presented and the reason for their selection discussed. The importance of using realistic traffic mobility patterns was discussed and the space-time plots showing the dynamics of the different vehicle traffic flow rates, used in the simulations, was presented. Finally, the functionality and settings of the network simulation model were presented.

CHAPTER 6

EMPIRICAL ANALYSIS OF MAC ACCESS DELAY CHARACTERISTICS

6.1 Introduction

Chapter 4 introduced the proposed DDF protocol which aims to reduce the shortcomings of the IEEE 802.11 CSMA scheme in order to provide reliable and timely dissemination of data for incident warning applications in the IVC environment. The proposed protocol employs channel contention mitigation techniques at the network layer. Instead of trying to mitigate for collisions through the binary exponential backoff mechanism which has the potential of introducing long transmission delays (not acceptable for ITS application), this research reduces the probability of the first collision occurring by introducing much smaller stochastic delays into the timing of packet retransmissions.

The channel contention resolution requirements incorporated into the calculation of the retransmission deferral timing employed in the DDF protocol were discussed in depth in §4.4.9. The retransmission timing is adaptive to offered data traffic intensity and node density within a node's LZ and ensures retransmissions occur according to the distance from the source of the last transmission. In order to adapt the timing to the offered traffic intensity and node density within a node's LZ, knowledge of the delay statistics to access the communication channel is required.

This chapter analyses the MAC channel access delay which is probed through the periodic exchange of beacon messages in order to characterise the distribution of MAC channel access delay within the LZ of a node, empirically. The analysis is carried out through simulation using realistic vehicle movement traces for varying densities of vehicle traffic.

This chapter briefly discusses the decision for characterising the distribution of MAC channel access delay through simulation as opposed to using analytical models. The

elements of the IVC simulation platform employed in the this study are then introduced along with the parameters of interest. An analysis of the results is presented, and the chapter ends with a discussion of the method used to derive the distribution of channel access delay parameters, used in the calculation of the DDF retransmission deferral delay discussed in §4.4.9.

6.2 Requirement for MAC Delay Characterisation

As mentioned in §2.5.1 using the IEEE 802.11 MAC scheme in broadcast mode provides no acknowledgments of successful frame reception. This expedites communications and economises on bandwidth usage, at the expense of reliability. The broadcasting mechanism employed in IEEE 802.11 requires that prior to transmitting a packet the communication channel must be sensed to be clear. The carrier sense scheme does not prevent collisions occurring due to hidden or exposed terminal problems. If either the channel is sensed to be busy or a collision has occurred a 'backoff' mechanism is invoked to resolve the channel contention at the expense of introducing a random delay, which can be large if the channel is heavily contested.

Given the above shortcomings of IEEE 802.11 MAC scheme when operating in broadcast mode, the DDF protocol has been designed to resolve channel contention occurring between nodes. Contention is resolved in this research by incorporating knowledge of the MAC access delay into the calculation of the retransmission deferral time in (4.25).

As has been seen from §4.4.9, the factors considered in the construction of the equation used to calculate the retransmission deferral time, τ_{def} require knowledge of the mean and variance of the MAC channel access delay stochastic variable. This is required for two reasons: firstly, to ensure that a node does not retransmit prior to the intended next node within the forwarding chain; secondly, to ensure retransmissions are ordered according to distance away from the next forwarding node.

There are a number of studies that have analysed the MAC broadcast delay characteristics of the CSMA/CA mechanism employed by 802.11 both analytically and through simulation. The next sub-section proceeds to investigate any synergies with the requirements discussed above.

6.2.1 MAC Channel Access Delay Studies

The channel access delay characteristics of radio channels have received extensive attention in the literature. Early, analytical models [138] assessing the throughput and delay

performance of CSMA determined under saturated offered traffic conditions (terminals always have a packet ready to transmit) and in an idealised environment (in terms of node position and density, as well as offered traffic distributions). The authors of [138] further developed their model to take into account performance analysis with hidden terminals and showed that hidden terminals significantly degrade the performance of CSMA [139].

Various analytical models and simulation studies have been advanced over recent years to evaluate the performance of the IEEE 802.11 MAC layer. The analysis of MAC access delay forms a key element of these models. In particular, Bianchi presented in [140] an analytical model which provided, at the time, the most complete closed form throughput and delay expressions for IEEE 802.11. However, Bianchi's model provides closed form expressions for the saturation throughput assuming ideal channel conditions for unicast traffic flows only. Bianchi's model has formed the basis of many subsequent publications. In particular, [141, 142] analysed packet delay for IEEE 802.11 assuming saturated conditions using a model evolved from Bianchi's work. Based on channel state probabilities the authors of [143] presented an analytical model and simulations for IEEE 8021.11 DCF MAC delay and computed the mean and variance of service time for a saturated network under ideal channel conditions. In [144] an analytical method is used to study the distribution of the back-off delay for 802.11 DCF, again under saturated conditions.

The above mentioned studies provide mean MAC service time for saturated networks for unicast traffic and are not applicable to the case of non-saturated broadcast traffic. Although, non-saturation conditions have been considered in the literature [145], the result is analyses that only describe the mean and at most SD and on very rare occasions go as far as evaluating delay distributions, but do so through complex numerical evaluations of the proposed models.

In addition, to the best of our knowledge at the time of carrying out this work [97], there were no studies analysing the delay distribution of the MAC access delay for broadcast traffic under non-saturating traffic conditions. However, more recently [146] modelled mean service time for IEEE 802.11 in the presence of both unicast and broadcast traffic under non saturated conditions. In [147] the throughput of IEEE 802.11 broadcast scheme was modelled analytically considering hidden terminals. The topologies assumed in both [146, 147] are unrealistic and do not reflect the types of topology and indeed mobility patterns which are required in this research analysis.

The most closely related work in the vehicular environment that has studied MAC access delay which was carried out concurrently with work in this thesis was reported in [10]. Similar to the approach used in this thesis, the mean MAC delay is modelled through simulation and is presented for periodic 1-hop broadcast traffic using varying contention

window size and packet generation rates. In addition, Ma *et al.* [148] formulated a function for the distribution of MAC service time and found mean packet transmission delay in a VANET environment. However, they could not make their solution tractable to include vehicle mobility and the hidden terminal problem and, therefore, omitted these critical considerations. For this reason, and in addition to the fact that the implicit numerical evaluation of the delay distribution function for service time is too expensive to compute in real-time, the work in [148] does not meet the requirements of the research in this thesis.

The requirement in this thesis is for an approximate analytical model which is parametrically linked to specific variables in the LZ environment in order to make the DDF protocol adaptive. For this, the distribution of the channel access delay is required rather than simply the mean and standard deviation. An analytical approach to determining the MAC delay over varying traffic loads, and varying mobility is beyond the scope of this thesis and would constitute a separate research programme. To the best of our knowledge, nobody has modelled the MAC delay distributions as opposed to the mean MAC delay analysis from a restrictive set of assumptions. Most studies fall back on numerical simulations because an analytical model becomes exceedingly complex to pursue. This is largely a consequence of the dynamic nature of vehicular traffic flow which makes node mobility, hidden terminal effects and variable data saturation conditions intractable to model analytically.

This research, therefore, carries out a numerical-statistical analysis through simulation of channel access delay for broadcast traffic using the IEEE 802.11 MAC scheme, with varying densities of realistic traffic flow traces and varying levels of offered traffic. This enables the first two moments of the MAC access delay distribution to be determined and the shape of the distribution to be characterised.

6.3 Simulation Study

The simulation tool used for the analysis of MAC access delay distribution, along with the radio propagation model and implementation of the IEEE 802.11 MAC protocol has been described in detail in §5.5. This section focuses on the functions and settings of the IVC simulation platform (presented in Chapter 5), which have been employed in this study.

6.3.1 Methodology

In order to derive the distribution of the MAC access delay for varying levels of offered traffic within a node's LZ the time delay in transmitting a packet needs to be measured, i.e. the MAC access delay, τ_{MAC}, and the number of nodes present within a nodes LZ

at the time of transmission under varying vehicular traffic densities and traffic generation rates. In this research the choice is made to measure the MAC delay versus the number of neighbours. This is because the distribution of this stochastic variable depends on levels of contention within a node's LZ, which in turn depends on the number of nodes within the LZ since the delay distribution changes depending on overhead variability within a node's LZ.

For the purposes of this research, MAC access delay is defined to be the measure of time between the time the packet is at the head of the MAC service queue to the time of transmission. This time accounts for back-off delays, interframe spaces and channel busy time. Channel propagation and processing times are accounted for in the calculation of τ_{def}. Queuing delay is not included in the measure of MAC access delay since it is assumed that emergency messages will be processed with the highest priority and its overall effect will be negligible.

The MAC delay is monitored through the periodic exchange of beacon messages according to the functionality described in §4.4.9. The RPC algorithm of the Local Zone Connectivity Tracking protocol is not required for this analysis and therefore is not operational. In order to provide an accurate picture of the number of nodes within a node's LZ, neighbouring nodes are removed from a nodes LzTable when a beacon message has not been received prior to NBR_UPDATE event occurring. This occurs at a time given by $T_{update} = T_{current} + (T_{beacon} \times \delta_{nbr})$, where δ_{nbr} is a factor which allows for potential channel access time delays.

At the start of simulation each node applies a jitter delay to their transmission time in order to prevent synchronisation between neighbouring nodes. For each packet transmitted to the radio channel, the MAC layer transmission delay is communicated to the network layer. When the network layer receives an interrupt generated by the MAC layer it proceeds to record the MAC delay, node ID, time, and number of neighbour nodes within its LZ to an external data file.

The simulation settings for the MAC delay analysis are shown in Table 6.1. The simulations were carried out at five different packet interarrival rates for three different vehicular traffic densities for 300 seconds each. This simulation time captures the varying dynamics of the vehicular traffic flow which are used to ensemble average over all positions along the closed loop highway shown in Figure 5.2. The attributes of the vehicular traffic trace files assigned to each node are described in §5.4.3. The simulation settings for the physical layer and IEEE 802.11 MAC layer are shown Tables 5.2 and 5.3, respectively.

Parameter	Value
Traffic flow rates ($veh/lane/hr$)	600, 800, 1100, 1500, 1700
Length of highway (km)	16
M_{beacon} interarrival time, τ_{source} (ms)	50, 100, 150, 200, 250
M_{beacon} size	250 Bytes
$Nbrupdate$	$2 * \tau source$
Time per simulation (sec)	300
MAC scheme	IEEE 802.11b

Table 6.1: Simulation settings for MAC delay analysis

6.4 Analysis of MAC Access Delay Results

This section proceeds to analyse the data generated from the simulations of beacon exchange between neighbouring vehicles, for varying traffic flow rates and packet interarrival times.

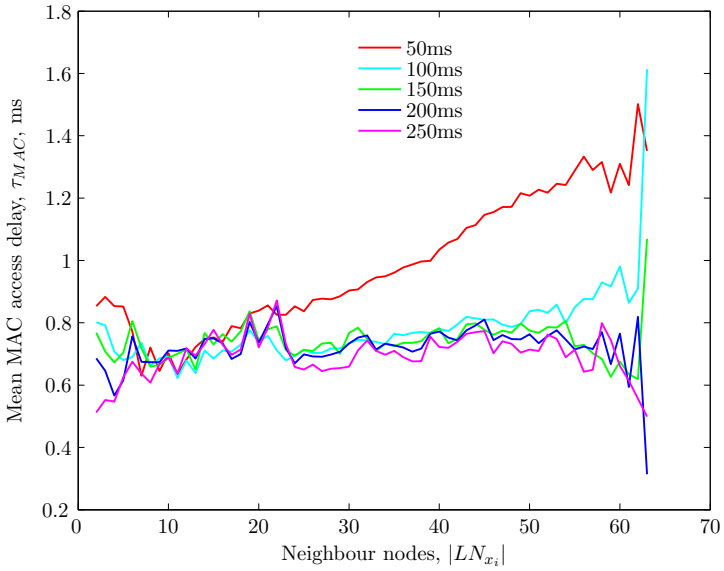
Firstly, this section analyses how the effect of competition for the channel affects the MAC delay. This is a function of the packet inter-arrival time and the number of neighbour nodes within a nodes LZ, $|LZ_{x_i}|$, at each vehicle traffic flow rate. The ensemble average is then taken over all MAC delays as a function of the number of $|LZ_{x_i}|$ and all vehicle positions for each of the five packet interarrival rates and three different vehicle traffic flow rates. MAC delays where $\tau_{MAC} = 0$ are omitted from the analysis since they represent cases of no channel contention.

For each of the traffic flow rates Figures 6.1(a) to 6.1(c), confirm, as expected, that the MAC delay is dependent on the offered traffic within a node's LZ. That is, the MAC delay increases with an increase in $|LZ_{x_i}|$ and the message interarrival rate. Varying the traffic flow rate from free flowing through to congested scenarios, captures the vehicular topology dynamics over the simulated geometry. In the case of the highest traffic flow rate the MAC delay begins to rise sharply for a mean interarrival interval of 50 ms as the offered load begins to reach the knee point.

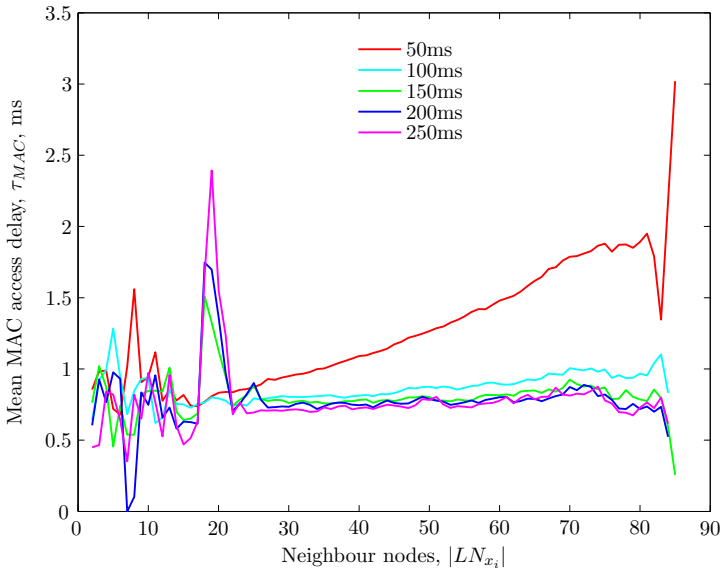
6.4.1 Dependence of Delay Statistics

Since the aim of this research is to make the DDF protocol adaptive to varying traffic flows and offered load within a node's LZ, this sub-section proceeds to investigate the dependence of the MAC delay statistics on the offered load. The vehicular traffic dynamics associated with the various traffic flow rates does not have any significant influence on the MAC delay characteristics apart from increasing with $|LZ_{x_i}|$ as expected.

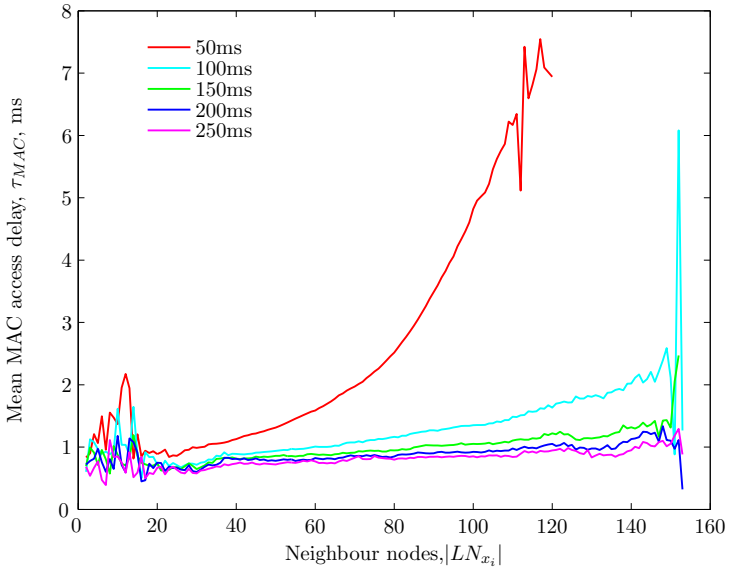
$$G_{LZ}^{x_i} = \frac{|LZ_{x_i}|}{\tau_{beacon}} \tag{6.1}$$



(a) Low density vehicle traffic flow



(b) Medium density vehicle traffic flow



(c) High density vehicle traffic flow

Figure 6.1: Mean MAC access delay per neighbour node density for varying mean beacon interval from 50 ms - 200 ms

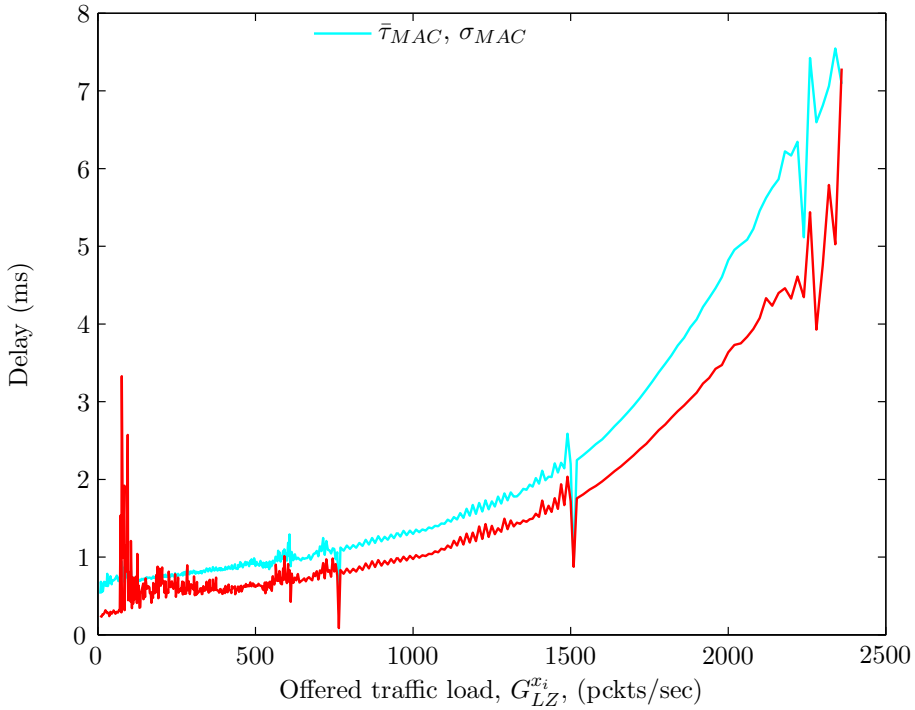


Figure 6.2: Mean and SD of MAC delay versus offered traffic

The plots of Figures 6.1(a) to 6.1(c) are coincident when they are re-plotted as $\bar{\tau}_{MAC}$, versus $G_{LZ}^{x_i}$, where $G_{LZ}^{x_i}$ is determined from equation (6.1) at each data point as seen in Figure 6.2. The standard deviation of MAC delay, σ_{MAC}, also shown in Figure 6.2 follows a similar curve.

MAC Delay Distribution

Firstly, n order to investigate the distribution of the MAC delay statistics the histograms of MAC channel access delay for each value of $G_{LZ}^{x_i}$ are plotted. When plotting the histograms of τ_{MAC} for each value of $G_{LZ}^{x_i}$, the existence of long delay outliers were observed, of the order of a few 10s of ms. Figure 6.3 shows the maximum observed values of τ_{MAC} at each value of $G_{LZ}^{x_i}$. Further analysis of the long delays shows that approximately 1 - 1.5% of observed data points can be classed as outliers, as shown in Figure 6.7. Data points are classed as outliers if $\bar{\tau}_{Mac} > 3 * \sigma_{MAC}$ which means that data set will consist of $\sim 99.7\%$ of the original data points if the outliers are removed.

On comparing the effect of removing the outlier points from the data set on $\bar{\tau}_{mac}$ and σ_{MAC}, it can be seen from Figure 6.5(a) that $\bar{\tau}_{MAC}$ is unaffected by the removal of the outliers whereas a negligible reduction ($< 5\%$) can be seen for σ_{MAC} at higher values of $G_{LZ}^{x_i}$ in Figure 6.5(b).

Given that the removal of the long values of τ_{MAC} does not have any significant effect on

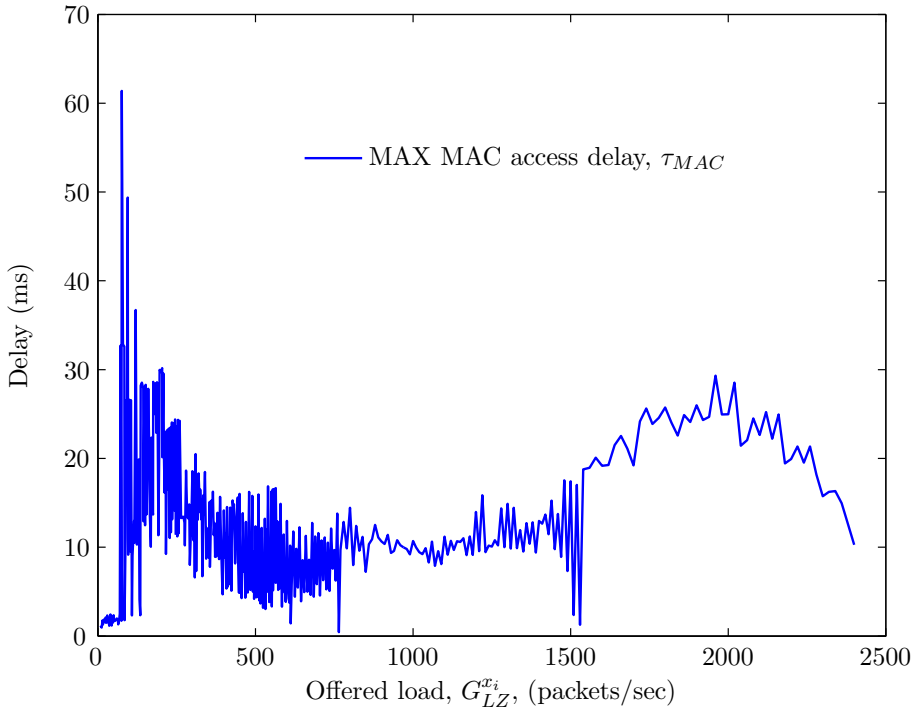


Figure 6.3: Maximum MAC access delay

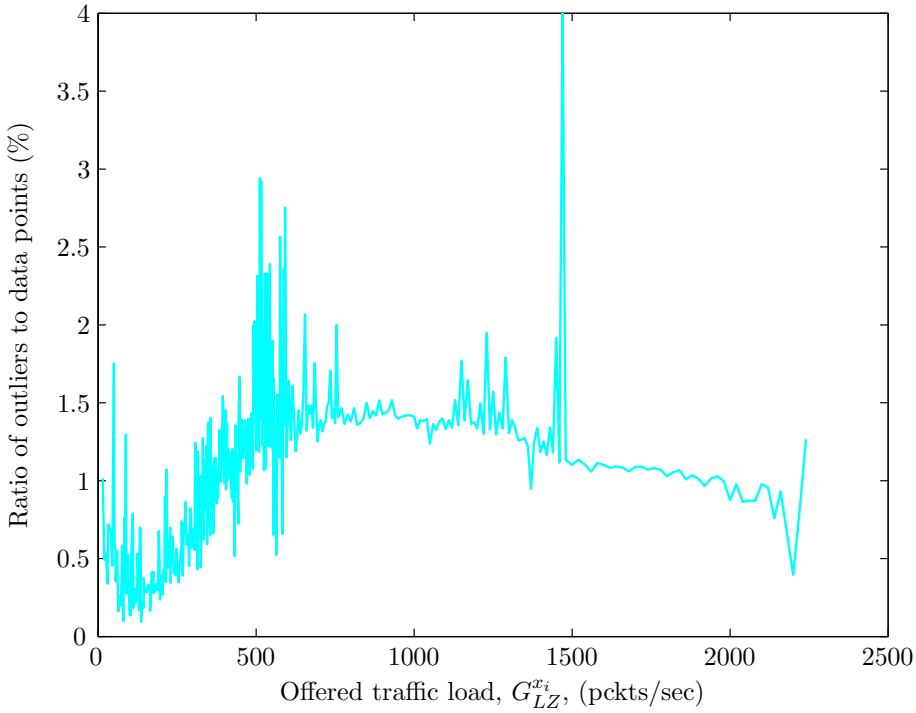
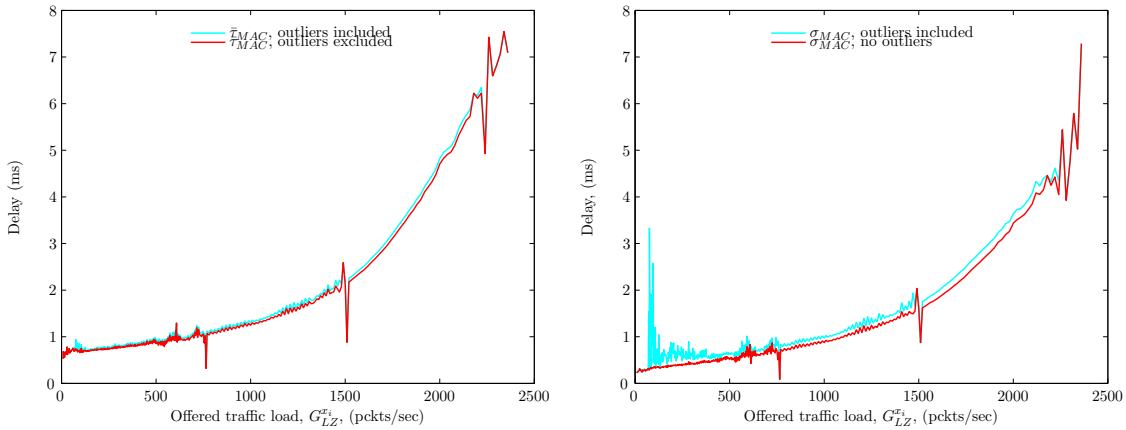


Figure 6.4: Ratio of outliers to data points at each value of $G_{LZ}^{x_i}$



(a) $\bar{\tau}_{MAC}$ with and without outliers

(b) σ_{MAC} with and without outliers

Figure 6.5: Comparison of Mean and SD of MAC access delay with and without outliers

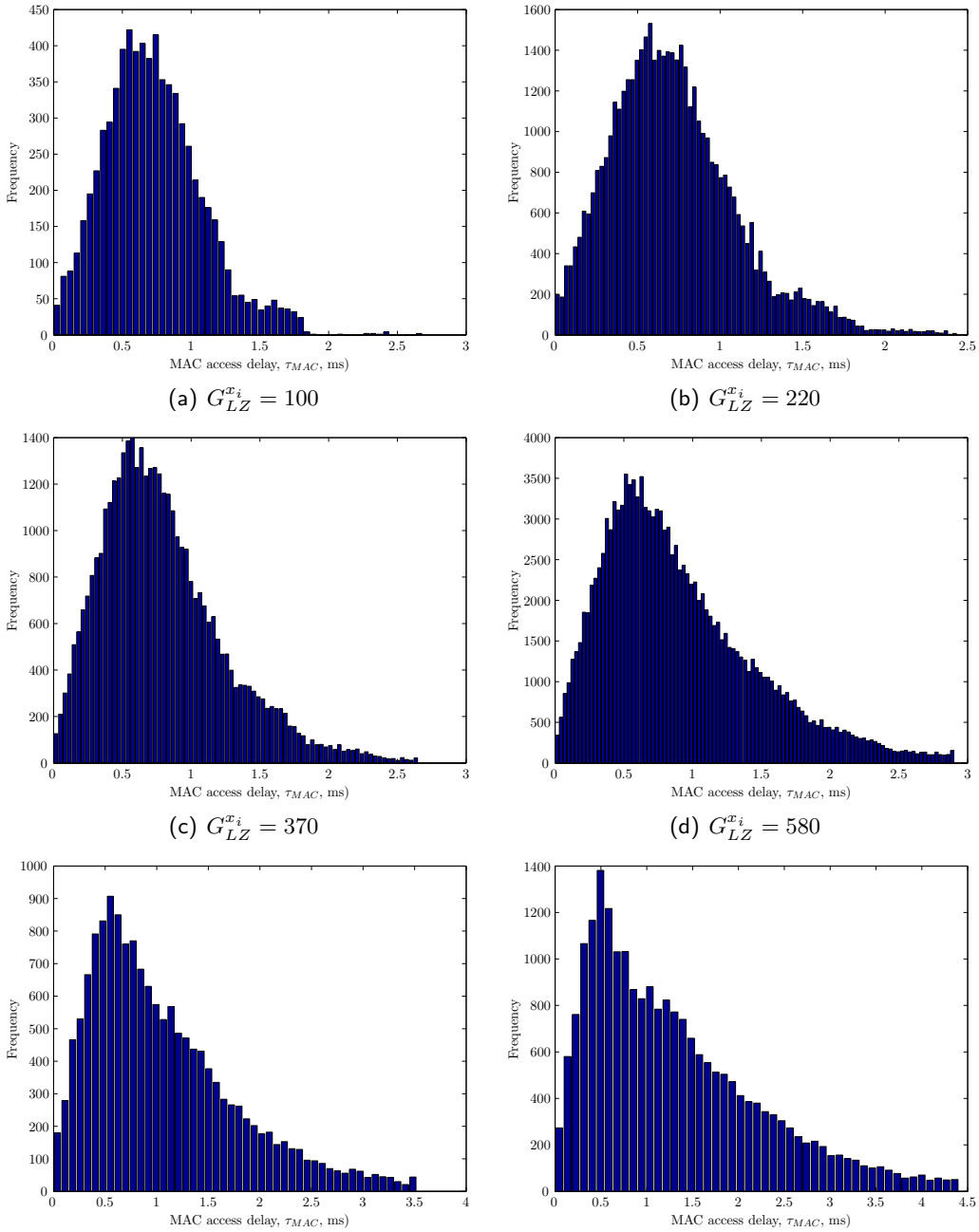
$\bar{\tau}_{mac}$ and σ_{MAC} it was decided to omit the outlier data points from any further analysis on the data set for the following reasons: firstly, there are insufficient numbers of such delay values in order for them to be modelled in a meaningful way; secondly, their incorporation would render the distance deferral mechanism inefficient since the data dissemination speed would be too slow to be of use for emergency applications, as they represent the worst case scenarios.

On plotting the histograms of the MAC delay at each value of $G_{LZ}^{x_i}$ with the outlier data points removed, it can be observed that for the histograms where $G_{LZ}^{x_i} \leq 1000$ (pckts/sec) the distributions are similarly shaped to Rayleigh or Rice ones. Whereas, for values of $G_{LZ}^{x_i} \geq 1000$ (pckts/sec) the distributions approximate a one-sided Gaussian distribution. Figures 6.6(a) to 6.6(f) show a selection of histograms which reflect the variation in distributions from lower to higher values of $G_{LZ}^{x_i}$.

6.4.2 Distribution Fitting

In order to model and derive the requirement for the retransmission ordering constant, κ, and the random jitter, τ_{jit} as discussed in §4.4.9, it is required to fit a distribution to the empirical histograms in order to define the distribution of τ_{MAC} as a function of $G_{LZ}^{x_i}$.

Given the variation in the shape of the histograms, performing a best statistical distribution fit for τ_{MAC} with $G_{LZ}^{x_i}$ as a free parameter will lead to an excessively complex model since the histograms cannot be described by a single distribution. Given that the approach of this researcch aims to demonstrate the concept of dynamically adapting the deferral time based on $G_{LZ}^{x_i}$, such a complex model is not necessary for modelling the distribution of $P(\tau_{MAC})$. Thus in order to err on the side of caution, which guarantees

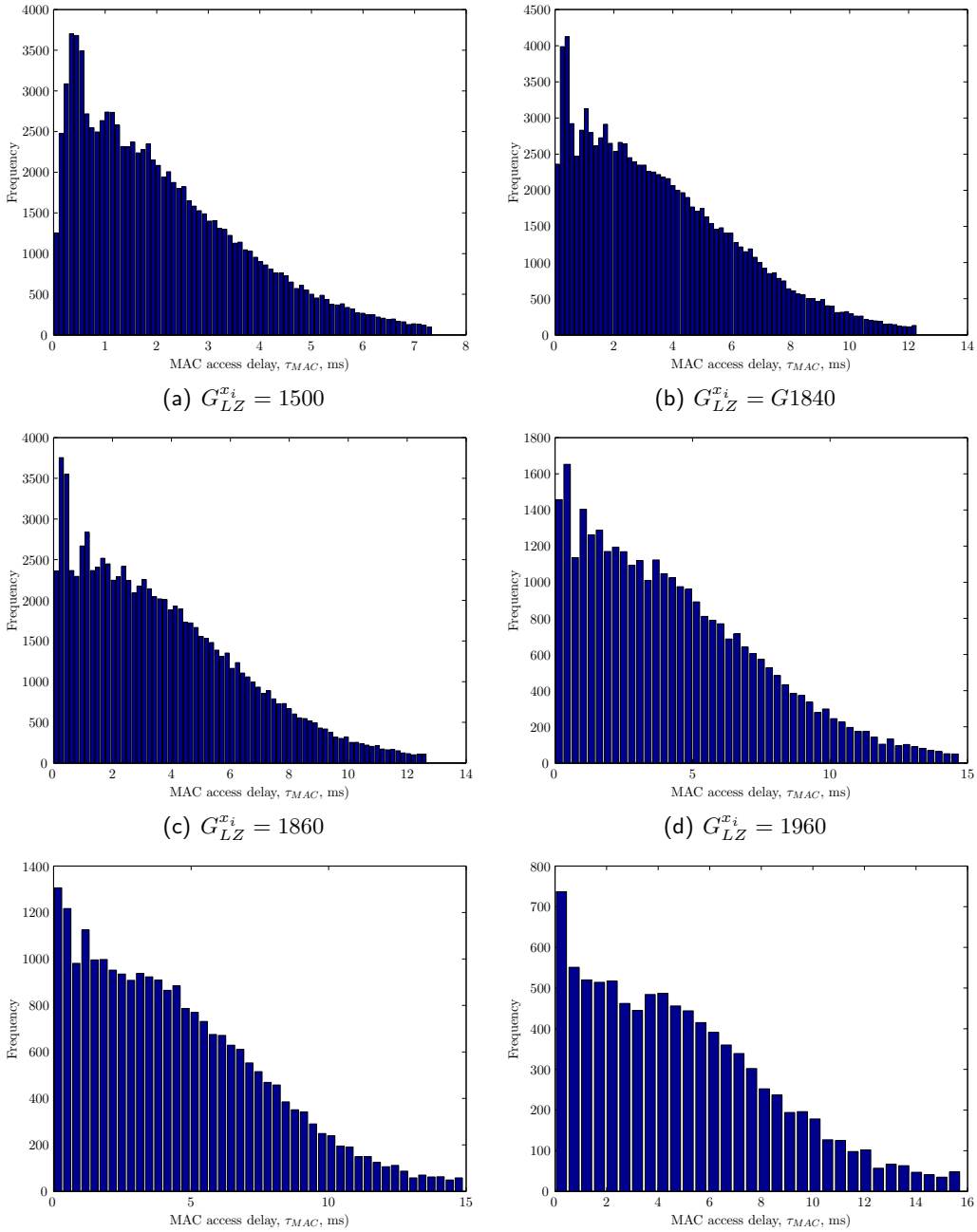


(a) $G_{LZ}^{x_i} = 100$ (b) $G_{LZ}^{x_i} = 220$

(c) $G_{LZ}^{x_i} = 370$ (d) $G_{LZ}^{x_i} = 580$

(e) $G_{LZ}^{x_i} = 750$ (f) $G_{LZ}^{x_i} = 1010$

Figure 6.6: Histogram of MAC access delay, τ_{MAC} at selected values of offered load, $G_{LZ}^{x_i}$



(a) $G_{LZ}^{x_i} = 1500$ (b) $G_{LZ}^{x_i} = G1840$

(c) $G_{LZ}^{x_i} = 1860$ (d) $G_{LZ}^{x_i} = 1960$

(e) $G_{LZ}^{x_i} = 1980$ (f) $G_{LZ}^{x_i} = 2040$

Figure 6.6: Histogram of MAC access delay, τ_{MAC} at selected values of offered load, $G_{LZ}^{x_i}$

that the DDF protocol will work adaptively for all LZ conditions, this research proposes to adopt the half-Gaussian PDF for τ_{MAC} for all Values of $G_{LZ}^{x_i}$. Moreover, since the MAC delay study is not the main focus of this thesis, the approach taken in this research provides a pragmatic and approximate solution to the modelling of the stochastic MAC delay.

As an additional check on the applicability of the half-Gaussian PDF, this research uses the well-known result that the ratio of the mean to SD is constant for Gaussian random variables (e.g. [149]). Appendix B shows that for a half-Gaussian distribution $\bar{\tau}_{MAC} = 0.798\sigma$ and $\sigma_{MAC} = 0.603\sigma$, where σ is the double-sided Gaussian distribution SD, which leads to the ratio of $\frac{\bar{\tau}_{MAC}}{\sigma_{MAC}} = 1.3$. The empirical ratio of $\frac{\bar{\tau}_{MAC}}{\sigma_{MAC}}$ from the histograms along with the theoretical value is plotted in Figure 6.7 and confirms that indeed for $G_{LZ}^{x_i} \geq 1000 pckts/sec$ the half-Gaussian distribution is acceptable for the purposes of this research.

Although a half-Gaussian PDF does not fully describe the shape of the histograms for $G_{LZ}^{x_i} < 1000$, the effect of using a half-Gaussian distribution will have little impact on the second moment of τ_{MAC} since the distributions as shown by the histograms show an initial rise which is steep over a small range of delays which occur with low probability. Therefore, this section of the distributions for $G_{LZ}^{x_i} < 1000$ has a small probability of occurrence and the effect of not modelling it will be small, since the second moment of τ_{MAC} will remain largely unaffected.

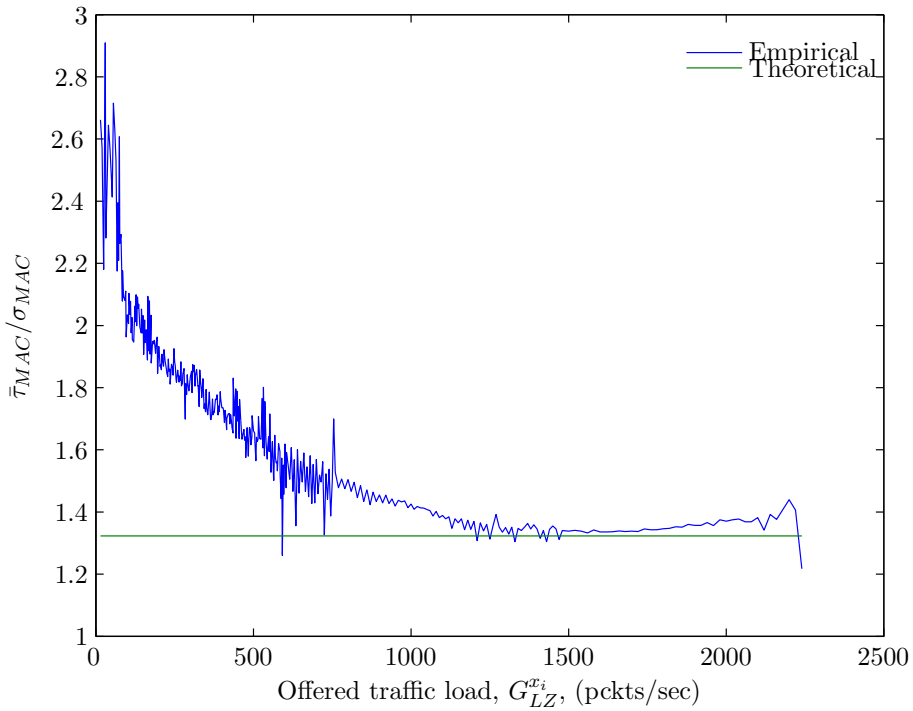
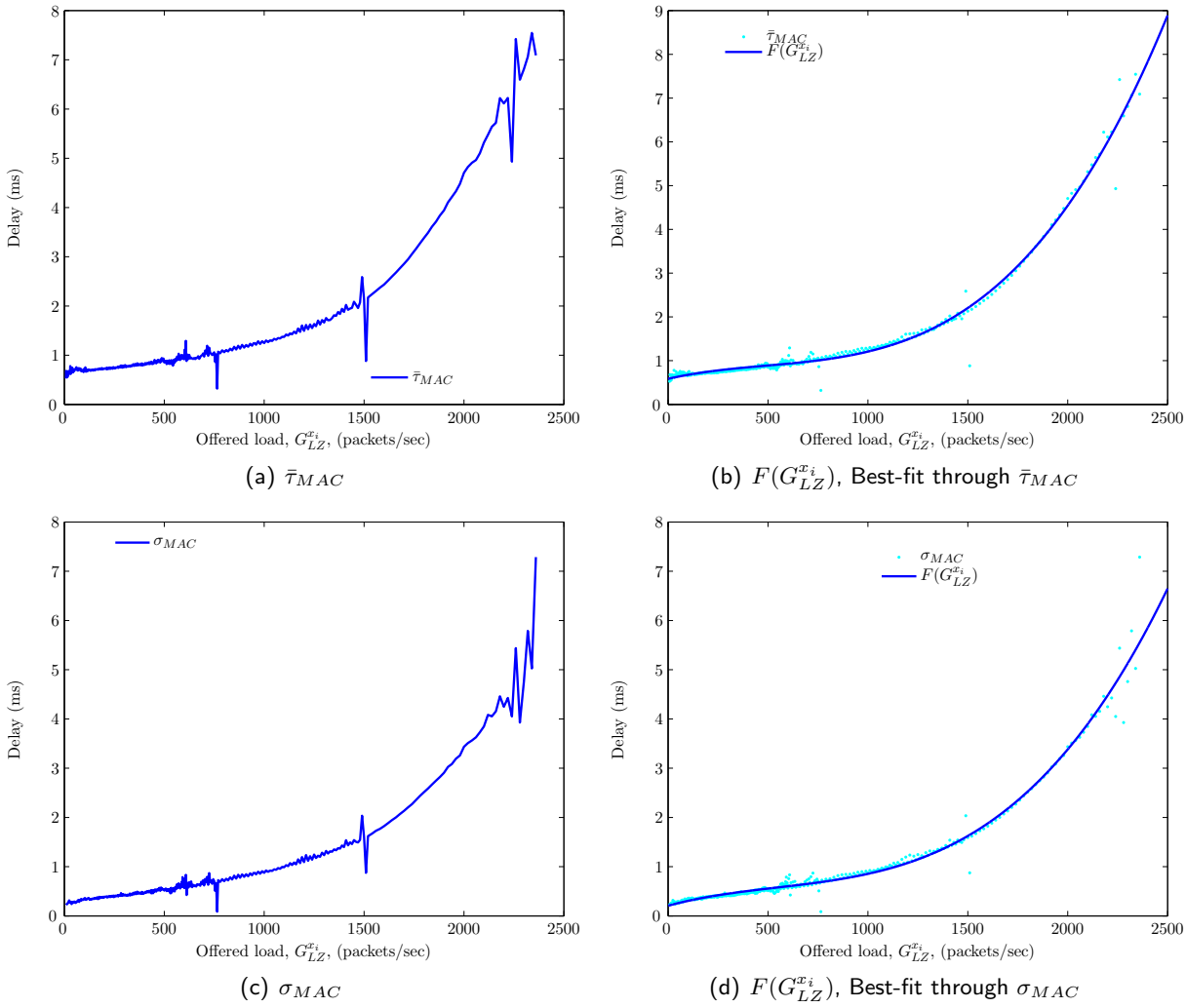


Figure 6.7: Ratio of mean and SD of MAC delay



(a) $\bar{\tau}_{MAC}$ (b) $F(G_{LZ}^{x_i})$, Best-fit through $\bar{\tau}_{MAC}$

(c) σ_{MAC} (d) $F(G_{LZ}^{x_i})$, Best-fit through σ_{MAC}

Figure 6.8: Best Fit through Mean and STD of MAC access delay

6.4.3 Mean and SD of MAC Delay

In order for each node to be able to adapt t_{def} as a function of the number of neighbours and packet interarrival rate within its LZ, there is a need to be able to predict $\bar{\tau}_{MAC}$ and σ_{MAC} empirically. This can be determined as a function of the free parameter $G_{LZ}^{x_i}$ from the data set for τ_{MAC}.

Using the method of Least-Squares-Polynomial-fit through $\bar{\tau}_{MAC}$ and σ_{MAC} versus $G_{LZ}^{x_i}$, it is found that in both cases the data can be best described by a third order polynomial equation over the range of values of the free parameter $G_{LZ}^{x_i}$ spanned by the empirical data as shown in Figures 6.8(b) and 6.8(d) respectively.

Therefore, it is now possible to calculate values of τ_{MAC} and σ_{MAC} given values of $G_{LZ}^{x_i}$ using the third order polynomial equations shown in equations (6.2) and (6.3) respectively.

$$\bar{\tau}_{MAC} \cong a_3 {G_{LZ}^{x_i}}^3 + a_2 {G_{LZ}^{x_i}}^2 + a_1 G_{LZ}^{x_i} + a_0 \tag{6.2}$$

$$\sigma_{MAC} \cong b_3 {G_{LZ}^{x_i}}^3 + b_2 {G_{LZ}^{x_i}}^2 + b_1 G_{LZ}^{x_i} + b_0 \tag{6.3}$$

where:

$a_3 = 8.286e^{-13}$, $a_2 = 1.1492^{-9}$, $a_1 = 9.7012e^{-7}$, $a_0 = 0.62012^{-3}$
$b_3 = 5.5032^{-13}$, $b_2 = -4.7352^{-10}$, $b_1 = 3.861e^{-7}$, $b_0 = 0.53468^{-3}$

6.5 Summary

The literature survey on the operation of IEEE 802.11 in broadcast mode established that, to the best of our knowledge, no similar studies have been performed which characterise the distribution of MAC delay for broadcast traffic in the vehicular environment, under non-saturated conditions. Therefore, in order to implement the adaptive deferral delay concept for message forwarding in the DDF protocol, this research characterised channel access delay statistics in dynamic topology scenarios, representative of vehicular networks on highways. The ensemble averaging over all stochastic variables cannot be performed analytically, therefore this research resorted to numerical simulations based on the exchange of beacon messages between 1 hop neighbours.

It has been observed that the MAC channel access delay is a stochastic variable, implicitly dependent on the number of actively transmitting neighbours and that the distribution of MAC delays can be approximated adequately for the purposes of this research by a half-Gaussian distribution.

Although, the analysis of the MAC delay distribution was based on the exchange of enhanced beacon packets (i.e. size of the beacon packets approximate that of the warning dissemination packets), one could assume that larger packets would alter the numerically observed channel medium access delay statistics. Moreover, this research takes into account the relative sizes of data packets to beacon packets in equation (4.27) which enables $G_{LZ}^{x_i}$ to be computed correctly, based on the results of the MAC delay calibration analysis. However, more extensive simulations are required to characterise the channel medium access delay statistics and varying traffic demands over different bandwidths to ensure robustness. However, this will not be pursued any further since the delay statistics covered in this chapter can be used to demonstrate the efficacy of the approach proposed by this research (i.e. this is a "proof of concept" rather than a properly engineered solution).

CHAPTER 7

PROTOCOL SIMULATION AND EVALUATION

7.1 Introduction

In this chapter a series of simulation experiments are executed to evaluate the performance of the DDF protocol against a simple flooding protocol and a comparable data dissemination protocol. The aim of the evaluation is to compare how efficiently the mechanisms of each protocol perform in successfully disseminating a warning message throughout an application-defined data dissemination area (*DDA*). For this reason, this research compares performance over an non-saturated (data) network with one event-driven message dissemination flow within the *DDA* at any one instance, in order to analyse protocol data delivery performance. Each vehicle within the simulation is assumed to be equipped with VANET technology since this represents a worst case scenario which pushes the network towards congestion. However, the underlying mechanism which disseminates the warning message in the scenario for sparsely connected networks is the same mechanism which tests a network with a smaller penetration of equipped vehicles. Therefore, such a scenario is covered when protocol performance is evaluated over low traffic flow rates which exhibit sparsely connected network behaviour.

7.2 Simulation Environment

As mentioned in §5.3, there are 3 entities (vehicle traffic simulator, network simulator and post processing filters) which form the simulation and evaluation platform and which enable the DDF protocol, along with the benchmark comparison protocols, to be implemented and assessed. This following section summarises the main points of each simulation entity and reiterates parameter settings where necessary.

7.2.1 Network Model

The DDF protocol is implemented using the OPNET simulator, as discussed in §5.5.1, and the IEEE 802.11b protocol is used as the MAC layer simulation model. The standard settings for IEEE 802.11b are detailed in Table 5.3 of Chapter 5 for a nominal bit rate of 11 Mbps with a transmission range of 300 m. The radio propagation model employs a two-ray ground model, with an omnidirectional antenna model. All the physical layer parameters are set prior to simulation and are listed in Table 5.2. All data warning dissemination packets for each protocol are 250 bytes in size and the beacon packets, in the case of DDF, are 100 bytes in size.

7.2.2 Mobility Models

In order to evaluate protocol performance the underlying mobility of the nodes must reflect realistic traffic flow dynamics. As detailed in §5.4.4 the output of the microscopic traffic flow simulator, FlowSim, is coupled to the network simulator in order to provide the trajectory for each corresponding node. The simulation geometry consists of a closed loop traffic network, which is approximately 16 km in length, with two lanes of traffic flow in both directions; overtaking and lane changing are allowed.

In order to assess how well each protocol scales with varying vehicular traffic flow dynamics, from highly congested stop-start traffic, through to fast flowing sparsely connected road traffic networks, five different traffic flow rates are used in the evaluation simulations. The traffic flow rates used in the simulations are 549, 822, 1094, 1376 and 1658 veh/lane/hr. The details of the traffic simulator settings are shown in Table 5.1.

7.2.3 Simulation Scenario

The protocol evaluation is carried out using the hypothetical scenario shown in Figure 7.1 where a vehicle has crashed into the central reservation barrier. The scenario requires that all nodes within a region covered by the *DDA*, in all lanes of both traffic flows, are warned about the incident. The source node, therefore, lies within the interior of the *DDA* requiring two forwarding directions and hence two forwarding flows stemming from the source node which disseminate the message throughout the region towards the boundaries of the *DDA*.

With more than one forwarding direction, there is a high probability that the first forwarding nodes (in each forwarding direction on either side of the source node) rebroadcast the message at the same time. This will result in collisions around the region of the source

node as a result of the hidden terminal problem (forwarding nodes are unlikely to be within transmission range of one another), leading to delays in the initial propagation of the message and unnecessary retransmissions. In order to avoid this problem the DDA is divided into two sub-areas[1], DDA_1 and DDA_2 and address a message to each sub-area separately. The source node randomly chooses a sub area to which the message is broadcast first and after a slight delay $\tau_{S_{jit}}$ the message is broadcast and addressed to the remaining sub-area.

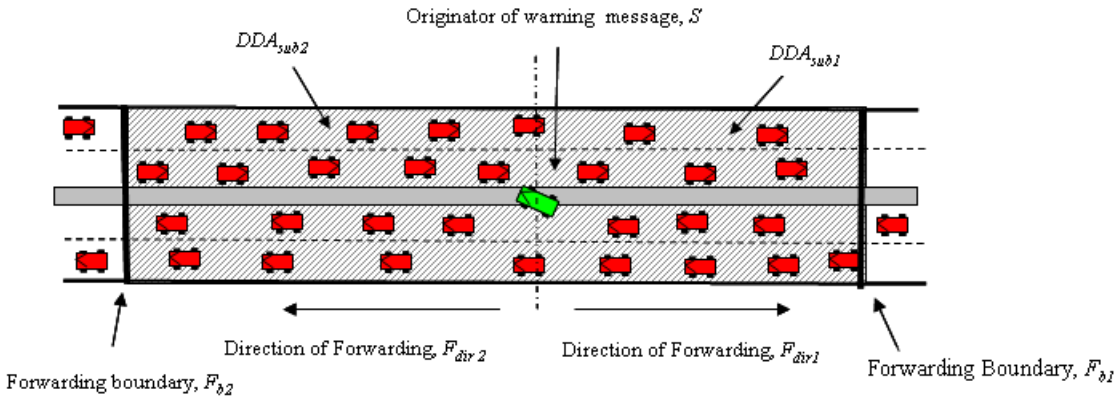


Figure 7.1: Simulation scenario

Simulation Termination

The time it takes for a message to propagate throughout the DDA and reach the forwarding boundaries will be variable and depends largely on the characteristics of the vehicular traffic dynamics within the DDA. Since the simulation run time is variable, the termination of the simulation run is scheduled to occur automatically when the message has reached the forwarding boundaries.

This functionality is implemented when a transmitting node within one of the sub-areas determines that it is within range of the forwarding boundary and sets a simulation end flag ($SimEndDDA_iFlag$) for that sub area. When both flags are set the simulation is scheduled to end at $t_{SimEnd} = t + t_{offset}$, where t is the instance in time when both simulation end flags are set and t_{offset} ensures any transmissions from the last transmitting node have been received and processed.

[1]Depending on the particular road network, e.g. a junction, may require the DDA to be divided into further subareas.

7.3 Evaluation Comparison Protocols

Within the VANET research community e.g. [18, 55, 65, 74] there is not a commonly used benchmarking protocol against which vehicular dissemination protocol performance is evaluated. Therefore, in order to evaluate and compare the performance of the DDF protocol, this research uses a baseline flooding protocol, (which provides a worst case benchmark in terms of overhead) in addition to ODAM [55] which is a basic time-deferral-based protocol employing similar distance deferral based dissemination design principles as DDF.

Each of the performance comparison protocols were implemented in state transition diagrams (STDs) using OPNET Modeler. Each of the protocol STDs were implemented in the IVC routing module of the IVC Node Model in OPNET Modeler as shown in Figure 5.6.

7.3.1 ODAM

The ODAM protocol proposed by Benslimane in [55] uses mechanisms which bear the closest resemblance to those employed by the DDF protocol. For this reason, ODAM has been chosen to provide a performance comparison. However, there are also significant functional differences between ODAM and DDF.

Operation of ODAM

The ODAM protocol was reviewed briefly in §3.1.1. This section describes its functionality in more detail in order to provide a better understanding of its operation.

In [55] the authors present a protocol for the VANET environment which disseminates alarm messages between vehicles within restricted zones called risk zones. Relay nodes are used to rebroadcast the message periodically and are chosen according to distance from the sender. ODAM does not include functionality for the exchange of beacon meaages and, therefore, local connectivity information between neighbouring nodes is not maintained. Subsequently, each node receiving the alarm message from the sender node must defer rebroadcasting according to its distance away from the sender node. A relay node is designated as the vehicle having the minimum value of computed defer time, therefore the defer time is inversely proportional to the distance from the sender node. The defer time is calculated from equation (7.1), where $D_{(sx)}$ is the distance between the sender, s and the receiving node x, R is the transmission range, max_defer_time is equal to twice the communication delay and $\varepsilon = 2$ was chosen by the authors to provide a uniform distribution of the defer time in $[0, max\_defer\_time]$.

$$defertime(x) = max\_defer\_time\frac{(R^{\varepsilon} - D^{\varepsilon}_{sx})}{R^{\varepsilon}} \tag{7.1}$$

The simulation scenario presented in [55] assumes an accident having occurred which affects one traffic flow only and assigns the restricted zone to the flow of traffic approaching the accident. Thus vehicles approaching the accident propagate the message to the nodes behind their current position away from the accident. Any nodes receiving the message outside the restricted zone do not participate in the dissemination process. A node determines if it is within the restricted area using position vectors based on information in the message header which includes current and previous position information of the source node, S, and the location of the accident.

The authors state that the first vehicle whose defer timer expires becomes the relay node and begins to rebroadcast the message periodically, the other nodes receiving the same message transition into message seen status. When a relay node receives a message from behind its current position it transitions into message seen status since the message is being propagated away from its current position. On the occasion that a node having previously received a message, receives it again, from a vehicle positioned in front of its current position, the relay node is assumed to have been overtaken and the nodes must repeat the rebroadcast process. In the case where multiple relay nodes occur as a result of being located equidistant to S, then this is resolved through the relay node with the lowest node identification number remaining as the relay node and the other node(s) transitioning to the message seen state.

The relay node transmits periodically in order to overcome fragmentation in the network. The periodic retransmission time for a relay node is chosen to allow a vehicle entering the risk zone enough time to break on the occasion that the relay node is stationary. In addition, the retransmission time ensures that any nodes entering the transmission radius since the previous rebroadcast, receive the message before overtaking the relay node. The retransmission time is calculated from equation (7.2), where R is the transmission range, V is the maximum travelling speed for the road type and $D_{braking(V)}$ is the braking distance which was calculated in [55] to be 102 m.

$$\Delta\theta_{max} = \frac{R}{V} - \frac{D_{braking(V)}}{V} \tag{7.2}$$

Implementation of ODAM

Defer Time: The authors state that the parameter, *max_defer_time* used in the calculation of the defer time in equation (7.1) is determined from the average communication

delay. However, the authors neglect to specify the sampling period over which the mean value of this delay is calculated. This issue is overcome by implementing an additional packet count structure, ODAMPcktCntList, and storing details of all received packets. ODAMPcktCntList is maintained and operated using the same principles as described for the packet count window presented §4.4.2 for the DDF protocol. Each time a node is required to calculate a defer time, it determines the average communication delay over a time window using the packets in ODAMPcktCntList.

Position History: A receiving node determines whether it is within the risk zone using current and previous position information contained in the message header from the sender node. However, there is no mention in [55] how this functionality is implemented, how often this information is updated, and what measures are taken when the sender node is stationary. Therefore, additional functionality was implemented which allows a node to update PosHistory structure periodically according to $\tau_{PosHist}$. The updating of $\tau_{PosHist}$ begins prior to the first message being broadcast. On the occasion that a node has not moved since the last update from the sender node, the previous position is not updated. This is because the receiving node would not be able to compute the direction of the sender node (since previous and current potion information would be the same) and hence determine whether it is within the risk zone.

Defer to relay node Status: Benslimane [55] makes the assertion that the first node whose deferral timer expires automatically assumes the role of relay node and all other nodes having deferred retransmission for the same message, change their status to timer expired (i.e. message seen). In addition Benslimane does not provide any details explicitly on the functionality behind how this is achieved. Therefore, this functionality is implemented from scratch and the following implementation decisions are made:

- For each alarm message for which a node is in a position to actively participate in rebroadcasting, it adds the alert message details to the OdamDeferRelayTable and sets its status as *DEFER* and schedules a defer event according to equation (7.1).
- If on reception of an alert message a matching entry is found in OdamDeferTable and the sender of the message is in the required forwarding direction, it will cancel the defer or relay event and add the details of the message to the OdamMsgSeenTable.
- When a node's defer timer expires it rebroadcasts the alert message, changes its status in ODAMDeferRelayTable to *RELAY* and schedules a periodic relay rebroadcast according to equation (7.2).

Relay node overtaking: The node overtaking functionality is implemented as summarised above, by firstly checking if a matching entry is found in OdamMsgSeenTable. If a matching entry is located and the sender node is located in front of the receiving nodes

current position (i.e. the sender is closer to the source of the alert message), a relay node is assumed to have been overtaken. The receiving node will remove the old entry from the OdamMsgSeenTable and proceed to operate as a deferring node adding the newly received message details to OdamDeferRelayTable and setting a defer time.

Additional Functionality

The scenario used for the evaluation of ODAM in [55] does not allow all the metrics of interest to this research and indeed the protocol mechanisms to be evaluated within geographic regions around a road traffic network. Moreover, the mobility pattern used to evaluate the ODAM protocol was relatively simple and the evaluation was time limited, but did not address a specific geographic region. Therefore, in order to provide a fair comparison of ODAM against DDF, using the chosen dissemination scenario, the following additional functionality needs to be implemented in order for ODAM to function using the *DDA* concept, as described in §7.2.3:

Forwarding restrictions inside the *DDA*: In order to evaluate ODAM within the chosen scenario where the accident affects both traffic flows and all vehicles within a defined *DDA* need to be alerted, then the position of the forwarding boundary as well as the position of the source node and the current and previous position of the sender node need to be included in the message header. The restriction for ODAM which only allows nodes moving in one direction to actively forward the message is maintained. In this implementation vehicles moving away from the source node actively forward the message only. However, since the accident effects both traffic flows, the number of nodes successfully receiving messages in both traffic flow directions, needs to be determined. Therefore, using position vectors, given the data in the message header, a node firstly determines if it is inside the *DDA* and secondly whether it is travelling in the correct direction to participate in the forwarding process. If both conditions are true then the node will proceed with the ODAM functionality as normal. However, if the node is inside the *DDA* but not travelling in the active forwarding direction, it will add the message to its OdamMsgSeenTable which indicates a successfully received message. On the occasion that the node is not inside the *DDA*, the message is dropped straight away.

Termination conditions: In order to provide a fair comparison with the DDF protocol, performance is evaluated up to the boundary of the *DDA*. Functionality similar to that previously explained for DDF, where a node determines its distance away from F_b is also implemented in this version of ODAM. If when a nodes defer timer expires and $D(fb\_x) \leq R$ then it will rebroadcast the message and subsequently enter the message details in to MsgSeenTable. Otherwise, if the relay timer expires and $D(fb\_x) \leq R$ the node will

rebroadcast the message for the last time and enter the message into OdamMsgSeenTable.

7.3.2 Flooding

The flooding protocol was selected for comparison purposes since it provides a known worst case comparison in addition to providing an indication of where real partitions occurred in each simulation instance.

Operation

The operation of simple flooding begins with a source node broadcasting the message to all neighbours. Each of these neighbours in turn rebroadcast the packet to their neighbour nodes until all reachable nodes within the network have received the packet. In order to minimise issues resulting from the well known broadcast storm problem (discussed in §2.5.1) and to reduce the probability of neighbouring nodes accessing the communication channel simultaneously, a delay jitter is applied to the scheduling of broadcast packets from the network layer to the MAC layer. This slight offset allows one neighbour to access the channel first, whilst the other neighbours detect that the channel is busy.

Implementation of Simple Flooding

Forwarding within *DDA*: In the case of flooding, both directions of traffic flow participate in the forwarding process. On reception of the message, a node firstly determines if it is located within the *DDA* given its current position and location information of the source node and F_b contained within the message header. If the receiving node determines that it is inside the *DDA*, and no entry exists in FldMsgSeenTable, then the node will participate in the forwarding process. The details of the warning message are entered into FldMsgSeenTable and the packet is scheduled to be sent to the MAC layer after a jitter delay randomly selected from a uniform distribution between 0 and $T_{flood}Max$ seconds ($R[0,\ FldMaxtau_{jit}]$), where $FldMaxtau_{jit}$ is the highest possible delay interval. Otherwise, if the receiving node is either outside the *DDA* or a matching entry is found in FldMsgSeenTable, the message is dropped.

Termination at F_b: In order to provide a fair comparison with DDF the DDF protocol performance is evaluated up to the boundary F_b of the *DDA*. Functionality similar to that previously explained for DDF in §4.4.8 where a node determines its distance away from F_b is implemented. Therefore, each time a node determines that it is eligible to rebroadcast the message, it also determines its distance to F_b. If the node is within transmission range

of F_b, the termination flag is set only, since a flooding node only rebroadcasts a message once.

7.4 Evaluation Metrics

7.4.1 Spatio-temporal Filtering

In order evaluate the performance of the protocols according to the evaluation metrics presented in §7.4.2, there is a requirement to filter the results using spatio-temporal criteria. This requirement arises from the need to evaluate the performance of the protocols over the *DDA* within the time window defined from the time the source node originates the dissemination message, t_{src}, to the time it takes for the message to propagate to the forwarding boundaries, $t_{F_{b1}}$ and $t_{F_{b2}}$. All simulation trace data generated from nodes located outside the boundaries of F_{b1} and F_{b2} within the time window defined by equations (7.3) and (7.4) respectively, are excluded from the evaluation.

$$t_{DDA_1} = t_{F_{b1}} - t_{src} \tag{7.3}$$

$$t_{DDA_2} = t_{F_{b2}} - t_{src} \tag{7.4}$$

Given the variability in the dynamics of vehicular traffic flow, it is inevitable that the time taken for the message to propagate from the source node towards both F_{b_1} and F_{b_2}, respectively, will differ. Therefore, performance metrics are evaluated over the total area within the *DDA* by: firstly analysing the results within DDA_1 between t_{src} and $t_{F_{b1}}$ and DDA_2 between t_{src} and $t_{F_{b2}}$ separately; secondly, combining the results in order to determine performance over the entire *DDA*.

If any of the protocols being evaluated terminate the dissemination process prior to reaching the forwarding boundary, the performance metrics are determined up to the position and time of the receiving or transmitting node closest to F_{b1} and F_{b2}. The position and receiving time of the closest node to F_{b1} and F_{b2} is then used to determine the time windows, $t_{F_{b1}}$ and $t_{F_{b2}}$, over which the protocol is evaluated. All proceeding metrics, at this simulation instance, are evaluated up to this position and time. It is necessary to evaluate performance in this way at these particular simulation instances in order to avoid reporting incorrect results. This is because the time at which the dissemination process reaches F_b is not known prior to the start of the simulation since it is dependent on the characteristics of the vehicle traffic flow dynamics within the DDA, which are unknown.

7.4.2 Performance Metrics

The following metrics are computed from the network simulation output results using the spatio-temporal filtering technique described above.

Area Coverage Ratio

The area coverage ratio provides a measure of the percentage of the total *DDA* covered by the message dissemination process. It provides a measure of the ability of the protocol to complete the dissemination process (i.e. covering areas defined by *DDA*) without terminating prematurely, prior to reaching the respective forwarding boundaries, F_{b1} and F_{b2}.

Message Delivery Ratio

This is a measure of the ratio of the number of nodes receiving the message to the number of nodes inside the *DDA* over the time window defined by t_{DDA_1} and t_{DDA_2}. The message delivery ratio is a reliability measure of the ability of the protocol to successfully deliver the message to all vehicles entering the *DDA*.

Forwarding Ratio

The forwarding ratio is the number of nodes assuming the role of forwarding node divided by the number of nodes within the dissemination area within the time windows t_{DDA_1} and t_{DDA_2}.

In dissemination protocols employing distance deferral techniques, the forwarding node is the most efficient mechanism for coverage of the *DDA*. The forwarding node is selected to cover the most new ground by each transmission and is therefore, the node farthest away from the previous transmitting node, since it will provide the greatest additional area coverage towards F_b. The forwarding ratio provides an efficiency measure of the ability of the forwarding and deferral mechanisms employed in the protocol.

Retransmission Ratio

The retransmission ratio is the number of nodes which retransmit the message after the forwarding node has transmitted the data packet, divided by the number nodes within the *DDA* between the time window defined by t_{DDA_1} and t_{DDA_2}.

A retransmitting node is the secondary mechanism ensuring that the message is propagated successfully towards F_b. On the occasion that coverage requirements have not been met successfully, then a node will retransmit the message. The retransmission ratio provides both an efficiency and reliability measure on how frequently coverage requirements were not met by the primary forwarding mechanism.

Overhead Ratio

The overhead ratio is the total number of packets transmitted divided by the number of nodes successfully receiving the message within the *DDA*. The overhead ratio provides a measure of how efficiently a protocol scales with an increase in node density.

Partition Count

The partition count provides a measure of the mean number of times network fragmentation occurred in the network causing a partition between the forwarding boundary and the node closest to the forwarding boundary.

Partitions Overcome

The count of partitions overcome provides a measure of the ability of the protocol to overcome network fragmentation and successfully continue the message dissemination process towards F_b.

Coverage Delay

The coverage delay is the time it takes each protocol to reach F_b or in the case of early termination, the node closet to F_b. The simulation instances at which partitions were recorded to have occurred have been excluded from this analysis. This is because "reconnection time" is dominated by the underlying vehicle mobility rather than the protocol properties and therefore this research is interested in comparing the mean propagation time for a fully connected network for this metric.

7.5 Independent Variables

The results of the simulations reflect the impact of the changing dynamics of traffic flow and varying vehicular densities on the performance of the protocols. The performance of each protocol is explored by varying the following independent (free) variables:

7.5.1 Road Network Characteristics

The dimensions of the road traffic network, the number of lanes of traffic and the directions of traffic flow all affect the performance evaluation. The choice of these variables have previously been discussed in Chapter 5 and are summarised in Table 5.1.

7.5.2 Vehicle Traffic Flow Rates

The vehicle traffic flow files enable varying mobility profiles to be evaluated from sparsely connected networks where a protocol's performance at handling and overcoming partitions can be evaluated, through to highly congested networks where the efficiency of each protocol to cope with increased channel access demand can be assessed. The vehicle traffic flow rates were discussed in depth in §5.4.4 and the traffic flow rates are listed in Table 5.1.

7.5.3 Size of the DDA

In order to evaluate how efficiently the protocol scales with an increased size of *DDA*, the size of the *DDA* is kept as a free parameter. However, for each range of *DDA* sizes it is required to average over realisations of the underlying mobility pattern in order to capture the varying dynamics of vehicular traffic for each traffic flow rate. Therefore, the originating node (the centre of the *DDA*) is positioned at various locations around the road traffic network, and at each of these locations the simulation is run at different times.

7.5.4 Protocol Parameters

DDF Parameters

The DDF protocol has a collection of parameters which are required to be set prior to running the simulation. Some of the parameter settings have previously been discussed in the text and are, therefore, not reported again in this section. The remaining parameters are listed in Table 7.1. The parameters listed in 7.1 are tunable and are set at optimum values (previously investigated through simulation trials) in order to maximise efficiency and reliability against reduced overheads.

Parameter	Value
Beacon Frequency, T_{beacon}	20 ms
Max number of Beacon misses, Max_{nbr_miss}	1
Access delay jitter, δ_{nbr}	0.8
Max forwarding node retransmissions, $ForRetrxMax$	2
Max intermediate node retransmissions, $IntRetrxMax$	1
Max erroneous retransmission, $SuppMax$	2

Table 7.1: DDF parameter settings

ODAM Parameters

The parameter settings in the case of ODAM are listed in Table 7.2.

Parameter	Value
Max road speed, V	30 m/s
Position history update, $\tau_{PosHist}$	0.5 s
Relay node retransmission delay	1.66 s

Table 7.2: ODAM parameter settings

Flooding Parameters

In the case of the flooding protocol only the maximum jitter delay, $FldMax\tau_{jit}$, is required to be set. For the purposes of this research $FldMax\tau_{jit}$ was set to $10ms$. The value of $FldMax\tau_{jit}$ was chosen as a suitable mean value from the plot of maximum MAC access delay observed in Figure 6.3.

7.6 Evaluation of Results

Each metric mentioned previously is evaluated in the following section for varying sizes of the *DDA* at different locations and times around the road traffic network. The sizes of the *DDA* are not exactly the same at each source point around the simulation geometry since a slight jitter is applied to the size of the *DDA* in order avoid systematic errors which could arise as a consequence of potential 'edge effects' when the protocol terminates at F_b. Consequently, the data is binned with respect to this free parameter (i.e *DDA* size) in order to have statistically meaningful error bars in the evaluation of the performance metrics.

For each of the proceeding investigations, each scenario is simulated with varying sizes of *DDA* and varying traffic flow rates ranging from free-flowing, sparsely connected, conditions at 549 veh/lane/hr, with increasing traffic flows of 822, 1094, 1376 veh/lane/hr to highly congested conditions with stop-start vehicular traffic at 1658 veh/lane/hr. The results reported for each metric are the results of averaging over 12 instances, in location

and time, for each size of *DDA*, with error bars denoting the maximum and minimum values recorded for each *DDA* size.

7.6.1 Area coverage ratio

In this first investigation, the robustness of the dissemination protocols is discussed in terms of area coverage ratio.

Figure 7.2 shows the results of the simulations with a traffic flow rate of 549 veh/lane/hr. From the results it can be seen that there is significant variability between the protocols in their ability to disseminate the data packet across the *DDA* successfully to F_b. DDF constantly maintains a mean area coverage ratio of 100%, whereas ODAM reaches a mean delivery ratio of approximately 95% with an increase in the size of the *DDA*. In comparison, the performance of flooding progressively deteriorates with an increase in the size of the *DDA*.

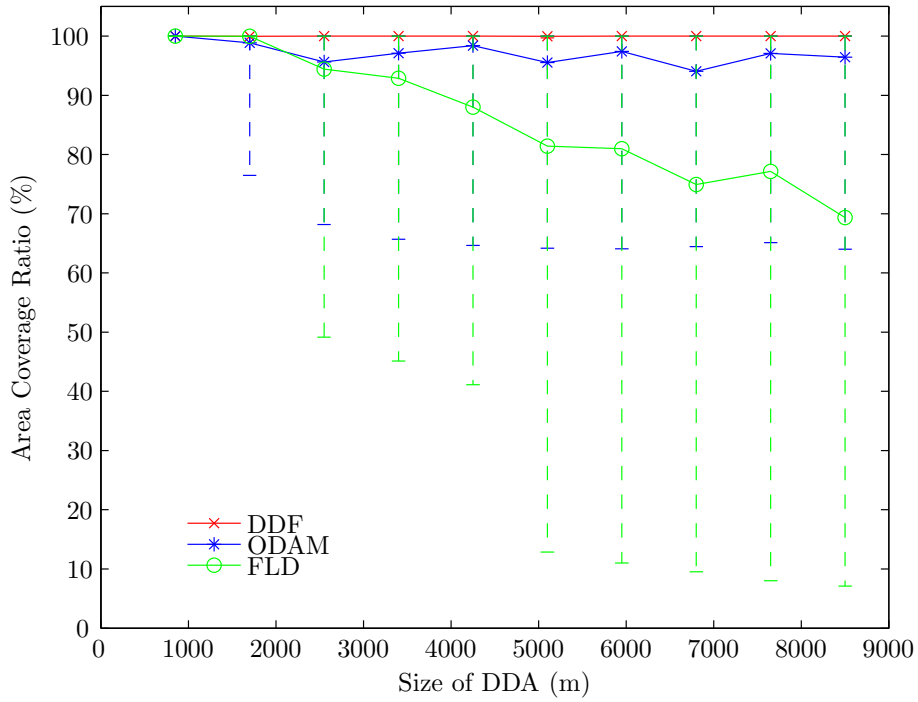


Figure 7.2: Area coverage ratio (549 veh/lane/hr)

The traffic flow rate of 549 veh/lane/hr represents a sparsely connected network in which fragmentation within the network topology of the *DDA* will occur frequently. The deteriorating performance of the flooding protocol is therefore expected in such circumstances since it does not employ a mechanism that is able to overcome network partitions. Moving the location of the *DDA* in both time and position enables such dynamics to be captured.

This is reflected in the size of the (max-min) error bars, where the worst case coverage was found to be approximately 10% in the case of flooding for *DDA* sizes above 5 km.

The error bars show that, in the worst case, ODAM was found to provide an area coverage of approximately 65%. However, on closer investigation of the simulation trace files, it was found that at certain *DDA* source points, the dissemination process was terminated prior to reaching F_b as a consequence of the vehicles receiving the data packet from the source node not travelling in the direction of F_b. Therefore, no nodes received the message that were allowed to propagate the message towards F_b. The area coverage performance of ODAM is therefore compromised by the protocol restricting the process of message dissemination to those vehicles travelling in the direction of the forwarding boundary.

Figures 7.3 and 7.4 present the results for the scenario with traffic flow rates of 822 and 1094 veh/lane/hr, respectively. From the results it can be seen that the performance of all three protocols is similar, with their mean area coverage ratio being approximately 100%. In comparison to Figure 7.2, the performance of both flooding and ODAM have improved significantly as the traffic flow rate increases and hence the vehicle density within a node's *LZ*. This is accounted for by the fact that the network connectivity is less fragmented and the frequency of nodes travelling in both directions is higher.

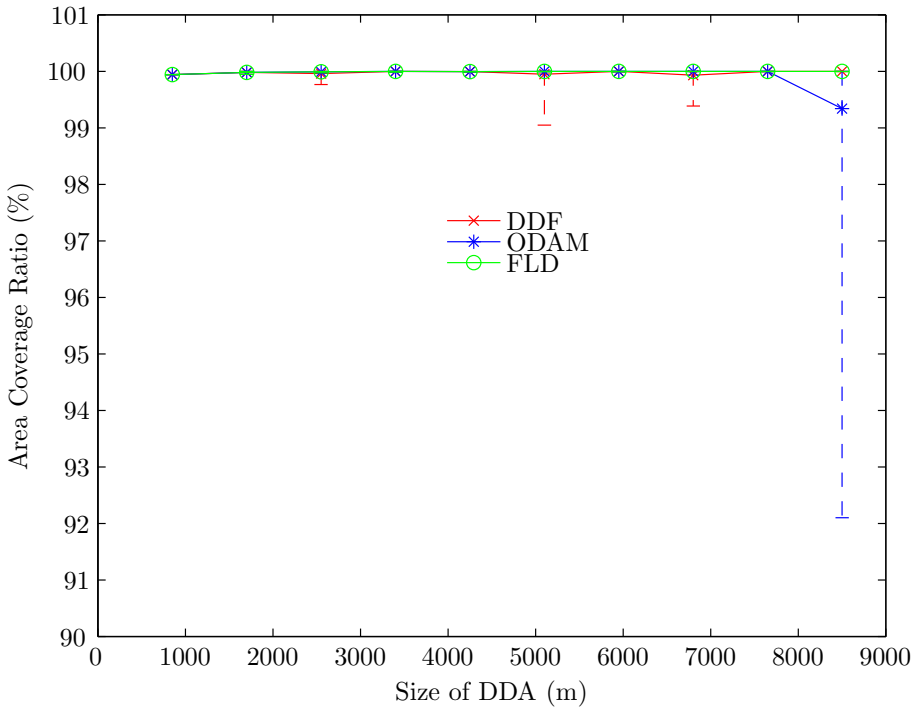


Figure 7.3: Area coverage ratio (822 veh/lane/hr)

In Figure 7.3 it can be seen that the performance of ODAM decreases slightly at the largest size of *DDA* to approximately 92%. From further investigation of the simulation

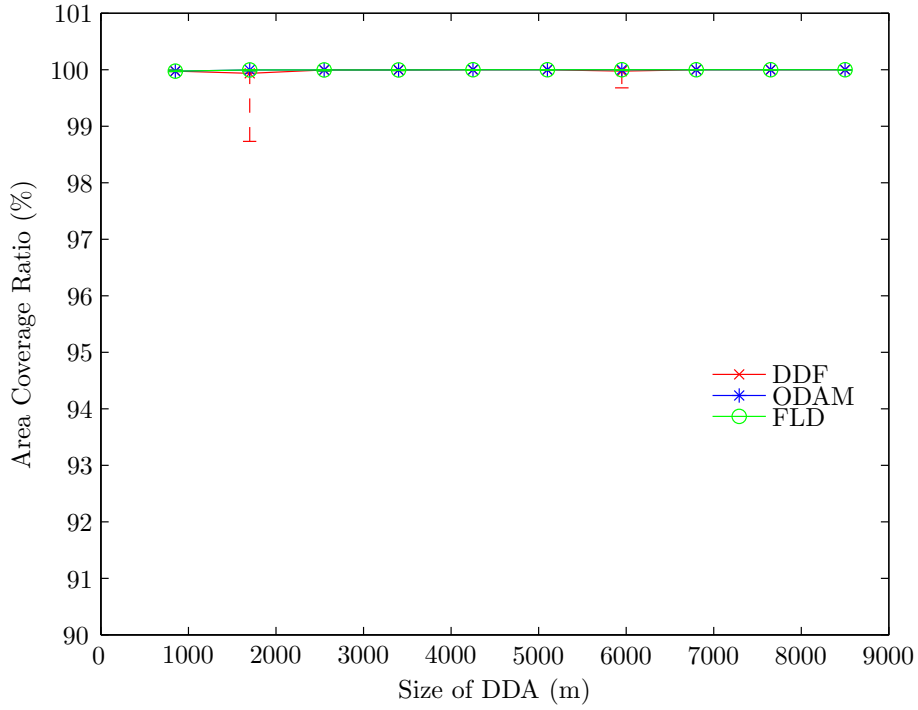


Figure 7.4: Area coverage ratio (1094 veh/lane/hr)

instances where a slight variability in the error bars for DDF can be observed in Figure 7.3, the protocol was found to terminate short of F_b as a result of the adopted retransmission rules in the proximity of F_b. Had DDF terminated prior to reaching F_b, as a result of the protocol failing, the error bars would remain at the worst case level as the size of the *DDA* increases.

Figure 7.5 shows area coverage performance with a traffic flow rate of 1376 veh/lane/hr. It can be seen from the results that flooding and DDF provide the most consistent area coverage performance at 100%. The performance of ODAM deteriorates in a similar manner to traffic conditions for sparsely connected networks, as shown in Figure 7.2. On closer investigation of the ODAM simulation trace files, ODAM was found to deteriorate, as shown by the worst case error bars, as a result of restricting the forwarding role to nodes travelling in the direction of F_b only. More specifically, the nodes receiving the message from the source node were travelling away from F_b and therefore not allowed to forward the message. As a consequence, any further dissemination of the message is terminated at the point where the node farthest away from the source node received the message. This particular instance was not observed with any other traffic flow rates and is purely a random event which could occur at any time depending on the traffic flow characteristics at the time the message was transmitted from the source node.

The area coverage of DDF can be seen to decrease slightly at a *DDA* size of approximately

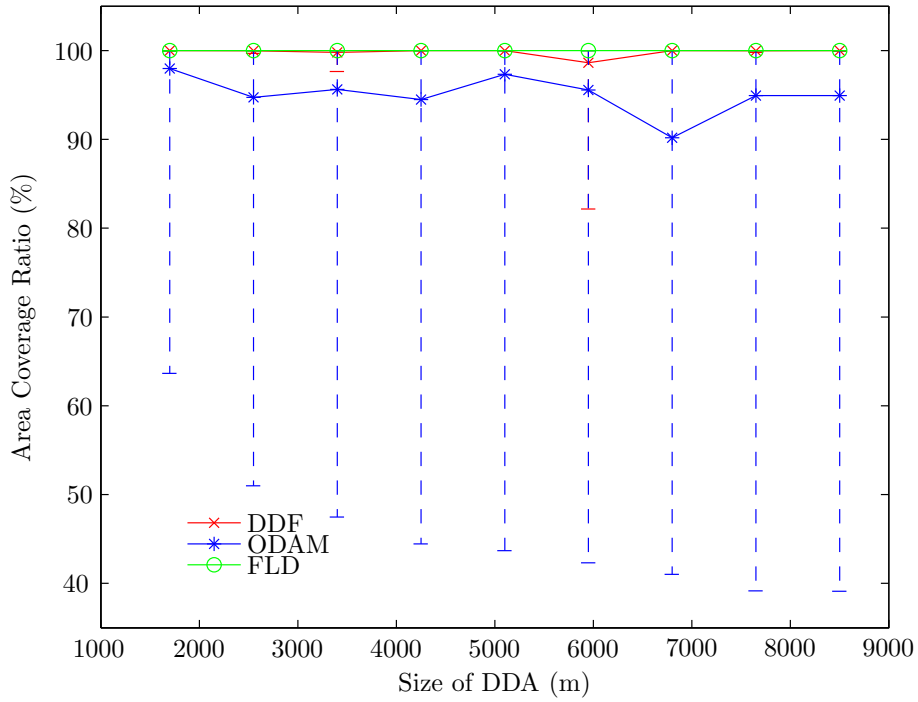


Figure 7.5: Area coverage ratio (1376 veh/lane/hr)

6 km, to a worst case value of approximately 82% in Figure 7.5. Closer investigation of the simulation trace files, revealed that this dip in performance occurred as a consequence of packet collisions and the termination rules as the forwarding chain is in proximity to F_b. More specifically, the forwarding node which would have formed the last link in the forwarding chain, x_{nlst} closest to F_b did not receive the data packet from the penultimate forwarding node, x_{nlst-1}.

From the distance deferral timing trace files, the nodes having received the data packet from x_{nlst-1}, determined that x_{nlst} is closest node to F_b and within range of x_{nlst-1} and therefore acted as intermediate nodes. However, some of the receiving intermediate nodes determined that they were within transmission range of F_b, in this instance the termination rules require that the intermediate nodes enter the data packet into the *MsgSeenTable*. Those nodes that were not within range of F_b retransmit as normal. Since x_{nlst} did not receive (and hence transmit) the data packet, the intermediate node that was not within range of F_b retransmitted the data packet. This satisfied the acknowledgment chain requirements for all intermediate nodes and the forwarding node within penultimate link. The forwarding chain, therefore, terminated at the last transmitting node and coverage is determined up to last receiving node closest to F_b. When the *DDA* is increased in size, a continuation in this performance dip is not observed since the intermediate nodes are not within range of F_b and therefore retransmit the data packet after their deferral timer expires.

Figure 7.6 shows the results for a highly congested network with a traffic flow rate of 1658 veh/lane/hr. All three protocols again achieve a consistent mean area coverage rate of 100% with the lowest variability. The high vehicle traffic flow means that the density of nodes within a node's *LZ* will be high and the speed of vehicles will be low, which minimises the probability of ODAM deteriorating as a consequence of insufficient traffic flow in the forwarding direction.

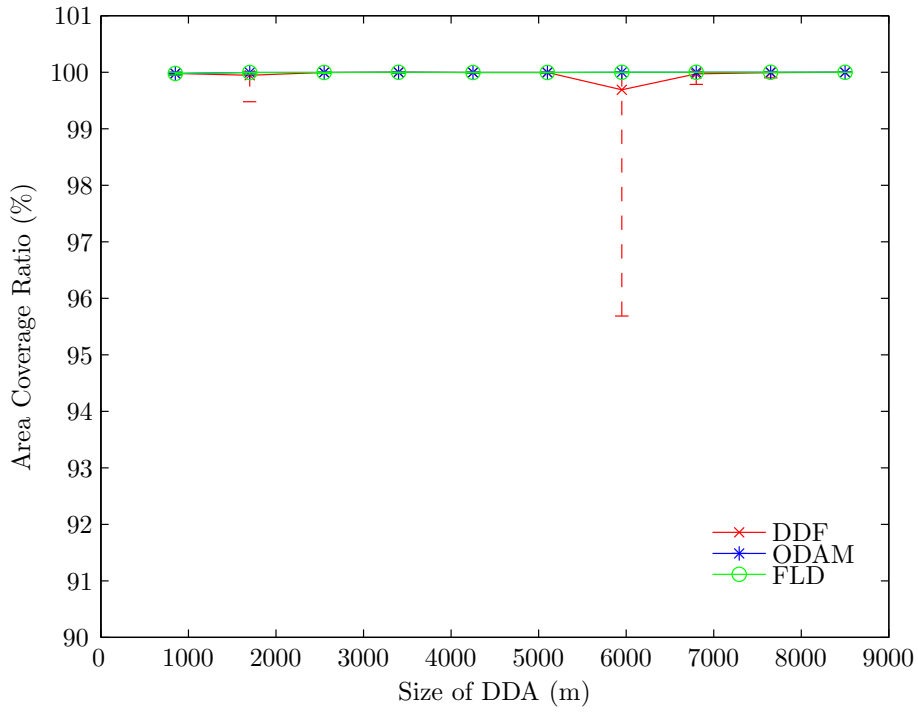


Figure 7.6: Area coverage ratio (1658 veh/lane/hr)

7.6.2 Message Delivery Ratio

Next, the delivery ratio is investigated in order to evaluate how efficiently each protocol delivers packets to the vehicles within the area covered at each of the simulation points for varying sizes of *DDA* and traffic flow rates. The delivery ratio reflects the fraction of nodes successfully receiving the data packet inside the *DDA*, as reported by the coverage ratio in Figures 7.2 to 7.6.

Figure 7.7 shows the simulation results for a traffic flow rate of 549 veh/lane/hr. From the results it can be seen that DDF consistently achieves a delivery ratio of 100% with minimal variability shown in the (max-min) error bars. Although both ODAM and flooding achieve a relatively high level of packet delivery within the *DDA*, they both perform less reliably than DDF for smaller sized *DDAs* and have a higher variability of delivery coverage as observed by the size of the error bars. In the case of ODAM the lower delivery coverage

can be attributed to the mechanism that the protocol employs in overcoming partitions in sparsely connected networks. ODAM aims to overcome partitions by broadcasting the data packet periodically. However, in sparsely connected networks vehicles move at faster speeds and, therefore, there is a higher probability that a node entering the *DDA* and passing the forwarding node may not coincide with a periodic broadcast. As discussed in §7.3.1, the periodicity in the retransmission rate of the data dissemination message for ODAM is determined by the maximum travelling speed and vehicle braking distance, which was approximately every 1.76 seconds.

Moreover, the higher variability in the delivery ratio for both flooding and ODAM can also be attributed to packet collisions occurring at the MAC layer as a result of nodes accessing the communication channel simultaneously. This occurs as a consequence of the short deferral times which are not based on knowledge of local connectivity information within a node's *LZ*. On the other hand DDF has, by design, a longer deferral time which is based on both local connectivity and channel activity within a node's *LZ* in order to avoid such occurrences.

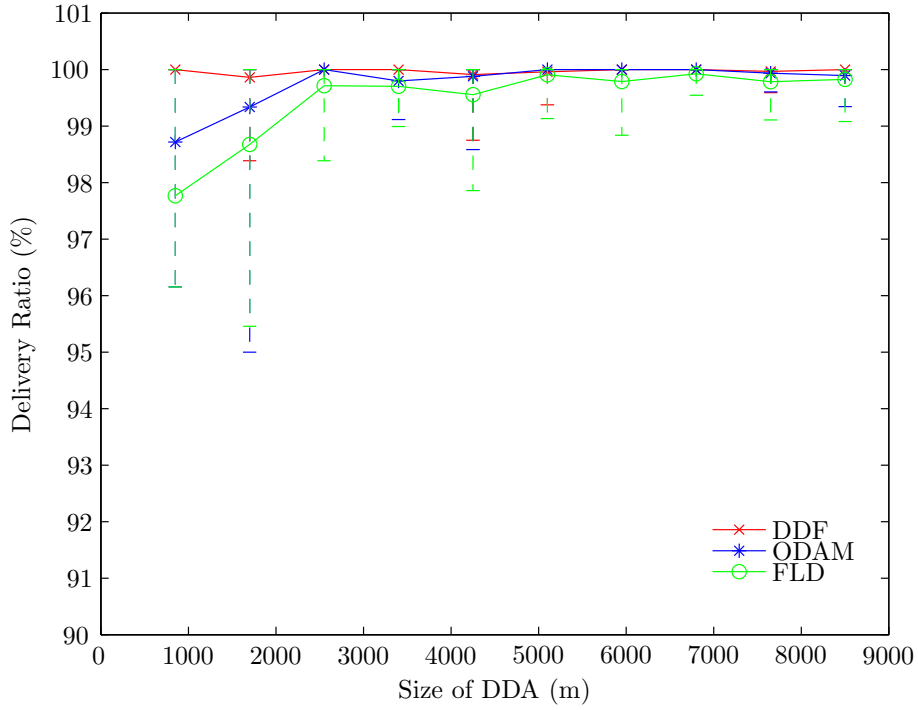


Figure 7.7: Message delivery ratio (549 veh/lane/hr)

It can be observed from Figures 7.8 to 7.11 that as the traffic flow rate increases the message delivery ratio of both ODAM and flooding is comparable with the performance of DDF, with few exceptions occurring at lower *DDA* sizes. Although DDF maintains a consistently high level of mean message delivery ratio as the traffic flow rate increases from sparsely through to highly congested networks, the level of the worst case error bars

increases as the traffic flow rate reaches highly congested levels. In particular, the lower end of the error bars for DDF shown in Figures 7.8 to 7.11, (with exceptions at 6 km in Figure 7.10 and 6 km in Figure 7.11), were found to occur as a result of the termination rules as the forwarding chain is in range of F_b.

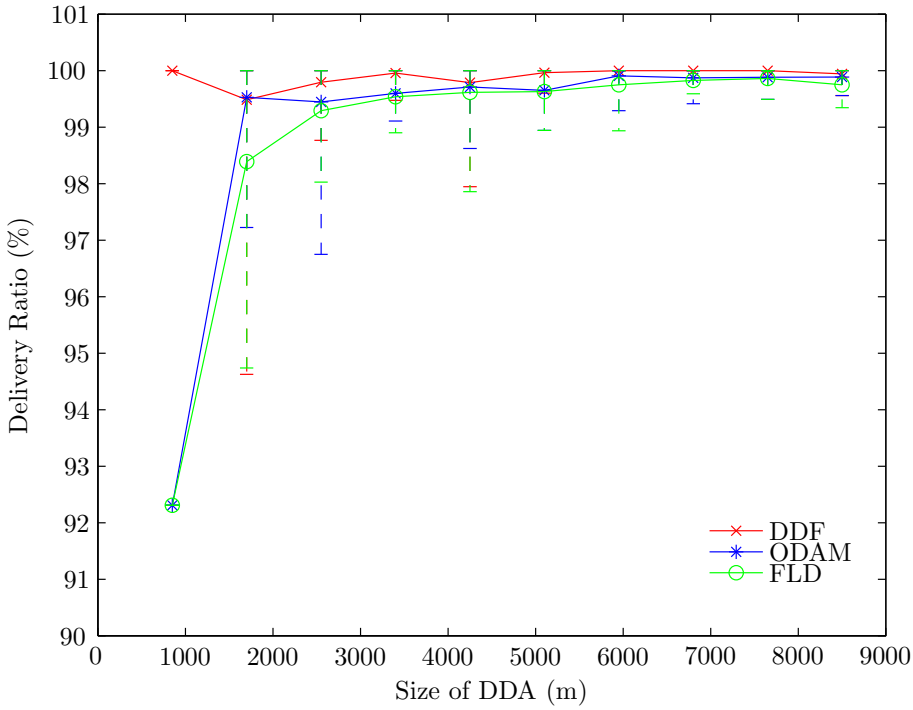


Figure 7.8: Message delivery ratio (822 veh/lane/hr). Worst case DDF error bar at 1.8 km caused by local collisions and termination rules at F_b

More specifically, investigation of the DDF simulation trace files for the instances where the lower end of the error bars occurred in Figures 7.8 to 7.11 found that the message delivery ratio deteriorated as a result of local collisions occurring between the last forwarding node and F_b, preventing nodes from receiving the message successfully. When the forwarding chain comes into range of F_b, the last forwarding node only transmits the message once in order to complete the forwarding link up to F_b and satisfy the requirements of the acknowledgement chain, the intermediate nodes do not retransmit. Therefore, any local collision which affects the nodes between the last forwarding node and F_b will not receive the message. The probability of this condition occurring is dependent on channel activity at the time of the last transmission within the LZ of the forwarding node closest to F_b, and is thus highly variable since the conditions occur randomly.

The worst case error bar for DDF in Figure 7.10 at a DDA size of approximately 6 km occurred as a result of local collisions which prevented nodes from receiving the message. Moreover, the collisions at this particular simulation instance coupled with termination rules F_b also caused a performance dip in area coverage (Figure 7.5). The reasons for the

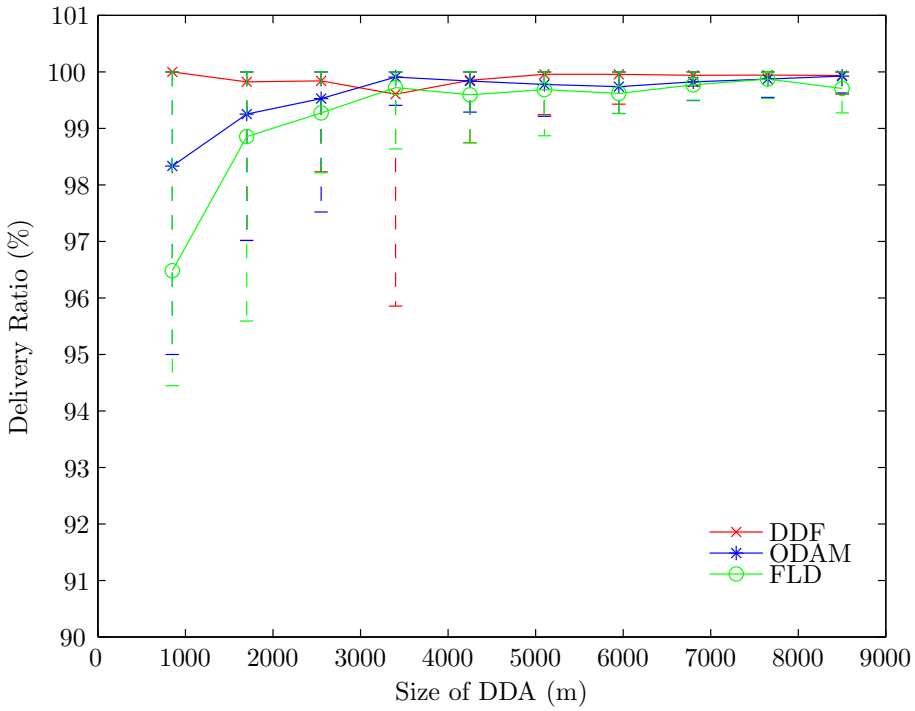


Figure 7.9: Message delivery ratio (1094 veh/lane/hr). Worst case DDF error bar at 3.5 km caused by local collisions and termination rules at F_b

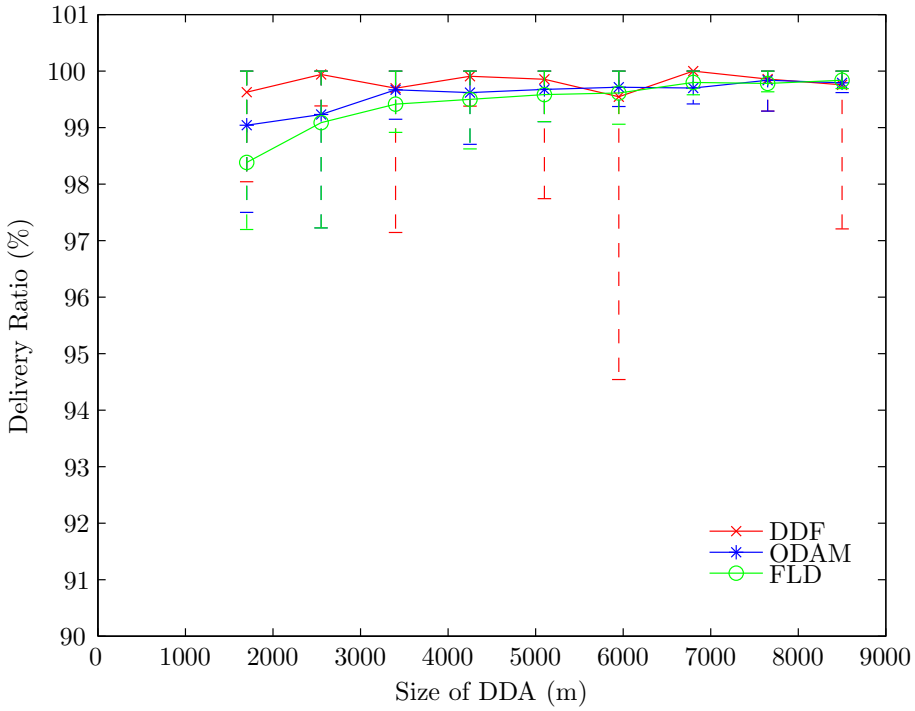


Figure 7.10: Message delivery ratio (1376 veh/lane/hr). Lower end of DDF error bar at 6 km caused by local collisions preventing nodes from receiving forwarding and acknowledgment chain messages

dip in performance have previously been discussed in § 7.6.1.

The lower end of the error bar at a DDA size of 6 km in Figure 7.11 occurred as a result of conditions which have not previously been observed at any other simulation instance for DDF[2]. The delivery ratio was reduced at this instance as a result of high network activity causing local collisions coupled with DDF termination rules at the forwarding boundary and the size of the time window over which performance was assessed.

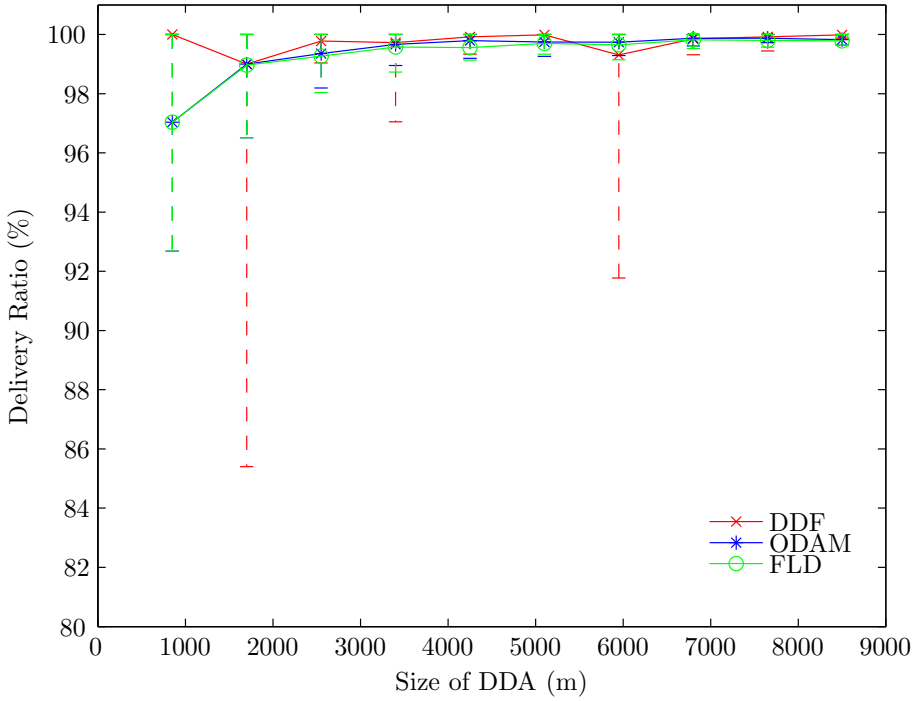


Figure 7.11: Message delivery ratio (1658 veh/lane/hr). Worst case DDF error bar at 1.8 km caused by termination rules and collisions at the boundary. Low end of DDF error bar at 6 km caused by local collisions in last two links of forwarding chain coupled with termination rules at F_b.

Investigation of the DDF trace files for this particular simulation instance found that local collisions occurring at two separate instances prevented both the transmissions from the forwarding and intermediate nodes from successfully delivering the message to a 'pocket' of nodes. More specifically, the forwarding node in link x_n transmitted the message, however, a local collision with a beacon packet prevented the majority of nodes within the forwarding region of x_n from receiving the message and the nodes in the previous link in the forwarding chain, x_{n-1} from receiving the acknowledgment. However, nodes at the edge of transmission range within x_n were not affected by the collision with M_{beacon} and received the message. The next forwarding node forming link x_{n+1} determined that it

[2]This simulation instance achieves 100% area coverage ratio and is not the same simulation instance responsible for the lower end of the error bar in Figure 7.6 at a DDA size of 6 km

was within range of F_b and transmitted the message for the last time before entering it into the MSgSeenTable. However, the transmission from the forwarding node in link x_{n+1} collided with an intermediate retransmitting node in link x_{n-1}. This caused a collision to occur again in the area of the nodes that did not previously receive the message in x_n, preventing them from receiving the message again. However, the nodes between x_{n+1} and F_b received the transmission because the collision with the intermediate node did not affect this region.

Since the forwarding chain reached F_b, the performance was evaluated up to the instant in time that the message reached F_b. Therefore, the nodes that did not receive the message as a result of collisions, were recorded as not receiving the message successfully. Moreover, the trace file for this simulation instance shows that after the time window, these nodes did in fact receive the message as a result of the forwarding node in link x_n retransmitting, after not receiving an acknowledgement that the forward chain progressed successfully. The effect of increasing the time window and hence the successful coverage of the nodes that did not receive the message, as a result of local collisions, can be seen in the longer *DDA* sizes, where the absence of the error bar indicates that all nodes received the message successfully.

Although DDF is performing at a more consistent level, variability in coverage ratio is observed to occur as the traffic flow rate increases as a result of the termination conditions at F_b mentioned above.

7.6.3 Forwarding Ratio

The efficiency of the primary mechanism for message dissemination within the *DDA* is now considered, by evaluating the forwarding ratio. The optimal theoretical forwarding ratio, $I_{x_{for}}$, for each size of *DDA* has also been included in the presentation of the results. This enables a comparison with performance of DDF, ODAM and flooding with a theoretical lower bound required in order to disseminate the data packet from the source node throughout the *DDA* to F_b.

The $I_{x_{for}}$ is determined from equation (7.5) which determines the number of hops required (in relation to the maximum transmission range), and hence optimal number of forwarding node's, to cover the *DDA* (excluding the transmission from the source node, S), where $i = \{1, 2 \ldots, n\}$, indicates the *DDA* subregion and n defines the maximum number of subregions within the *DDA*, $P_{F_{b_i}}$ the position of the forwarding boundaries and R is the maximum transmission range.

$$I_{x_{for}} = \left(\sum_{i=1}^{n} \left[\frac{|P_{F_{b_i}} - P_s|}{R_{max}} \right] \right) - n \tag{7.5}$$

The forwarding ratio for flooding will be 100% since each node receiving is also responsible for forwarding data packet. Given that the performance of flooding in terms of its retransmission ratio is known a priori, it is included here as a worst case performance comparison.

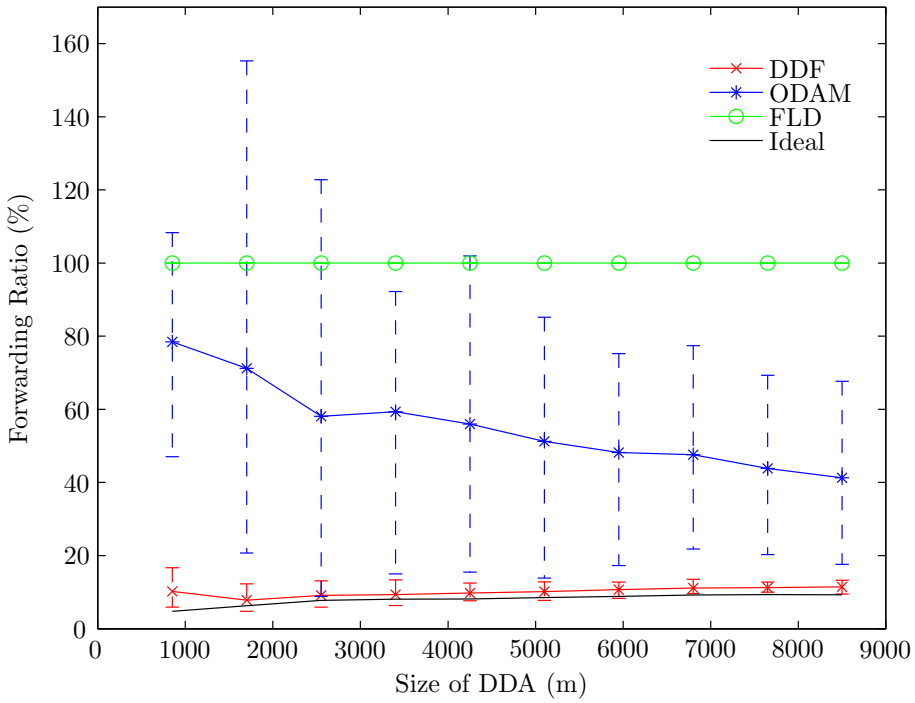


Figure 7.12: Forwarding ratio with traffic flow rate 549 veh/lane/hr

Figure 7.12 shows the forwarding ratio for a traffic flow of 549 veh/lane/hr. From the results it is observed that DDF performs the most efficiently and at a consistent forward ratio level of 10% showing that it scales well with an increase in the size of the *DDA*. Moreover, DDF closely follows the trend of the theoretical forwarding ratio, deviating by approximately 2% from the mean at each data point. Conversely, ODAM reaches a forwarding ratio of 80%, rising to approximately 40% as the size of the *DDA* increases. In the worst case ODAM approaches and exceeds flooding. At best ODAM initially appears to behave like DDF and the optimal forwarding ratio. However, the performance must also be analysed with reference to the area coverage ratio in Figure 7.2, where it can be observed that the worst case coverage of the total area for ODAM ranges from 50 to 20%. This means that only a fraction of the total *DDA* was covered and therefore fewer forwarding nodes were required up to the point where the protocol terminated prior to reaching F_b. Thus, the best case for ODAM does not actually approach the optimal level

of forwarding nodes. Moreover, the forwarding node ratio should not decrease with an increase in *DDA* size, it should either remain constant or increase slightly with an increase in *DDA*.

Consequently, in the case of ODAM, given that the area covered is only a fraction of the size of the *DDA*, fewer forwarding nodes will be required in order to cover the fraction of area covered within the *DDA*. Therefore, the 'best-case' (as mentioned above) for ODAM does not approach the theoretical forwarding ratio, and as a consequence the mean forwarding ratio deceases 'artificially' as the size of the *DDA* increases. The results for ODAM, therefore, should also be analysed whilst taking the area coverage ratio into account.

ODAM is considered to have a considerably high forwarding ratio for the following reasons: Firstly, unlike DDF where the forwarding node is determined implicitly and retransmits the message without delay, in ODAM all eligible retransmitting nodes defer retransmission. When the ODAM deferral timer expires and the forwarding requirements have not been met, a node then designates itself as a forwarding node. Through a thorough investigation of the ODAM simulation trace files, in most instances, the deferral time was found not to provide sufficient time discrimination between neighbouring nodes. Nodes are, therefore, more prone to access the communication medium at a similar time, leading to collisions and channel contention which delays the transmission and reception of the message. This means that the probability of overhearing the message prior to the deferral timer expiring is low. As a consequence, this leads to more nodes assuming that their coverage requirements have not been met and hence more nodes electing themselves as forwarding nodes. Secondly, given that ODAM restricts forwarding nodes to those travelling in the direction of the forwarding boundary only, one would expect that the worst case forwarding ratio would fall to around 50 - 60% and not follow or exceed the results for flooding. This situation is observed as a result of nodes having previously received the message, forwarding it if they are closer to F_b than the source of the transmitting node. This condition arises frequently as a direct consequence of the channel contention issues explained above. Unlike DDF, ODAM does not employ a suppression mechanism that would prevent erroneous forwarding nodes from retransmitting the data packet, which would greatly reduce the retransmission ratio.

Figure 7.13 shows the forwarding ratio for a traffic flow of 822 veh/lane/hr. DDF performs at a consistent level and closely follows the trend of the optimal retransmission range. In certain instances (e.g. Figure 7.13) DDF is performing better than the theoretical minimum forwarding ratio of the bottom error bars. This can be explained by the fact that equation (7.5) assumes static nodes when they are moving. However, the performance of ODAM is observed to deteriorate with an increase in *DDA* size and at worst exceeds that of flooding by approximately 18%. At best ODAM deviates from the ideal theoretical

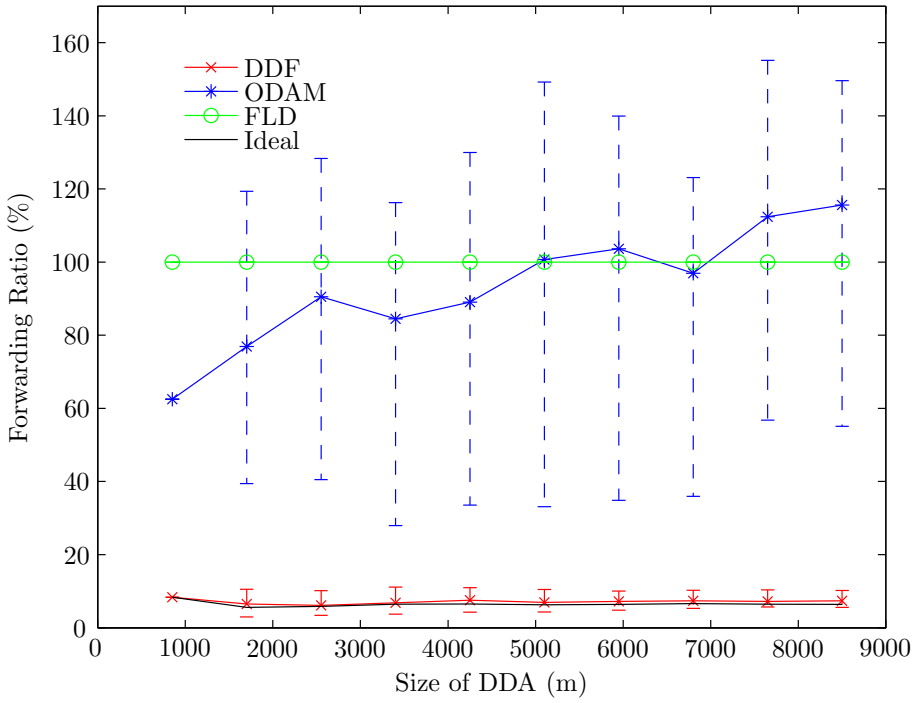


Figure 7.13: Forwarding ratio with traffic flow rate 822 veh/lane/hr

forwarding ratio by approximately 20%. The area coverage results for the corresponding traffic flow rate in Figure 7.3, show that all three protocols reached an average area coverage ratio of 100% which means that the best case performance of ODAM can be compared with the ideal forwarding ratio.

In Figure 7.13, ODAM does not follow the downwards trend in the forwarding ratio which was observed in 7.12 because a higher area coverage was achieved and therefore more forwarding nodes are used to cover the *DDA*. As traffic flow rate increases slightly and area coverage achieves approx 100% ODAM has tended to approximate flooding more closely and at high sizes of *DDA* performs worse than flooding in terms of the number of nodes actively forwarding the message.

On closer examination of the network simulator trace files for ODAM, it can be observed that the deferral timing at low levels of network activity is less than the time it takes for the processing and transmission time of the packet. As a consequence, the deferral timing for such instances does not provide time-ordered retransmission according to furthest distance away from the source of the transmission. Moreover, the deferral timing will expire at similar times for all nodes within the *LZ* of transmitting node, resulting in unnecessary retransmissions as a result of both contention and collisions at the MAC layer.

When the channel access time has increased so that the deferral time is larger than the packet processing and transmission time, time ordering according to distance is achieved.

However, in the case that a node retransmits according to the distance criteria and the channel contention time in the LZ of the next hop has a higher contention time, then the deferral timer of the nodes in the previous hop will start to retransmit. Essentially, since the deferral timing in ODAM does not allow for consideration of potential timing differences within the next hop, additional unnecessary retransmissions arise.

DDF, by design, ensures that the deferral time is always greater than the combined packet processing and transmission time, always ensuring time ordered transmission. Additionally, the DDF deferral timing includes an additional random delay which aims to minimise the effect of higher channel activity (and hence longer channel access times) in the LZ of the next hop, as described previously in the case of ODAM.

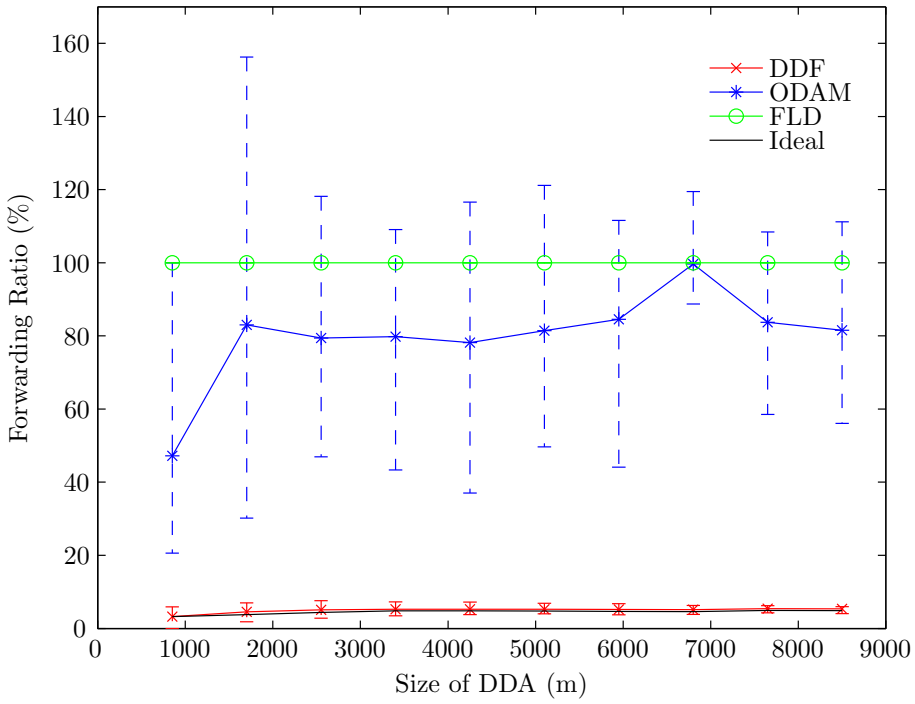


Figure 7.14: Forwarding ratio (1094 veh/lane/hr)

Figures 7.14 to 7.16 show the forwarding ratio as the traffic flow increases from 1094, 1376 and 1658 veh/lane/hr respectively. DDF can be seen to perform the most efficiently. As the traffic flow rate increases the slight difference between the optimal forwarding ratio and DDF diminishes further, with DDF performing at the optimal level in highly congested traffic. On the other hand, the performance of ODAM decreases with an increase in the traffic flow rate.

The performance of DDF scales well with an increase in the traffic flow rate and hence the node density within a node's LZ. This is achieved through the implicit decision making process, coupled with the deferral timing efficiently adapting to LZ dynamics. Moreover,

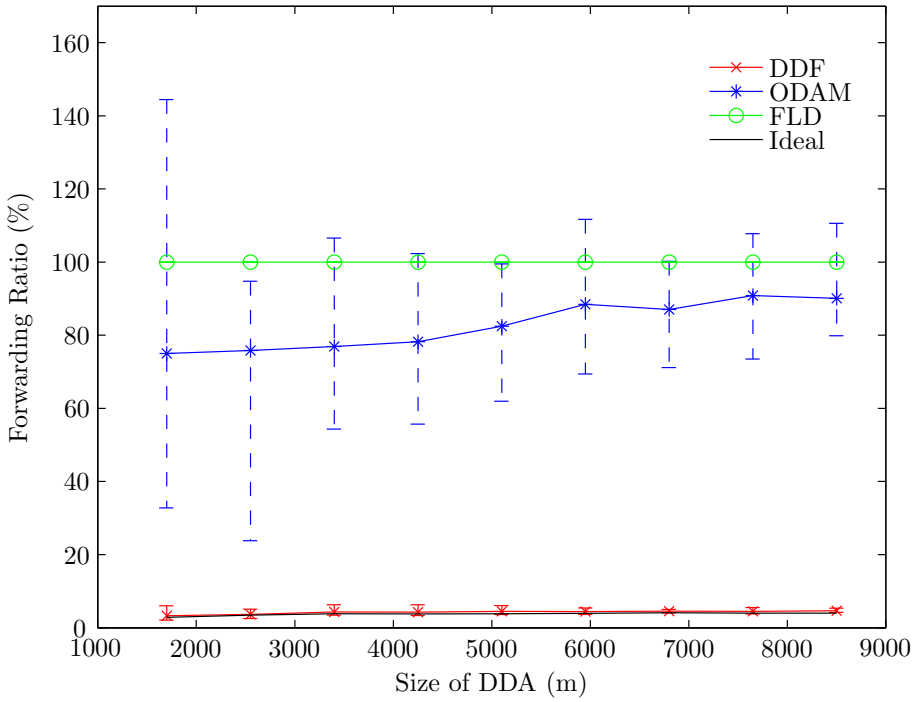


Figure 7.15: Forwarding ratio (1376 veh/lane/hr)

the design of the deferral timing calculation specifically aims to minimise the occurrence of the deferral timer expiring at the same time between deferring neighbouring nodes. Thus, ensuring time ordering according to position and taking into account the potential for deviations in contention delay within a node's LZ, reduces the number of unnecessary retransmissions and prevents increased overhead.

It can be observed from Figures 7.14 to 7.16 that the performance of ODAM decreases with an increase in node density as the traffic flow rate increases. As discussed previously, this occurs as a result of the deferral time not providing a sufficient time discrimination between nodes accessing the communication channel and in particular allowing for potential deviations between adjacent links in the chain towards the forwarding boundary. Once the deferral event expires each node will send a packet to the MAC delay which will be transmitted once the channel is sensed to be free. If during this time window the node overhears a data packet from the required forwarding direction, the same data packet will still be transmitted at the MAC layer. Therefore, as a result of the high node density within a nodes local zone the forwarding ratio will increase with an increase in the traffic flow rate.

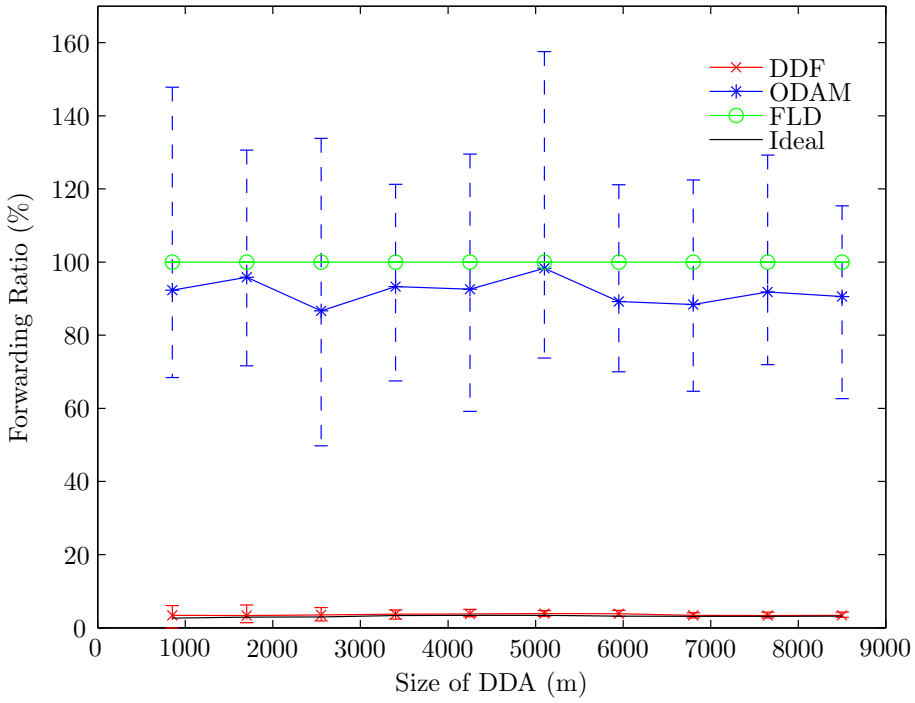


Figure 7.16: Forwarding ratio (1658 veh/lane/hr)

7.6.4 Retransmission Ratio

The retransmission ratio is is considered next, and is another efficiency measure that provides an indication of how frequently the secondary forwarding mechanism is operational on the occasion that the primary forwarding mechanism has been detected not to be progressing the forwarding chain towards F_b.

The retransmission ratio for DDF reports retransmissions from the forwarding and intermediate nodes on the occasion when the forwarding requirements have not been met; additionally transmissions required for partition handling are also counted. In the case of ODAM the retransmission ratio reports all additional transmissions by the relay nodes after the deferral transmission event has occurred and a node has transitioned to relay status. The flooding protocol is not included in this performance comparison since each node receiving the data packet forwards the packet only once, and therefore, does not detect the status of the forwarding chain and does not retransmit the data packet.

In the case of a sparsely connected network with a low traffic flow rate of 549 veh/lane/hr where partitions are encountered frequently (see Figure 7.32(a)) within the *DDA*, it is observed from Figure 7.17 that DDF has a significantly lower retransmission ratio than ODAM. In the case of ODAM, the retransmission ratio is observed to increase approximately linearly with increasing *DDA* size. In contrast, DDF maintains a relatively constant retransmission ratio of approximately 2.5% as the size of the *DDA* increases.

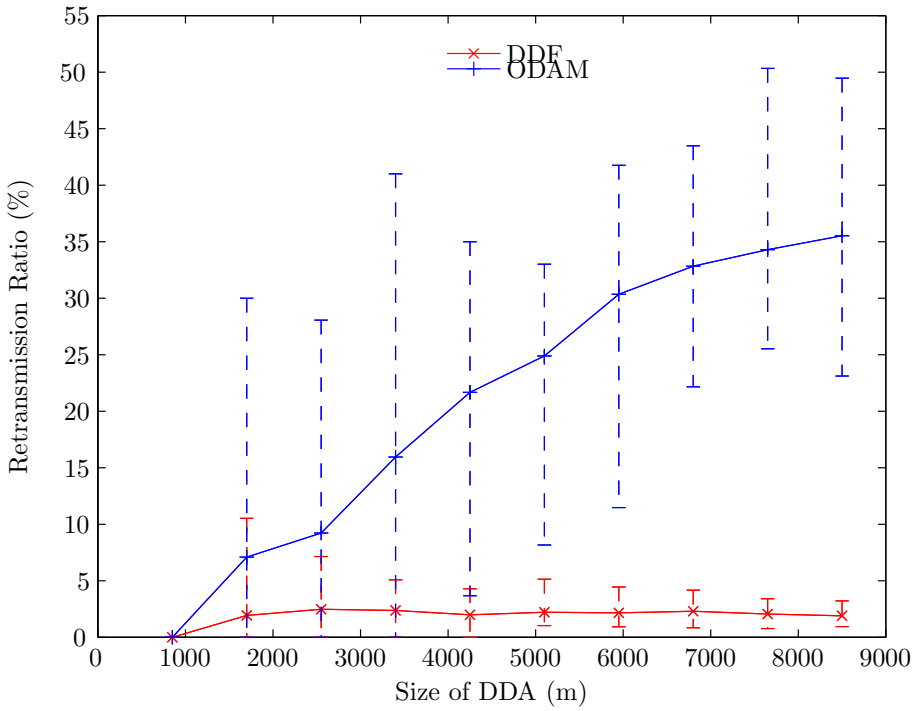


Figure 7.17: Retransmitting ratio (549 veh/lane/hr)

The performance differences between DDF and ODAM can be attributed to the mechanisms employed by each protocol in overcoming partitions. DDF incurs minimal retransmission overhead during partitions since it maintains local connectivity information and only retransmits M_{DDF} when local connectivity information detects that there is a node closer to F_b than the current forwarding nodes position. This enables the forwarding node at the head of the partition to transfer responsibility for forwarding the data packet towards F_b when it detects a node closer than its current position to F_b. Therefore, additional retransmission overhead is only incurred during the handover process and when the forwarding node retransmits the data packet to satisfy the requirement of the acknowledgement chain, prior to storing the message in its NbrWaitTable. On the contrary, ODAM incurs a higher retransmission overhead during a partition as a direct consequence of overcoming partitions by retransmitting the data packet periodically. Therefore, the longer the time window over which the partition towards F_b lasts, the higher the retransmission ratio in the case of ODAM.

As the traffic flow rate begins to rise to 822 and 1094 veh/lane/hr and the occurrence of partitions within the *DDA* decreases (shown in Figures 7.33(a) and 7.34(a) respectively), and hence the overhead required to overcome partitions in the case of both DDF and ODAM reduces. The requirement to retransmit as a result of detecting that the forwarding chain has not progressed successfully now becomes more apparent, as shown in Figures 7.18 and 7.19 respectively. This is particulary apparent in the case of ODAM for a sparsely

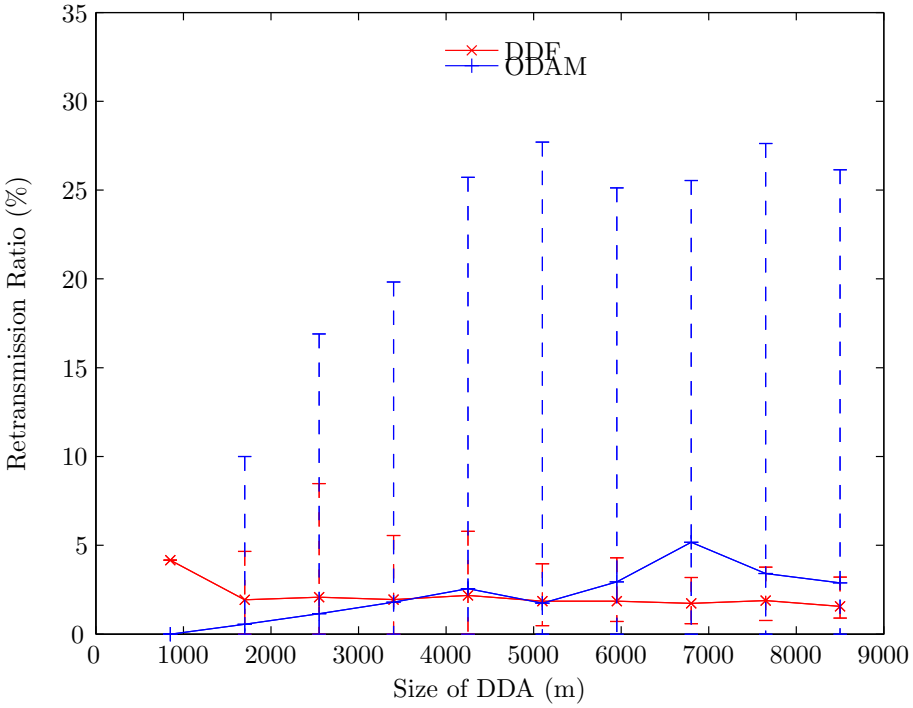


Figure 7.18: Retransmitting ratio (822 veh/lane/hr)

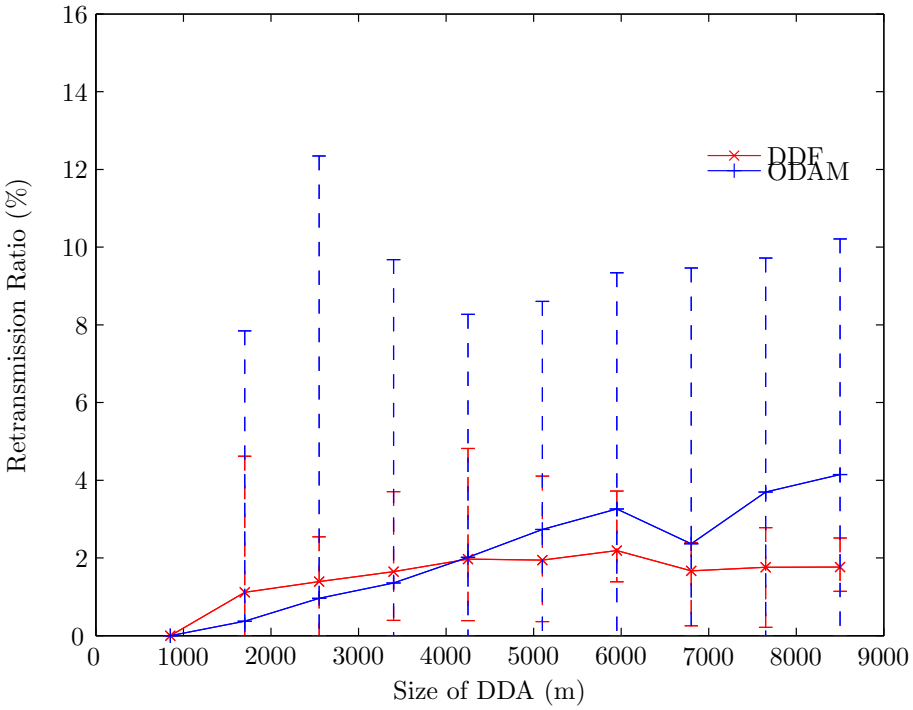


Figure 7.19: Retransmitting ratio (1094 veh/lane/hr)

connected network (Figure 7.17), where periodic retransmissions to over come partitions previously dominated the retransmission overhead.

DDF can be seen to operate at a constant retransmission ratio as both the traffic flow rate and the size of the *DDA* increase. It can be observed that, in the case of ODAM, the retransmission ratio becomes increasingly less dominated by the number of retransmissions, as a direct consequence of fewer partitions occurring. For smaller sizes of the *DDA*, ODAM achieves a lower retransmission ratio than DDF. However, as the *DDA* size begins to increase the retransmission ratio increases beyond that of DDF. The slight increase in retransmission ratio for ODAM correlates with the increase in the number of partitions, shown in Figures 7.33(a) and 7.34(a), as the size of the *DDA* increases.

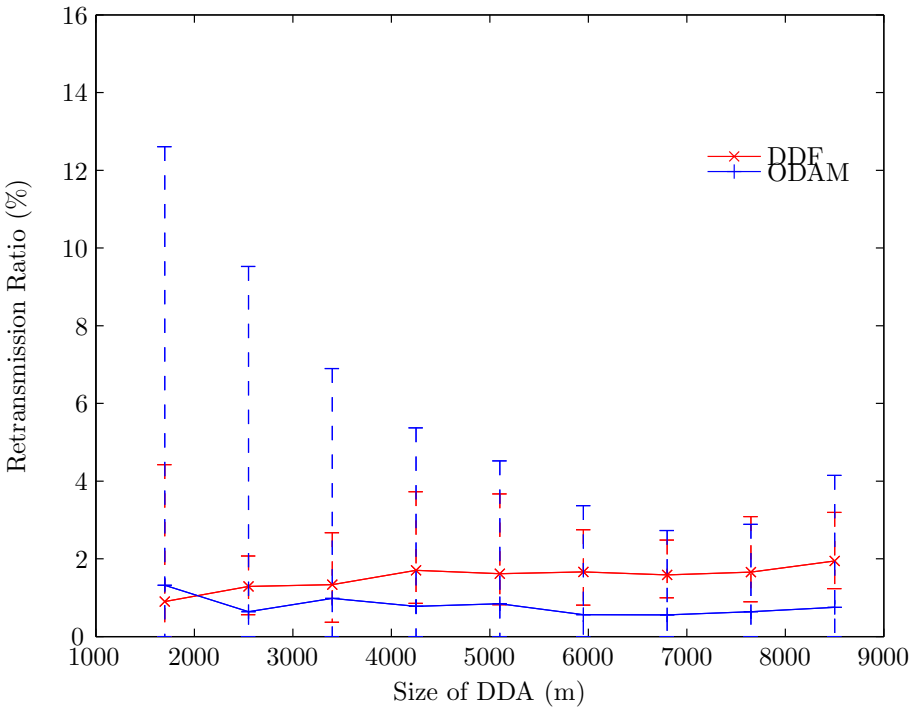


Figure 7.20: Retransmitting ratio (1376 veh/lane/hr)

It can be observed from Figure 7.20 that as the traffic flow rate increases to 1376 veh/lane/hr the retransmission ratio for DDF remains consistent as the size of the *DDA* increases. ODAM, achieves a slightly lower retransmission ratio than DDF. The consistent level of retransmission ratio correlates directly to the consistent trend in the number of partitions for the corresponding traffic flow rate for ODAM shown in Figure 7.35(a).

In the case of the highly congested network with a traffic flow rate of 1658 veh/lane/hr, it can be seen in Figure 7.21 that DDF maintains a consistent level of retransmission ratio as the size of the *DDA* increases. ODAM has a retransmission ratio of 0% up to 5 km thereafter it begins to increase slightly as a result of partitions occurring inside the *DDA*.

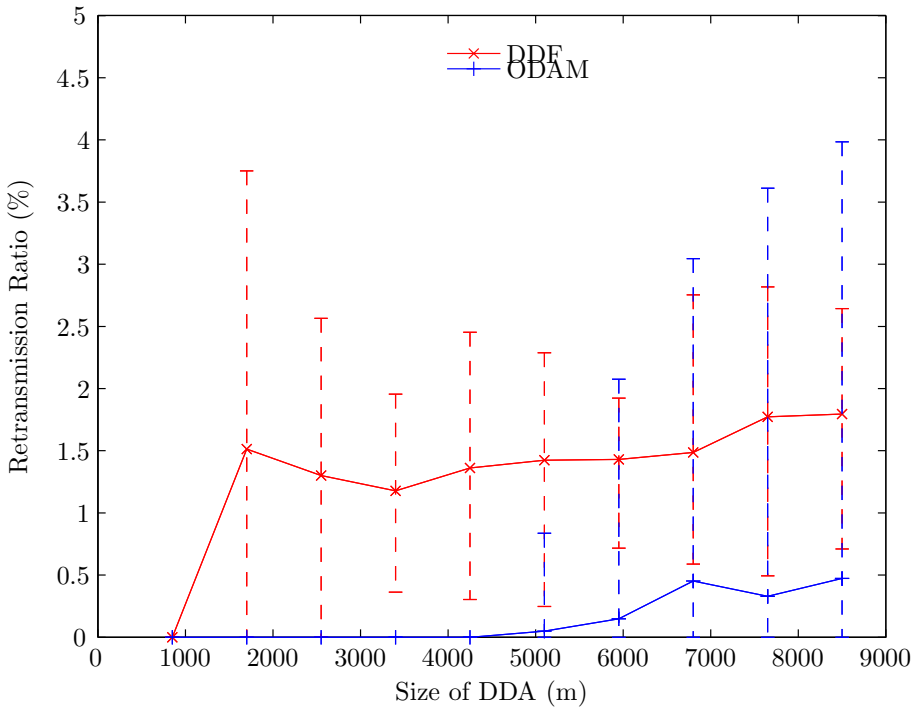


Figure 7.21: Retransmitting ratio with traffic flow rate (1658 veh/lane/hr)

It can be observed from the results that in the case of ODAM, the high retransmission ratio occurs as a direct consequence of the periodic retransmissions required in overcoming partitions. In particular, when the network is sparsely connected the retransmission ratio exceeds that of DDF in a linear trend as the size of the *DDA* increases. In comparison, DDF maintains a consistent retransmission ratio with increases in the size of the *DDA* and traffic flow rate. The retransmission ratio shows that the occurrence of coverage conditions not being met by the primary forwarding mechanism is relatively low. The additional retransmissions occurring as a result of partitions has a minimal effect on the retransmission ratio. As the traffic flow rate increases the mean retransmission ratio can be seen to decrease slightly. In the case of ODAM, it can be observed that it does not incur retransmissions as a result of coverage requirements not being met. This is because the number of nodes retransmitting as a forwarding node is high and the unnecessary retransmissions by forwarding nodes are acting like the intermediate nodes in DDF. ODAM is essentially operating like a flooding protocol as a result of the short deferral timing which generally ensures that the requirements of the acknowledgement chain are met, at the expense of unnecessary message retransmissions.

Overhead Ratio

The total efficiency of each protocol in terms of their dissemination overhead is considered next. This is determined from the total number of packets transmitted during the process of forwarding the warning message towards F_b divided by the number of successful receiving nodes, which allows the comparative performance of each protocol to be evaluated in terms of the amount of overhead each protocol generates.

In Figures 7.22 to 7.26 it can be seen that this research has achieved one of the defined goals in the design of DDF. That is, to provide efficient message dissemination using minimal retransmissions in order to disseminate the data packet towards the boundary of the *DDA*. From Figures 7.22 to 7.26 it can be observed that as the traffic flow rate increases (and hence the density of nodes within a node's *LZ* increases), the mean overhead ratio decreases progressively from approximately 14% in the case of a sparsely connected network, to approximately 4% in the case of a highly congested network. Moreover, the overhead ratio remains at a constant level as the size of the *DDA* increases for each of the traffic flow rates. The DDF protocol can be seen to scale extremely well with increases in both the size of the *DDA* and traffic flow rates, which indicates that the DDF protocol is able to cope with the varying traffic dynamics experienced on vehicular traffic networks.

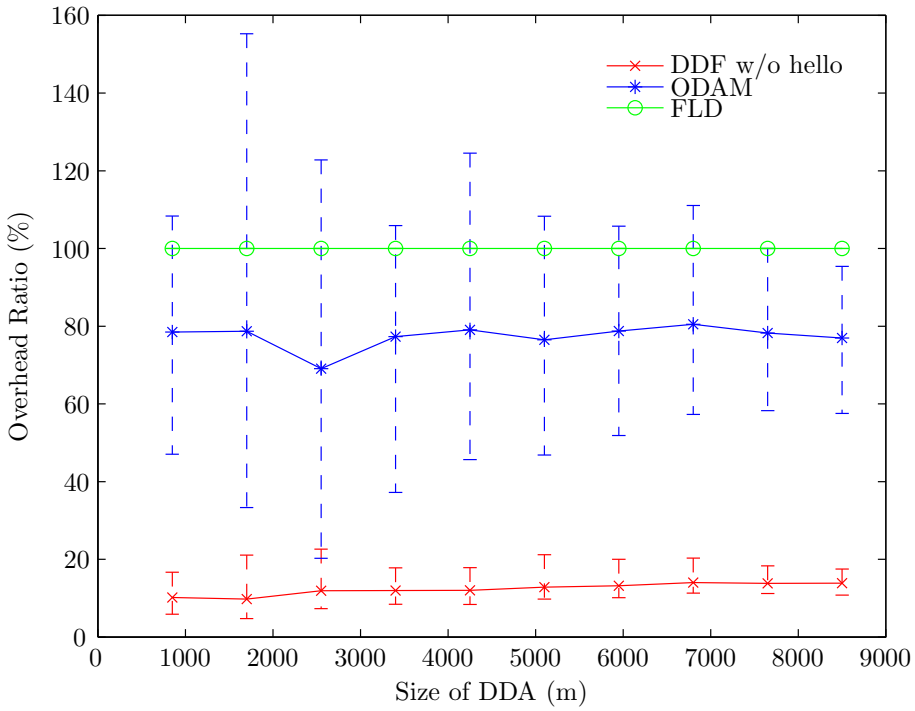


Figure 7.22: Overhead ratio (549 veh/lane/hr)

In comparison, the performance of ODAM can be seen to be closer to that of flooding, and at some instances it exhibits a higher overhead ratio than flooding. The overheads are

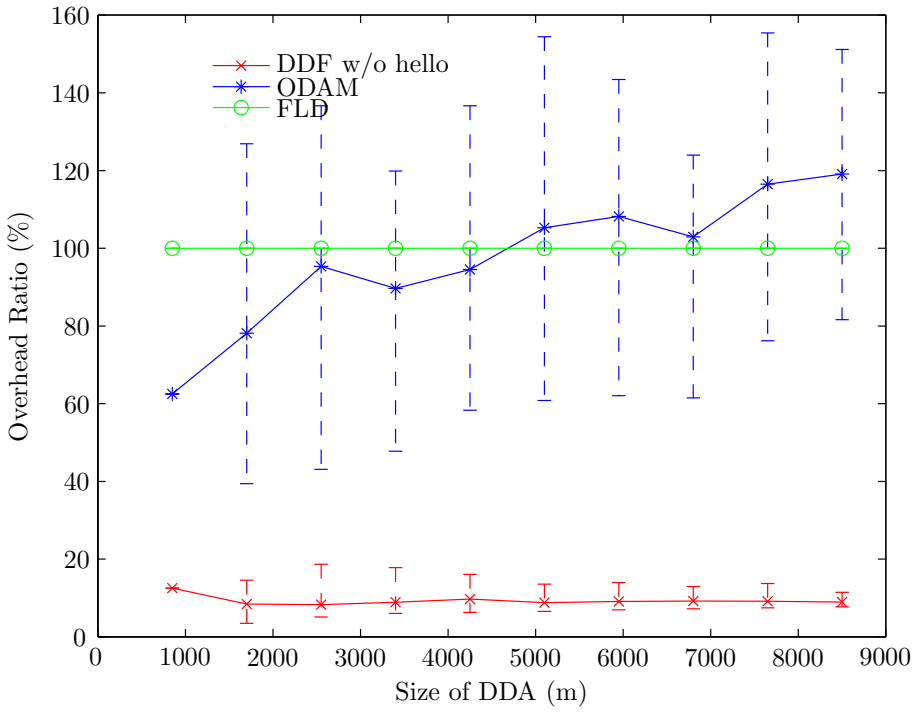


Figure 7.23: Overhead ratio (822 veh/lane/hr)

largely dominated by retransmissions from the primary forwarding mechanism as a result of the inability of the deferral timing employed in ODAM to adapt the dynamic demands within its *LZ*, as previously discussed in §7.6.3-4. As a consequence the performance trend of ODAM is largely unpredictable for lower traffic flow rates as observed in Figures 7.22 and 7.23 where the corresponding partition levels within the *DDA*, shown in Figures 7.32(a) and 7.33(a), occur more frequently. It can be observed from Figures 7.24 to 7.26 that as the traffic flow rate and hence node density within a nodes *LZ* increases, the performance of ODAM begins to deteriorate, tending to follow the flooding protocol at the highest traffic flow rate.

Overhead Including Beacon Traffic

The effect that the beacon traffic has on the overhead ratio of the comparative performance of the protocols is investigated next. Although, the beacon traffic is a supporting mechanism of the DDF protocol and the packets are a fraction of the size of the data packets it provides a comparative look at total traffic for each protocol.

Figure 7.27 shows that the total overhead ratio for DDF exceeds both flooding and ODAM as the size of the *DDA* begins to increase past 4 km. As the size of the *DDA* increases, the number of partitions and hence the size of the time window taken to cover the *DDA* from the source node to F_b increases in a sparsely connected network. The beacon traffic

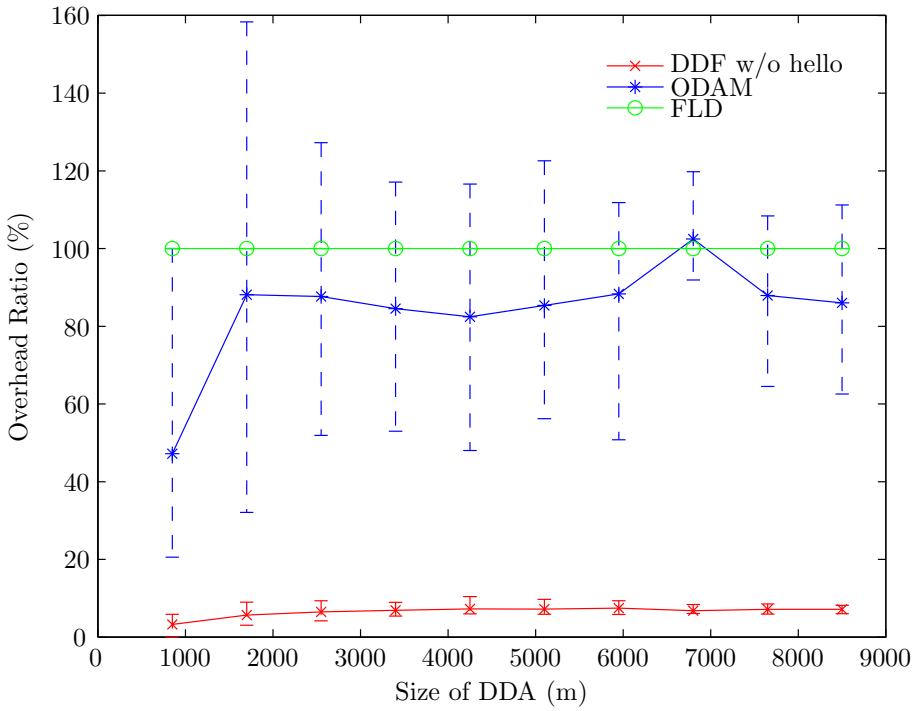


Figure 7.24: Overhead ratio (1094 veh/lane/hr)

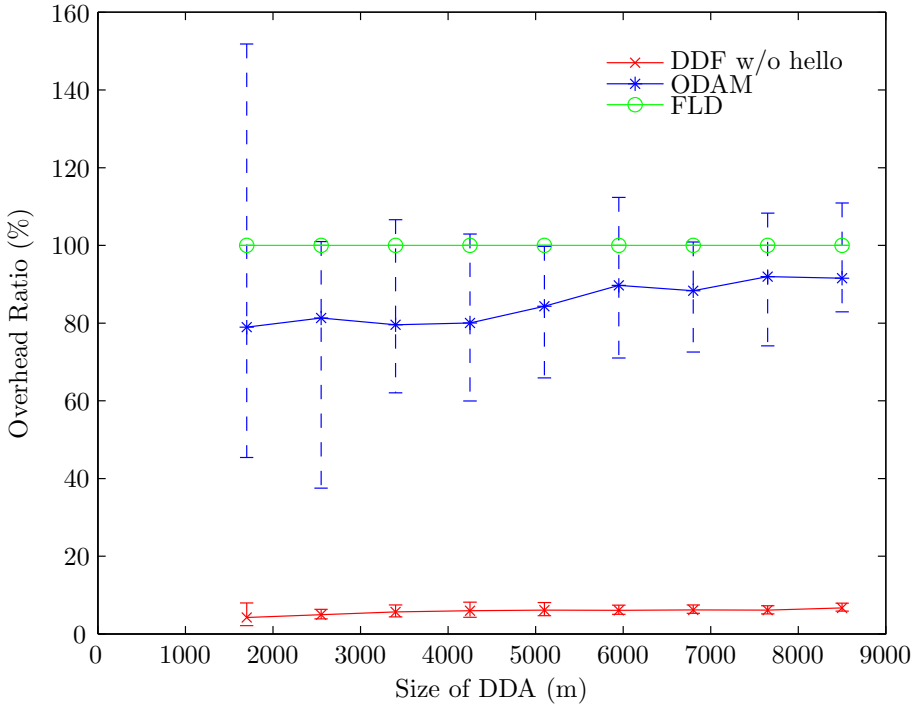


Figure 7.25: Overhead ratio (1376 veh/lane/hr)

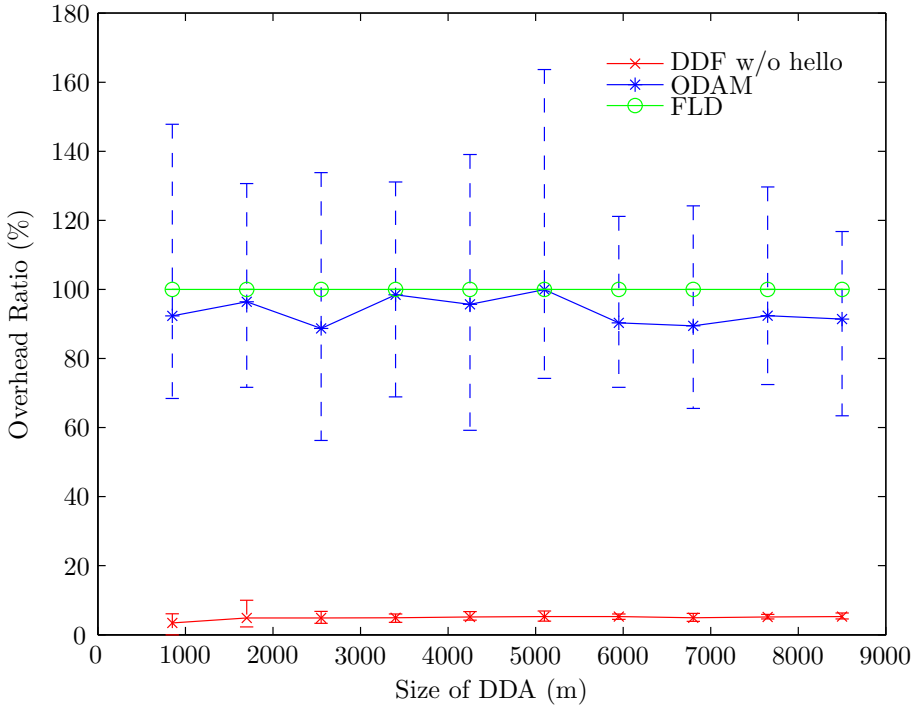


Figure 7.26: Overhead ratio (1658 veh/lane/hr)

overhead increases proportionally with an increase in the size of the time window since each node transmits a M_{beacon} periodically.

However, in the case of a sparsely connected network the increased overhead resulting from the beacon packet traffic does not effect the protocol performance as the network is too lightly loaded for contention at the MAC layer to exist.

In Figures 7.28 to 7.31 it is observed that DDF still outperforms both ODAM and flooding when the beacon traffic in the case of DDF is considered. Figure 7.28 shows that in the case of a free flowing network where no partitions occur within the *DDA*, the beacon traffic increases the mean overhead ratio to approximately 48%. As the traffic flow rate increases and the network becomes more congested the overhead ratio begins to decrease as a result of the more densely packed nodes which allow the *DDA* to be covered at a faster rate. The time window to cover a *DDA* decreases as the traffic flow rate increases and hence the beacon traffic decreases proportionally.

7.6.5 Partition Handling

Partition handling evaluates the ability of the protocol to overcome partitions within the *DDA*. Firstly, the mean partition count for each *DDA* and traffic flow rate is investigated and then secondly the occurrence of the partition and whether it was overcome successfully

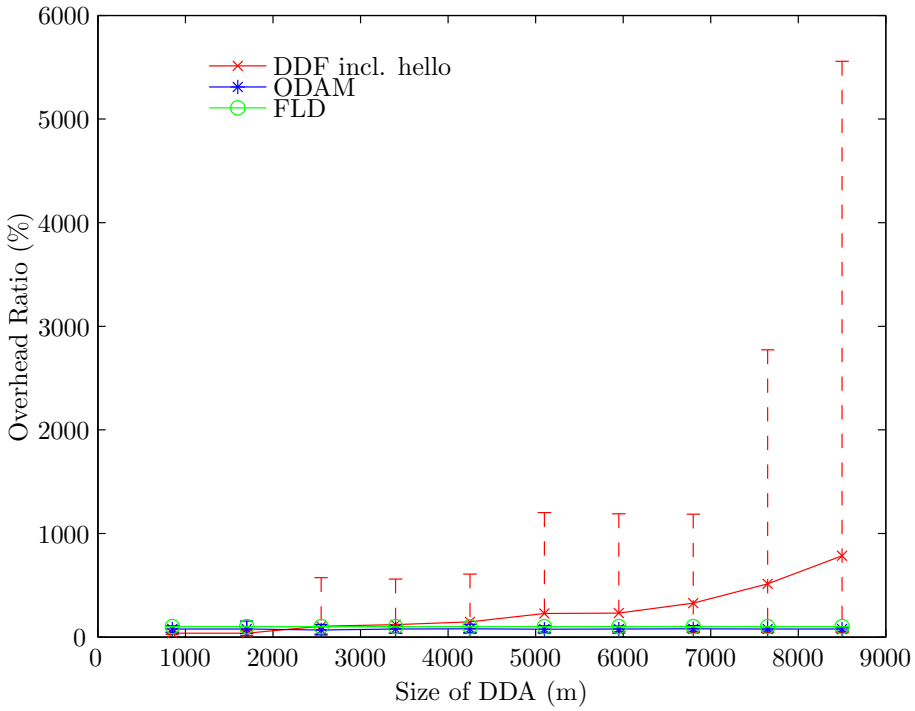


Figure 7.27: Overhead ratio with beacon traffic (549 veh/lane/hr)

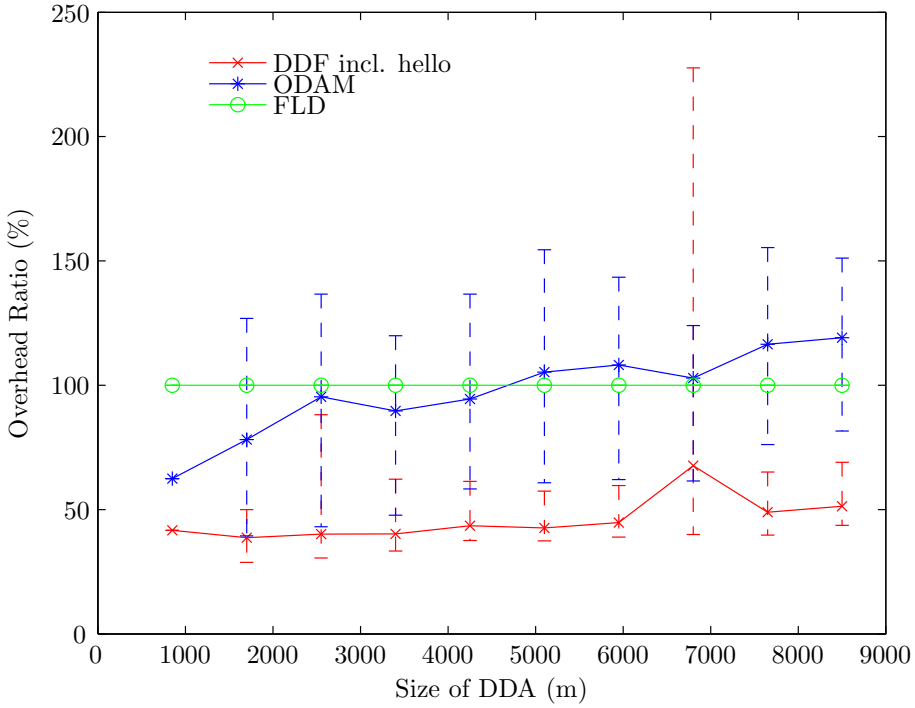


Figure 7.28: Overhead ratio with beacon traffic (822 veh/lane/hr)

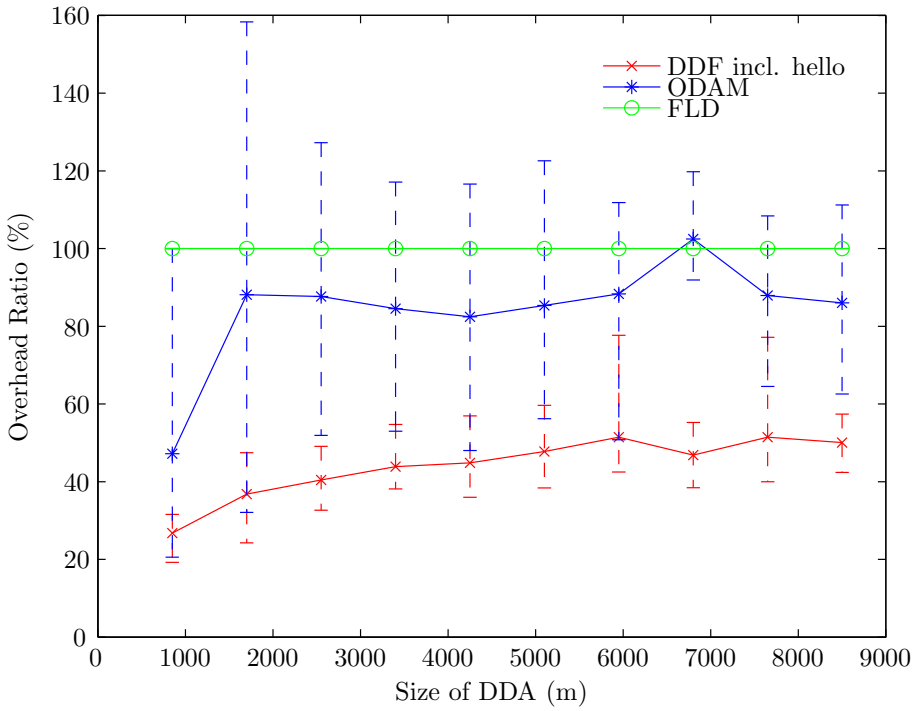


Figure 7.29: Overhead ratio with beacon traffic (1094 veh/lane/hr)

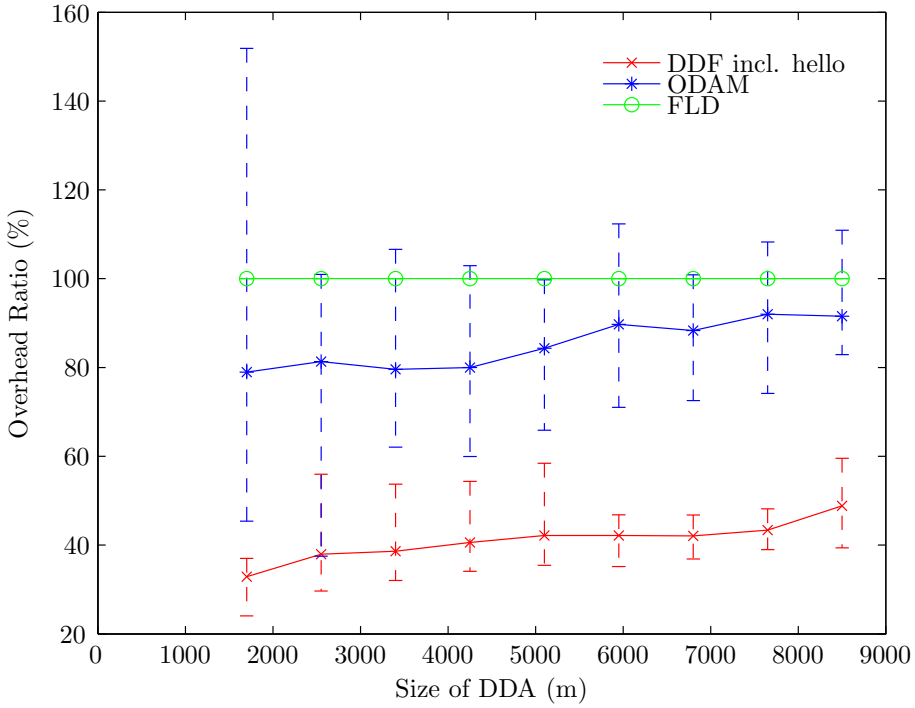


Figure 7.30: Overhead ratio with beacon traffic (1376 veh/lane/hr)

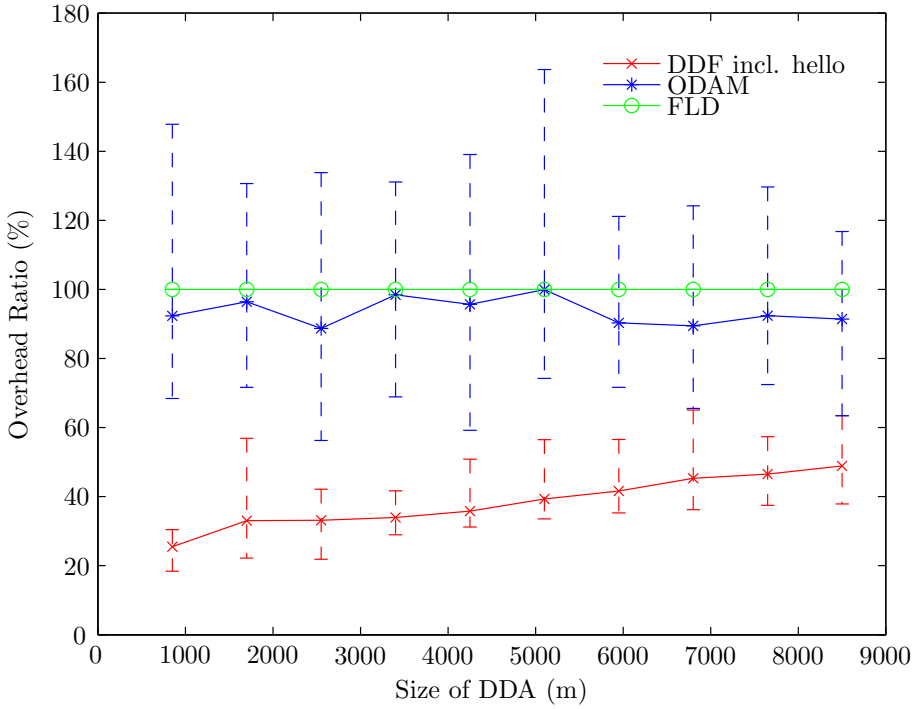


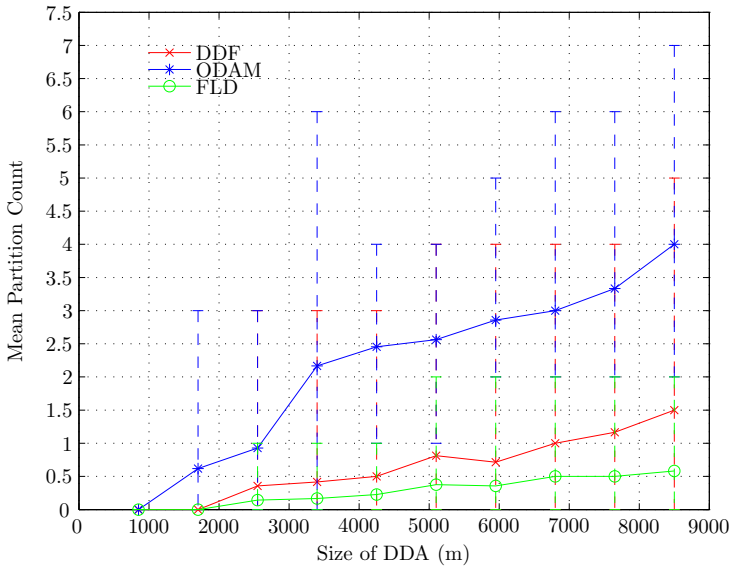
Figure 7.31: Overhead ratio with beacon traffic (1658 veh/lane/hr)

is investigated. The partition overcome count is presented as an average of the 12 instances for each *DDA* and traffic flow rate.

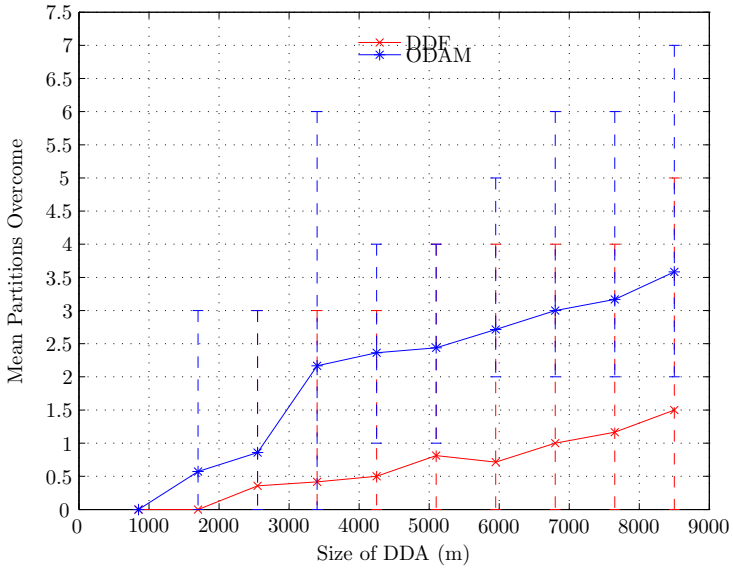
Although flooding is not able to overcome partitions, it is included in the evaluation to provide an indication of whether partitions were actually experienced. This can not be used as a reliable comparison since both DDF and ODAM will experience further partitions when they progress past the point where flooding terminates at the first partition.

In Figure 7.32(a) where the traffic flow rate and hence the density of nodes is low, partitions occur frequently within the *DDA*. ODAM experiences a higher number of partitions with each *DDA* in comparison to DDF because it only allows nodes travelling in one direction to forward the data packets, unlike DDF which allows both flows of traffic to forward the message. From Figure 7.32(b), it is observed DDF successfully overcomes all the partitions identified in 7.32(a). Although ODAM is also successful at overcoming partitions, its ability to overcome partitions decreases slightly as the *DDA* increases in size from approximately 7 km onwards.

As the traffic flow rate increases to 822 veh/lane/hr it can be observed from Figure 7.33(a) that the flooding protocol did not experience any partitions. This indicates that, theoretically, DDF should not encounter any partitions, since, like flooding, it allows both traffic flows to forward the data packet. However, from Figure 7.33(a) it can be observed that there were two instances at 2.5 km and 7.8 km where partitions occurred for DDF. Closer



(a) Partition count

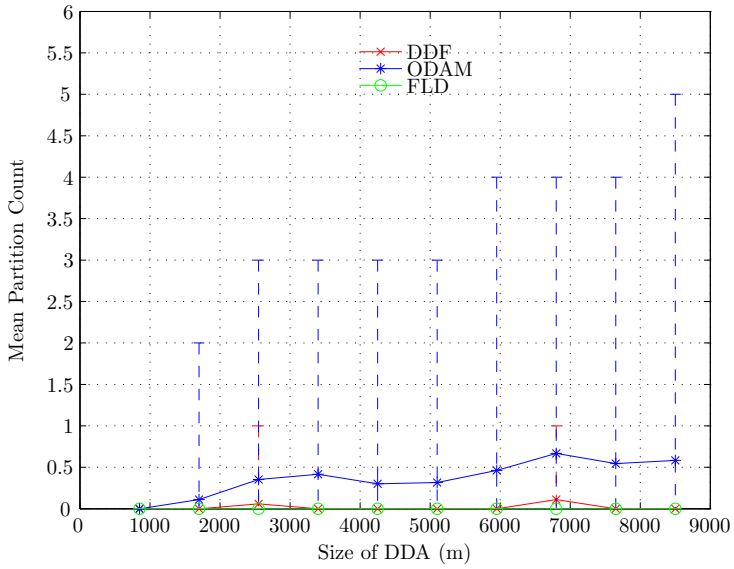


(b) Partitions overcome

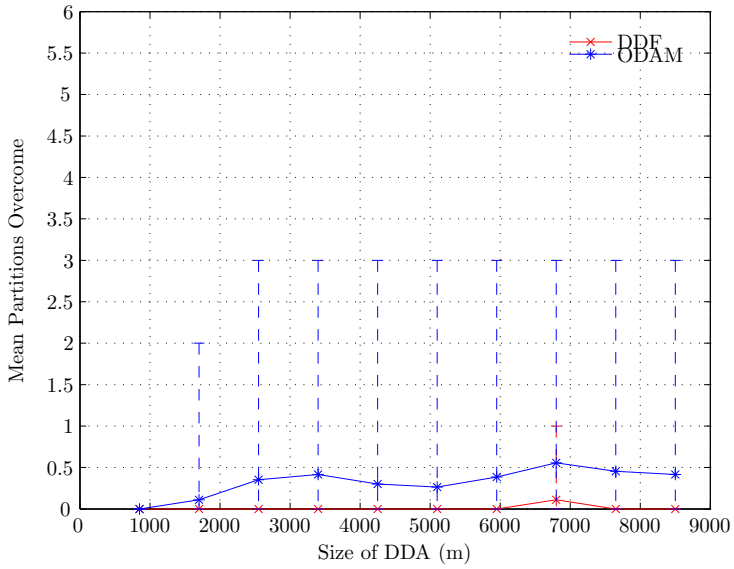
Figure 7.32: Partition handling (549 veh/lane/hr)

examination of the simulation trace files indicates that, in both instances, the partition occurred as a result of 'soft' partitions. However, in the case of ODAM, the maximum error bars in Figure 7.33(b) show that as the *DDA* increases in size from approximately 5 km, the ability of ODAM to overcome all partitions successfully begins to decrease slightly.

As the traffic flow rate increases it can be seen from Figures 7.34 to 7.36 that unlike ODAM, both DDF and the flooding protocol did not encounter any partitions. As previously discussed, ODAM again encounters partitions as a result of forwarding restrictions. As the vehicle traffic increases in density it can be observed in Figures 7.34(b) to 7.36(b) that

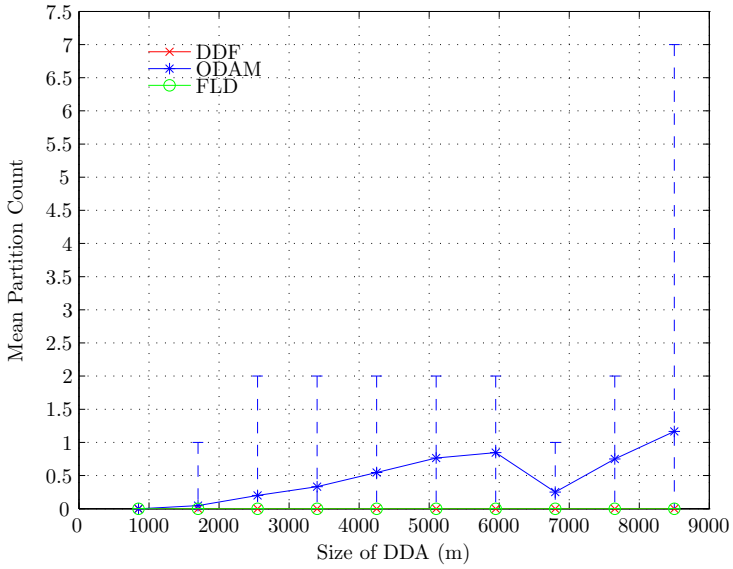


(a) Partition count

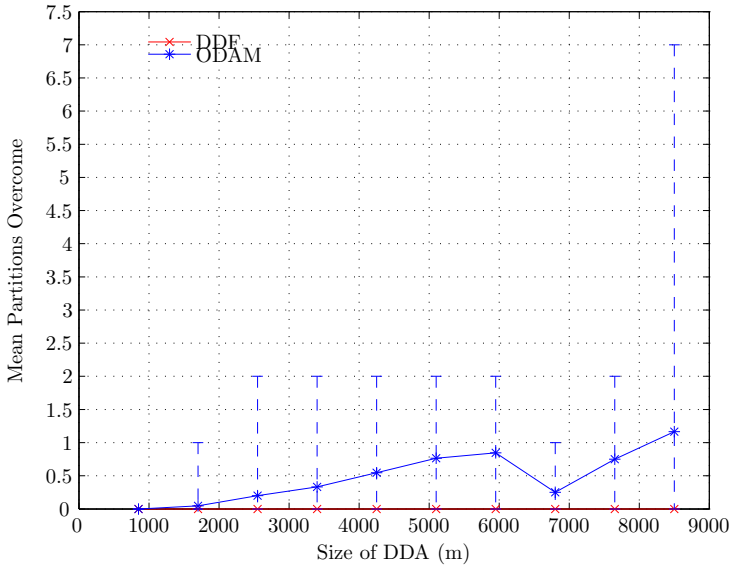


(b) Partitions overcome

Figure 7.33: Partition handling (822 veh/lane/hr)

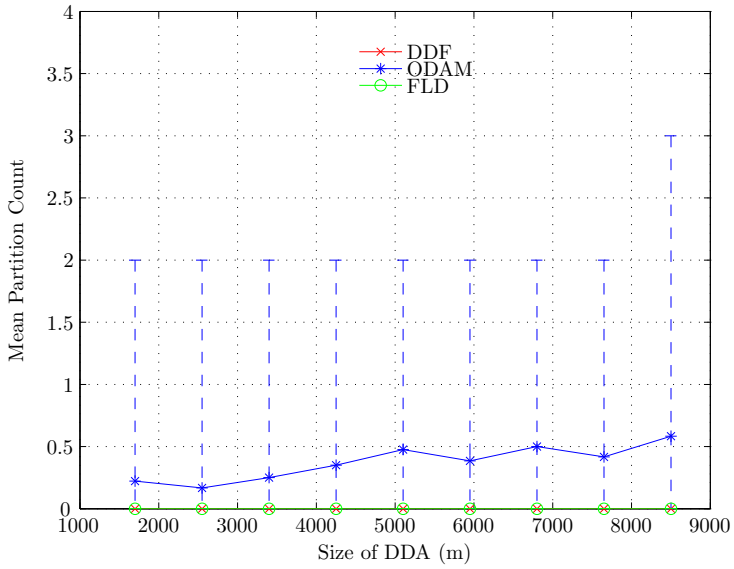


(a) Partition count

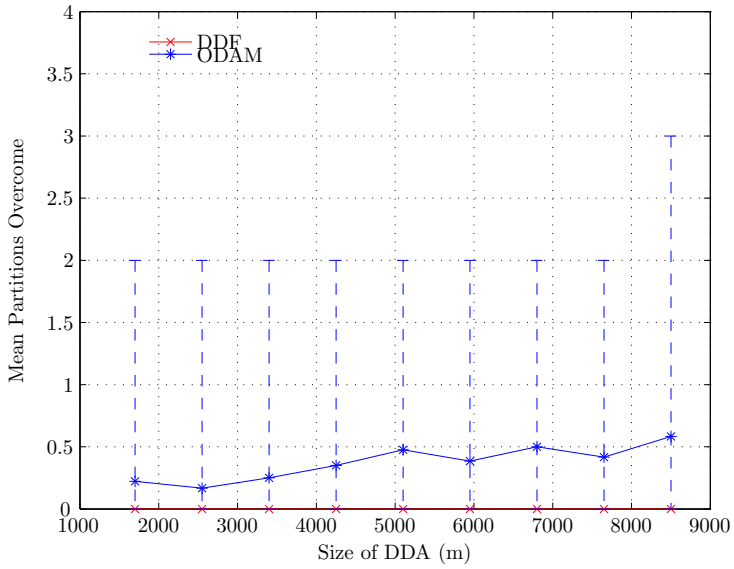


(b) Partitions overcome

Figure 7.34: Partition handling (1094 veh/lane/hr)

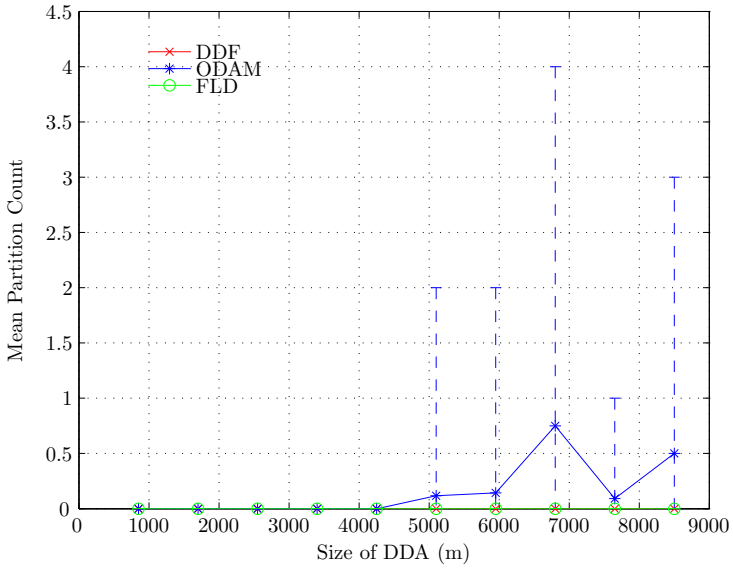


(a) Partition count

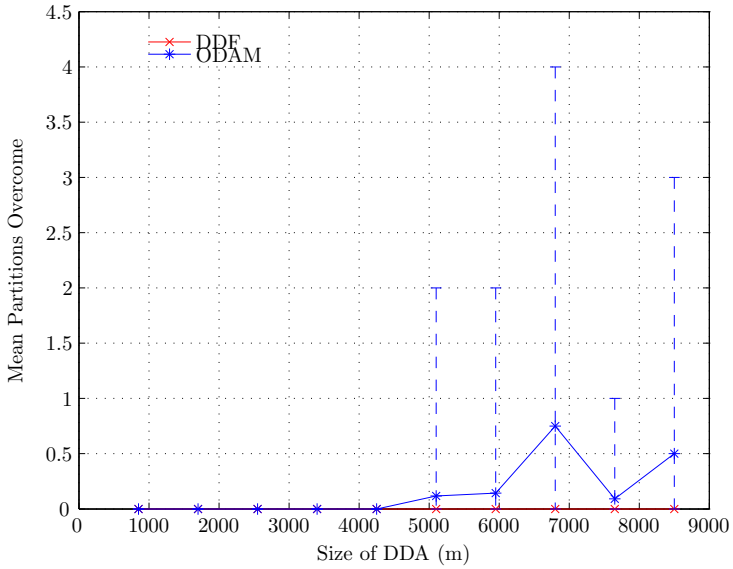


(b) Partitions overcome

Figure 7.35: Partition handling (1376 veh/lane/hr)



(a) Partition count



(b) Partitions overcome

Figure 7.36: Partition handling (1658 veh/lane/hr)

ODAM successfully overcomes all partitions.

Although ODAM successfully overcomes partitions most of the time, DDF has been observed to slightly outperform ODAM and hence provide a higher reliability that partitions will be overcome and the data packet will be disseminated successfully within the *DDA* to F_b in the case of fragmented networks.

7.6.6 Coverage delay

Finally, the coverage delay is compared within the *DDA* in order to provide an approximate speed of propagation of the data packet for each protocol.

Since the occurrence of a partition within the *DDA* will result in long coverage delays this sub-section proceeds to investigate the coverage delay in two separate studies: the first study investigates coverage delay for fully connected networks; the second study investigates coverage delay and time to overcome partitions in the case of a sparsely connected network within the *DDA*.

Fully Connected Networks

In order to investigate the coverage delay in the case of a fully connected network within the *DDA*, all simulation instances in which a partition was encountered are omitted. This allows the propagation speed of the data packet, from the source node to f_b, to be investigated with both an increase in *DDA* size and traffic flow rate. A linear best-fit is performed through the mean of all the simulation instances in order to provide an approximate comparison of propagation time.

Figure 7.37, shows the coverage delay as the size of the *DDA* increases in the case of a low traffic flow rate. In the case of low traffic flow rate and hence low *LZ* node density, the coverage delay for all three protocols is similar. The coverage delay slope for ODAM on the right-hand-side of Figure 7.37 produces an unreliable slope as a consequence of the data being omitted for larger *DDA* sizes where partitions occurred; the slopes however, remain comparable.

As the traffic flow rate begins to increase from 822 to 1658 veh/lane/hr as shown in Figures 7.38 to 7.41 it can be observed that the both ODAM and flooding outperform DDF in terms of the speed of propagation in addition to maintaining a constant rate of data packet propagation as the traffic flow rate increases. In comparison, in the case of DDF, the speed of propagation increases with an increase in traffic flow rate because the deferral mechanism is more sensitive, by design, to changes in *LZ* node density. Therefore,

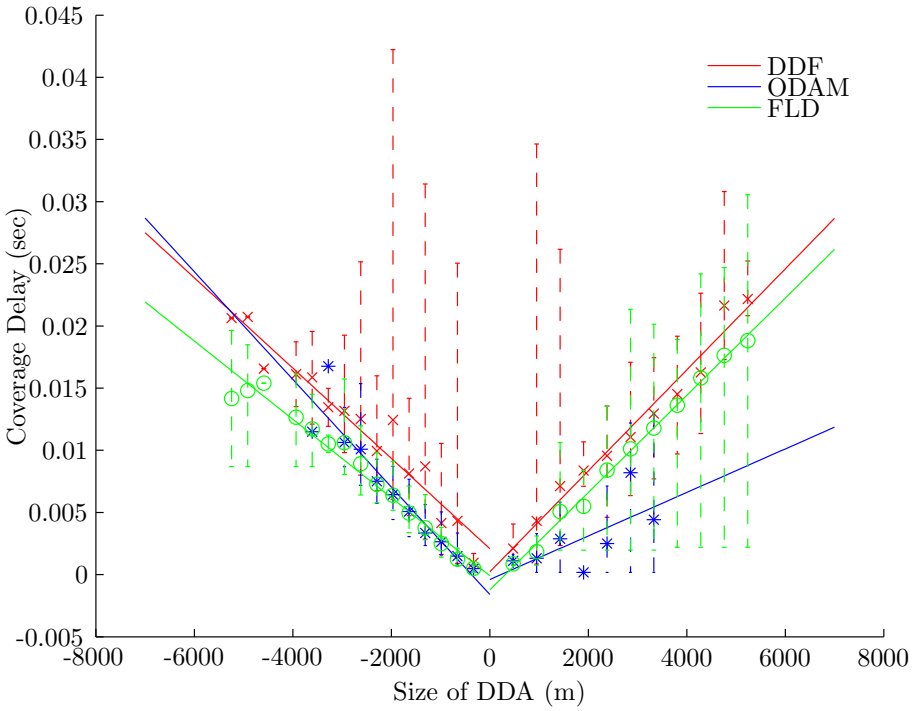


Figure 7.37: DDF:coverage delay without partitions (549 veh/lane/hr)

as the traffic flow rate increases along with LZ node density and hence channel activity, whenever the secondary propagation mechanism forms a link in the propagation chain, a corresponding delay determined by the deferral calculation will be incurred. Therefore, the cost of ensuring minimal protocol overheads results in a lower data packet propagation speed.

Although DDF has a lower data propagation speed than both ODAM and flooding, the absolute delay remains in the 10's of ms order of magnitude for $DDAs$ up to a few tens of km in length. This is inconsequential for the application scenarios of interest and meets the dissemination requirements discussed in Chapter 2.

The slope of the best-fit line through the mean of all simulation instances for each protocol in Figures 7.38 to 7.41 describes the message propagation speed and is constant in each case. In the case of DDF, the propagation speed averages to approximately 2 ms per hop for a transmission range of 300 m, whereas for ODAM and flooding it averages to approximately 1 ms per hop. In Figure 7.41 there is more variability in propagation speed as a result of removing simulation data points where partitions occurred. As can be seen from Table 2.1 in Chapter 2, the latency requirements of typical emergency applications vary from 50 - 100 ms. Thus, all the protocols discussed in this thesis have a warning delay margin of safety of 1 to 2 orders of magnitude. In reality, this margin is likely to be less than when taking into account the additional delays incurred by the communication

protocol processing stack and its effect on packet size. A further complication arises from the fact that the standard for emergency messages stipulate a message repetition frequency as high as 10 Hz which would place a significant offered traffic loading on the communications channel potentially giving rise to longer delays. This further stresses the significance of adaptation to congestion that such protocols must employ.

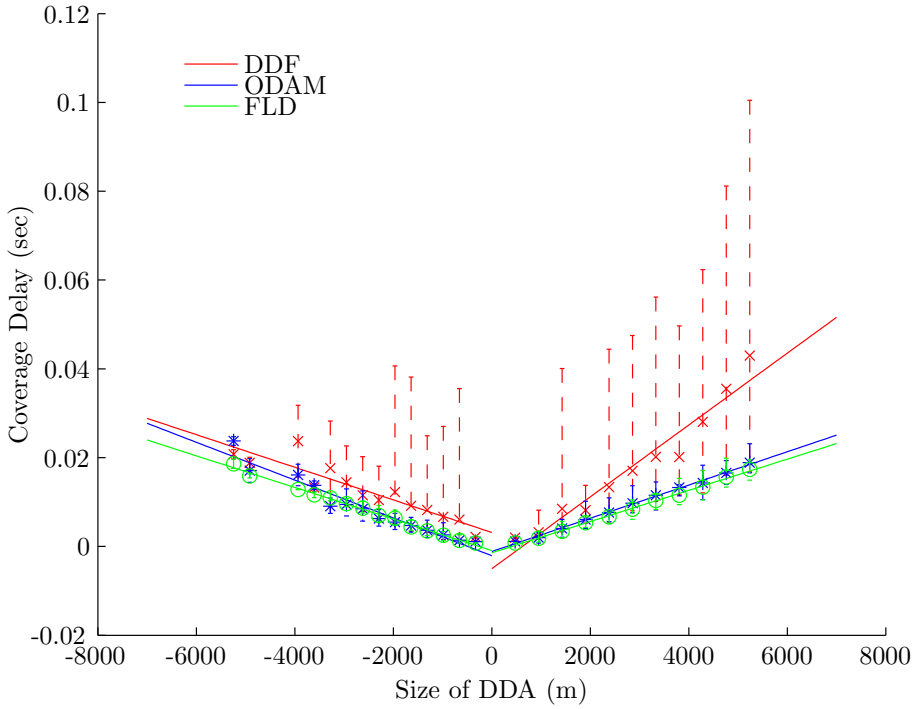


Figure 7.38: DDF:coverage delay without partitions (822 veh/lane/hr)

Partially Connected Networks

In order to evaluate the comparative performance of the protocols when faced with partitions, ODAM and DDF are considered only since flooding does not overcome partitions. Additionally, the focus is on the case of the low traffic flow rate only, since DDF only experienced 'hard' partitions at this traffic flow rate, as previously discussed in the results for partition handling. In order to investigate the coverage delay for a partially connected network, simulation instances when a partition occurs in the *DDA* are included only.

It can be observed from Figures 7.42 and 7.43 that ODAM takes considerably longer to overcome partitions. This occurs as a result of transmitting the data packet periodically, in addition to restricting the forwarding of the message to one direction of traffic flow unlike DDF which hands over the forwarding role when a node closer to F_b is detected within its LZ connectivity information, regardless of the direction of traffic flow.

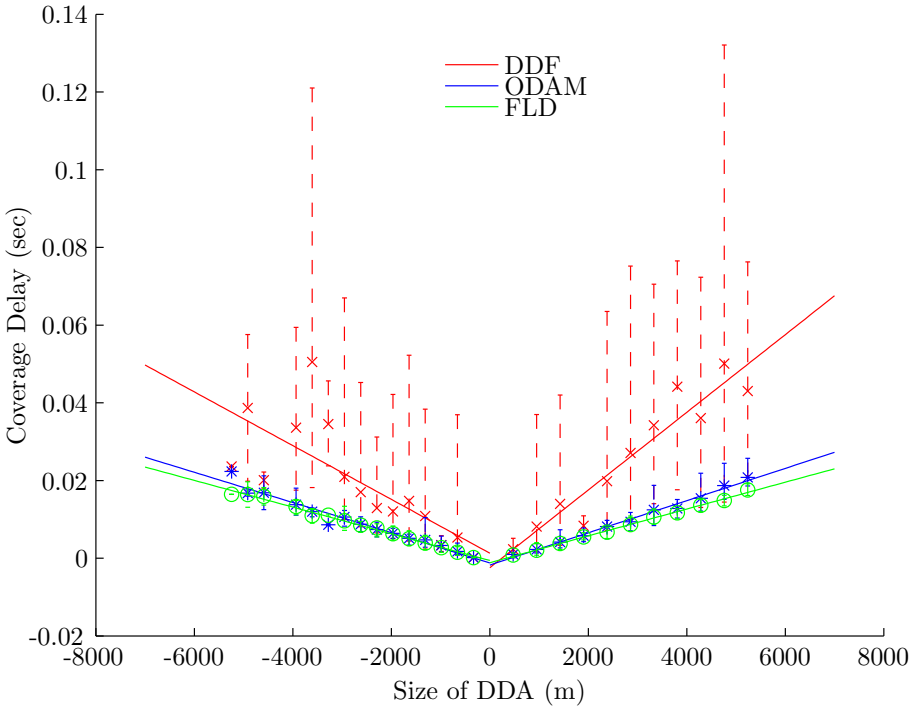


Figure 7.39: DDF:coverage delay without partitions (1094 veh/lane/hr)

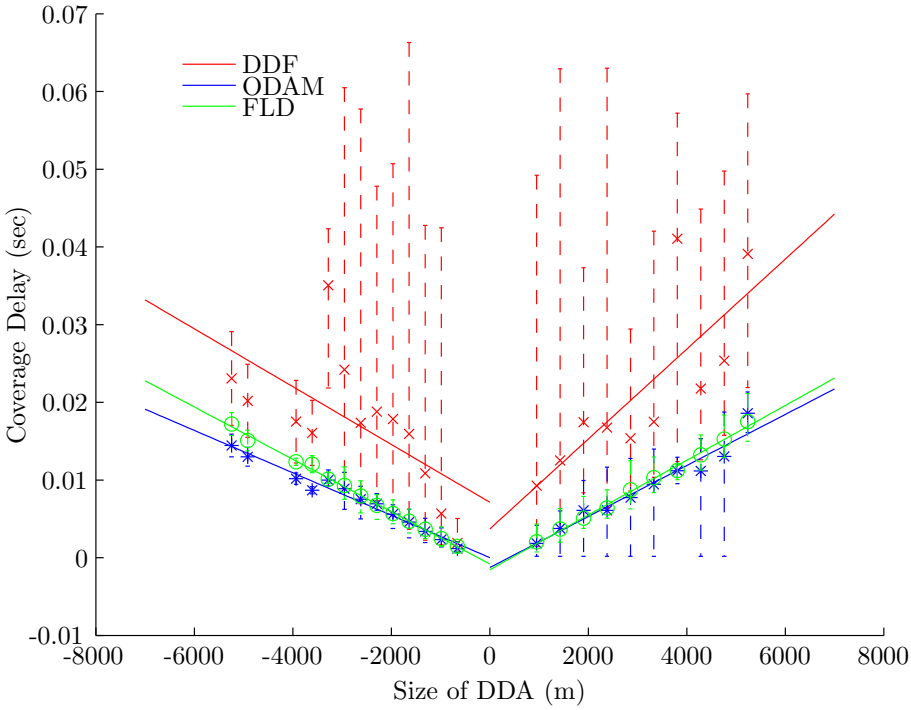


Figure 7.40: DDF:coverage delay without partitions (1376 veh/lane/hr)

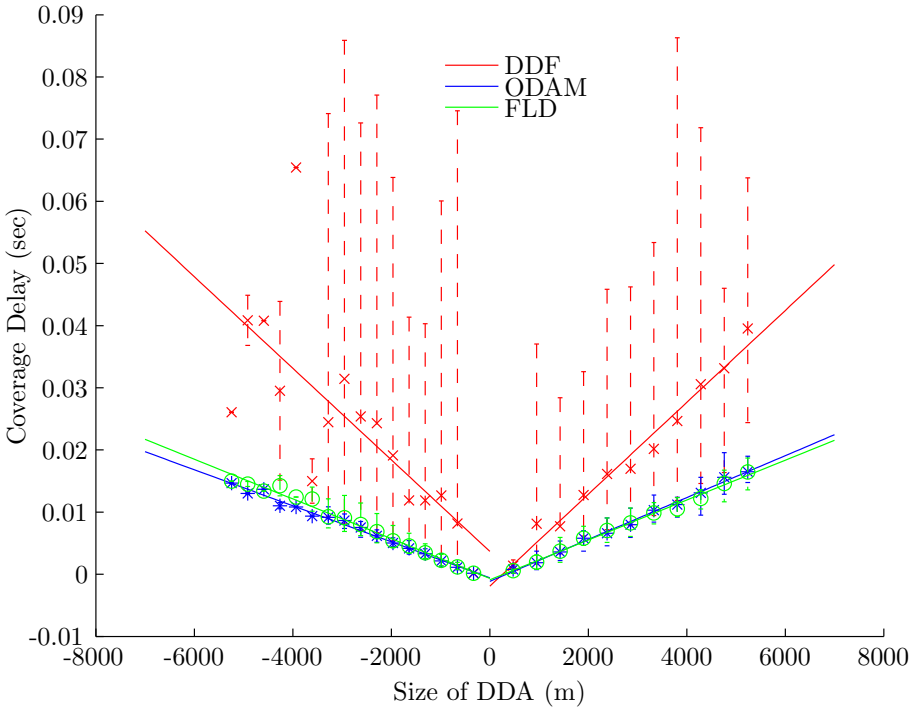


Figure 7.41: DDF:coverage delay without partitions (1658 veh/lane/hr)

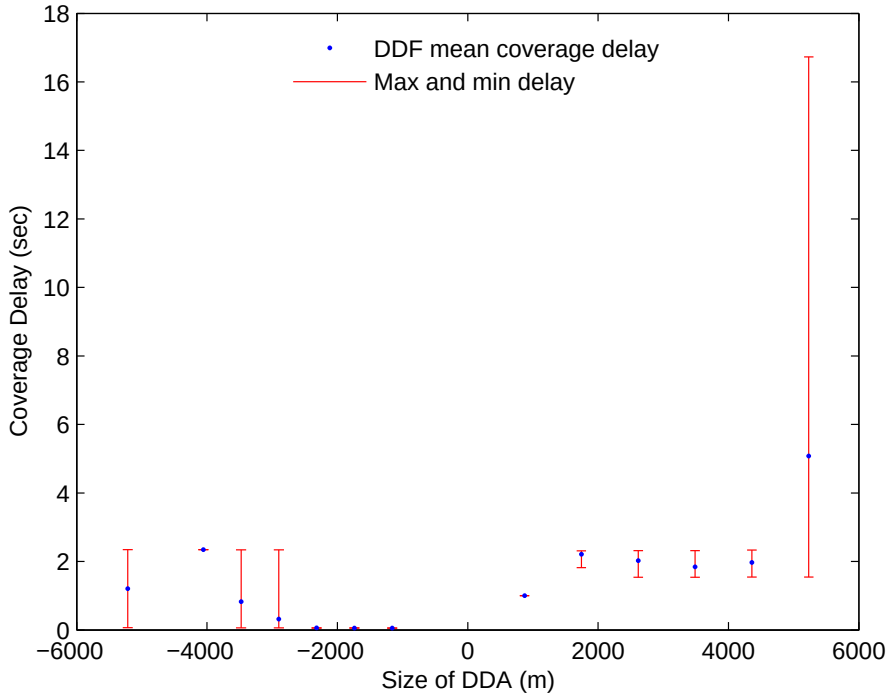


Figure 7.42: DDF:coverage delay with partitions (549 veh/lane/hr)

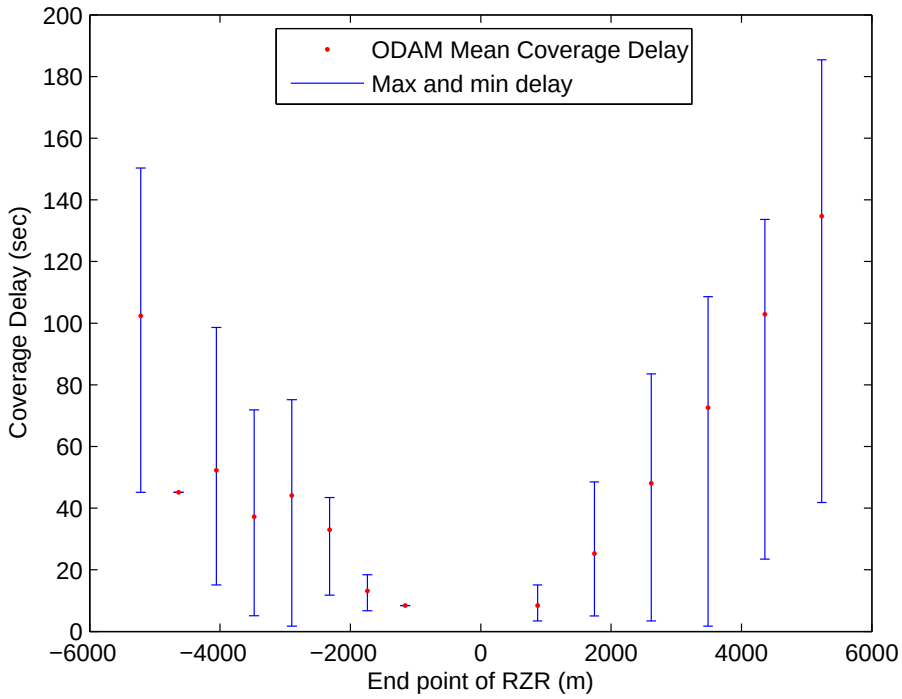


Figure 7.43: ODAM:coverage delay with partitions (549 veh/lane/hr)

7.7 Synoptic Discussion of Results

In order to gain a comprehensive picture of the performance of the three protocols, their relative merits are summarised below with respect to each of the metrics considered in this chapter.

Area coverage ratio

DDF was found to provide a mean area coverage ratio of approximately 100% as both the size of the *DDA* and traffic flow rate increase. Indeed, in the case of sparsely connected networks, the partition handling mechanism overcame network fragmentation, allowing the dissemination process to successfully cover the *DDA*. In comparison to DDF, although ODAM is able to overcome partitions, the area coverage was found to deteriorate with increasing *DDA* size. This occurred as a direct consequence of the dissemination process terminating prematurely as a result of the transmission from the source node being received only by nodes travelling in the non-forwarding direction. However, this problem can be potentially overcome through message persistence at the source node.

Delivery Ratio

The delivery ratio is generally comparable between the examined protocols. However, DDF achieves the most consistent mean performance level with an increase in the size of

DDA and traffic flow rates. The slight variability in performance for both ODAM and flooding, as shown by the (max-min) error bars in the plotted results, was found to occur as a result of local MAC layer collisions. Additionally, in the case of ODAM, a small number of nodes did not receive the message as a result of missing the periodic transmission used to overcome partitions. However, the isolated instances of variability shown by the error bars for DDF appear to be worse than that of both ODAM and flooding, in the case of the more congested networks. Termination rules at the forwarding boundary (F_b) coupled with local collisions prevented nodes from successfully receiving the message. However, when the *DDA* size is further increased in these instances, it was established that these nodes receive the message successfully, hence confirming that the termination conditions cause an artificial dip in performance.

Forwarding ratio

The primary forwarding mechanism is the most efficient method of disseminating the message throughout the *DDA*, therefore, the lower the forwarding ratio the more efficient the algorithm. From the evaluation of the forwarding ratio, it was seen that DDF protocol significantly outperformed ODAM. Moreover, DDF achieved a forwarding ratio within approximately 2% of the theoretical forwarding ratio throughout increases in both the traffic flow rate and sizes of *DDA*.

On the other hand, ODAM was found to perform worse than the baseline flooding protocol as the size of the *DDA* and traffic flow rate increased. The poor performance of ODAM in comparison to DDF can be attributed to a number of reasons: Firstly, unlike DDF, an ODAM node transitions into the role of forwarding node when its deferral timer expires and it has not overheard the message being forwarded. However, in most instances the deferral time was found to be too short which meant that nodes had not overheard the message being forwarded prior to their deferral timer expiring. This means that the majority of nodes within a transmission radius acted as a forwarding node retransmitting the message. Secondly, similar to DDF, a node in ODAM is able to act as a forwarding node if it has previously received a message and is positioned closer to the forwarding boundary than the source of the transmission. However, delayed transmissions from forwarding nodes waiting to access the communication channel will be received by nodes having previously forwarded the message and, therefore, operate as a forwarding node again. Hence, the ODAM forwarding ratio exceeds that of flooding. DDF minimises this problem using data maintained for the suppression tracking mechanism in conjunction with additional rules which restrict a node from participating in the forwarding process again, further along the forwarding chain.

In the DDF protocol, a node receiving a message for which it has a copy in its message seen table is only allowed to participate in the forwarding process again if its current position is closer to the forwarding boundary than both the position of the transmitting node and the value stored in the position field of the matching entry in the message seen table. The position field entry in the message seen table records the location of the node last time it either participated in the forwarding process or last overheard the message closer to the forwarding boundary than the previous entry.

Retransmission Ratio

The retransmission mechanism is the secondary forwarding mechanism which operates on the occasion that the forwarding chain has not been overheard to be progressing successfully towards the forwarding boundary.

From the evaluation of the retransmission ratio it was seen that DDF maintains a consistently low retransmission ratio with increases in the size of the *DDA* and traffic flow rate. In the case of ODAM, the retransmission ratio was found to occur as a direct consequence of retransmissions resulting from overcoming partitions within the network. In particular, when the network is sparsely connected ODAM's retransmission ratio exceeds that of DDF significantly. The retransmission ratio of ODAM increases linearly in proportion to the size of the *DDA*.

The DDF retransmission ratio results show that the occurrence of coverage conditions not being met by the primary forwarding mechanism is relatively low. Moreover, the additional retransmissions occurring for DDF as a result of overcoming partitions, have a small effect on the retransmission ratio. In the case of ODAM, it can be observed that it does not incur retransmissions as a result of coverage requirements not being met. This is because the number of nodes retransmitting as a forwarding node is high and the unnecessary retransmissions by forwarding nodes are acting like the intermediate nodes in DDF. ODAM is essentially operating like a flooding protocol as a result of the short deferral timing which generally ensures that the requirements of the acknowledgement chain are met, albeit at a significant expense of unnecessary message retransmissions.

The occurrence of retransmission resulting from the coverage conditions not being met by the primary DDF forwarding mechanism is a small fraction of the protocol overhead, at lower than 5% for all occasions. This is observed from the comparison of retransmission ratio and protocol overhead ratio plots in §7.6.4 and §7.6.4 respectively.

The DDF forwarding rules do not assume an orderly sequence of events, but can mitigate for the uncertainties and stochasticity inherent at the physical layer of wireless communications, resulting in significant efficiencies and reliability at the same time.

Overhead Ratio

The analysis of the overhead ratio shows that DDF is able to scale extremely well over the varying traffic dynamics tested, from sparsely through to highly congested vehicle traffic networks, with increasing sizes of *DDA*. On the other hand ODAM's operation was found to deteriorate with an increase in both traffic flow and size of *DDA*. Moreover, as the traffic flow rate increases the performance of ODAM tends to follow the performance observed for the flooding protocol, and in some instances performs worse than flooding.

ODAM's decreased performance occurs as a consequence of the small deferral times compared to typical channel access delays. In addition, the fact that ODAM does not include a suppression mechanism means that erroneous nodes are not prevented from forwarding the warning message, which implies that unnecessary message retransmissions occur in comparison to DDF.

Overhead Ratio with Beacon Messages

When the beacon traffic is taken into account in the analysis of overhead ratio, it was observed from the results that the overhead for DDF was considerably higher in the case of sparsely connected networks. This is attributed to the length of the partitions within the *DDA* and hence the time window over which the *DDA* is covered. However, in the case of a sparsely connected network the increased overhead resulting from the beacon packet traffic does not affect the protocol performance as the network is too lightly loaded for contention at the MAC layer to exist.

In terms of the number of packets generated, it was observed that as the traffic flow rate increases DDF outperforms both ODAM and flooding. As the traffic flow rate increases and the network becomes more congested the overhead ratio begins to decrease as a result of the more densely packed nodes which allows the *DDA* to be covered at a faster rate. The time window to cover a *DDA* decreases as the traffic flow rate increases and hence the beacon traffic decreases proportionally.

Partition handling

Although both DDF and ODAM are able to successfully overcome partitions, DDF was found to successfully overcome all instances of partitions in comparison to ODAM which failed in a small number of instances. ODAM was observed to experience more instances of partitions as a result of the restriction of allowing only one traffic flow to forward the message. Thus, the performance of ODAM is expected to match that of DDF if both flows of traffic are used in the forwarding process.

Coverage delay

The coverage delay provides a measure of the length of the time window over which the warning message is disseminated from the source node to the forwarding boundary for varying traffic dynamics and varying sizes of *DDA*. In the case of sparsely connected networks where no partitions occurred the coverage delay was found to be comparable between all three protocols. However, as the traffic rate increases both flooding and ODAM were found to outperform DDF. This is because the DDF deferral mechanism is more sensitive to changes in *LZ* activity and therefore as *LZ* conditions change with an increase in node density (traffic flow rate) the deferral timing adapts accordingly. Therefore, the cost of ensuring minimal protocol overheads results in a slightly lower data packet propagation speed.

Although DDF has a lower data propagation speed than both ODAM and flooding, the absolute delay remains in the 10's of ms order of magnitude for *DDAs* up to a few tens of km in length. This is inconsequential for the target application scenarios and meets stated research dissemination requirements of this thesis.

When the coverage delay is considered in the case of a partially connected network, DDF was found to outperform ODAM. This can be attributed to the difference in the mechanisms employed in overcoming partitions; ODAM retransmits the warning message periodically in addition to restricting the forwarding of the message to one direction of traffic flow, whereas DDF uses local connectivity knowledge through the exchange of beacon messages to detect when a partition has ended, in a more timely manner.

7.8 Summary

This chapter has evaluated the performance of DDF against a baseline flooding protocol and ODAM, which is a basic distance deferral based protocol. The simulation scenario was carried out over varying traffic flow rates representing realistic traffic conditions on UK highways, ranging from free-flowing (sparsely connected networks) through to highly congested start-stop traffic conditions, over varying (increasing) sizes of *DDA*.

The simulation results show that DDF has achieved the research design goal by scaling efficiently with both increasing traffic flow rates and *DDA* size. The forwarding and suppression mechanisms along with the maintenance of local connectivity information ensure that small overheads are consistently incurred in the dissemination of the message, meeting the research design aim in terms of economy of messaging. Moreover, the adaptive deferral mechanism ensures unnecessary retransmissions do not occur, in comparison to ODAM. In ODAM such retransmissions occur as a result of the deferral time being too

short in comparison to MAC access delays, resulting in unnecessary retransmissions and hence increased overheads.

In [150], Sedletsky proposes a geocasting protocol for highway environments whose performance is also evaluated in comparison to ODAM. Results from the performance evaluation in [150] show that ODAM also incurs a significantly higher number of retransmissions and hence an increase in message collisions as the traffic flow rate increases and that the dissemination time to cover the addressed region was found to be similar. Sedletsky also concludes that ODAM is less stable to changes in vehicle traffic flow rate. Although the assumptions and simulation conditions used in the performance evaluation of the DDF protocol with ODAM in this thesis are very different from what is presented in [150], it can be seen that similar conclusions have been drawn in terms of the stability of ODAM and the overhead it generates as traffic flow rate increases.

In terms of reliability this research has realised the design goal of providing a high level of message delivery throughout the *DDA*, ensuring that the protocol successfully overcomes partitions whilst incurring minimal overheads.

DDF delivers the message in a timely manner throughout all simulation scenarios evaluated, in a matter of 10's of ms. Its time to cover the *DDA* is slightly higher than that of both ODAM and flooding as the traffic flow rate increases to highly congested levels. This slight coverage delay in comparison with ODAM and flooding is the tradeoff that ensures efficiency in order to incur minimal dissemination overheads.

CHAPTER 8

CONCLUSIONS AND FURTHER WORK

8.1 Thesis Summary

The goal of this research is to propose a data dissemination protocol for vehicle-to-vehicle communications which is able to adapt and overcome the challenges of operating in the highly dynamic vehicular environment, in order to support safety related cooperative vehicle applications over an unreliable MAC scheme. In order to address this goal the work carried out is summarised below.

Chapter 2 considers a number of applications which would benefit from, or whose deployment relies on, direct radio communication between vehicles. Current standardisation activities are reviewed in relation to the enabling communication access technology which is based on the IEEE 802.11 WLAN standards. Having considered the communication requirements for a number of safety related applications through use case analysis (Appendix A) a data dissemination framework is then defined by considering ad hoc routing techniques used in the MANET field and their suitability for safety related applications in the vehicular environment. Finally, the goals of the research are presented, which was to focus on the geocasting of event driven messages through the provision of local connectivity information obtained through the periodic exchange of beacon messages. Moreover, this chapter highlights particular challenges such as the hidden terminal and broadcast storm problems which arise from the unreliable channel access mechanism specified by IEEE 802.11 MAC scheme in broadcast mode.

Chapter 3 provides a review of geocasting and broadcasting mechanisms used for the dissemination of vehicular messages in the literature. From this review it was determined that there is a lack of publications which take a holistic view in implementing dissemination solutions that consider operation over all traffic conditions; from low density, sparsely connected networks, to high density, congested networks.

A novel adaptive data dissemination protocol called Data Dissemination Forwarding (DDF) is proposed by this research in Chapter 4. Firstly, design decisions made in specifying the requirements of the DDF protocol are covered and then its four main mechanisms are introduced: Forwarding; local zone connectivity; node suppression and partition handling. The remainder of the chapter provides a detailed description on the operation of the DDF protocol. This discussion also includes the methodology used to construct the retransmission deferral timing equation employed by the forwarding mechanism which uses results from an analysis of the spread of MAC channel access delays documented in Chapter 6.

In Chapter 5 the methodology used to evaluate the performance of the proposed protocol is presented. Firstly, this chapter considers methods used to assess the performance of vehicular *ad hoc* network protocols and concludes that a simulation methodology using realistic traffic flow movements best meets research objectives. The simulation environment is then presented and followed by a review of methods for generating traffic flow profiles, introducing the simulator and the generation of the traffic flow files used in the evaluation. A short survey of network simulators follows along with an overview of the chosen simulator and its modelling methodology. The discussion of the simulation environment concludes with the methodology used to analyse the data resulting from the simulation of the vehicular *ad hoc* network. The remainder of the chapter presents an implementation of the proposed protocol in the chosen network simulator.

An empirical analysis of MAC channel access delay based on simulation of the periodic exchange of beacon messages between one-hop neighbouring nodes is discussed in Chapter 6. This chapter firstly considers the requirements for the analysis and provides a brief review of channel access delay studies. The methodology used to carry out the evaluation is then introduced. In the remainder of the chapter, an analysis of the results is presented and the steps taken to extract the spread of MAC access delays, which are used by the forwarding mechanism in the proposed protocol to provide adaptivity to local traffic density and data packet intensity variations.

Chapter 7 finally presents the results of the simulation experiments used to evaluate the performance of the proposed protocol against a similar data dissemination protocol and a simple flooding based protocol. The simulation scenario used to evaluate the comparative performance of the protocols is presented along with the implementation and settings of the comparison protocols. The evaluation metrics are described and finally the results of the simulations along with a detailed discussion of the results are presented.

8.2 Conclusions

This research has provided a reasoned framework for data delivery from an application perspective. In the context of this thesis the research focuses on a particular aspect of this framework, which is the delivery of event driven messages using geocasting techniques in addition to periodic beacon messages used as an underlying mechanism for the deployment of safety related applications. The specific goal of this research is to propose a geocast protocol using local zone knowledge which is able to adapt to the highly dynamic nature of vehicular traffic, providing reliable, efficient and timely delivery over an unreliable MAC scheme.

Thus, this research proposes a protocol for the dissemination of event driven safety-related messages and a mechanism for the exchange of periodic bacon messages used to build up local zone connectivity knowledge, which in turn, is used by cooperative vehicular and event driven applications. The proposed DDF algorithm uses four main mechanisms to enable it to meet the research goal. The main properties of these mechanisms are summarised as follows:

Forwarding Mechanism: The forwarding mechanism is responsible for efficient and reliable message dissemination addressed to a geographic area. The propoosed forwarding scheme is loosely based on the concept of distance based contention forwarding where forwarding decisions are made by each node independently, based on implicit local zone connectivity knowledge. The forwarding scheme ensures reliability and timeliness of delivery by controlling channel contention at the network layer whilst providing distance ordered retransmission timing to avoid excessive channel access delays at the MAC layer. Adaptability to vehicle density and packet intensity has been implemented through local sensing of these conditions in conjunction with MAC delay estimation parameters determined from a comprehensive analysis of the spread of MAC channel access delay. In highly congested networks the adaptability of the retransmission timing mechanism ensures efficient and reliable delivery.

Local Zone Connectivity: This research implements periodic beacon exchange between one hop neighbours and classifies neighbouring vehicles according to their relative position and driving direction. This classification enables a node to determine neighbouring nodes which will take part in the forwarding process according to the application-defined RZR. Additionally, LZ connectivity information expedites the forwarding decision process. Moreover, the classification process supports the partition handling mechanism to overcome partitions in an efficient and speedy manner when an eligible node is detected, so as to continue the dissemination process. LZ monitoring enables application processing to detect hazardous situations.

Suppression Mechanism: Protocol efficiency is further increased in this research through explicitly suppressing nodes which are detected to be transmitting erroneously. This is achieved through a three-step process which detects, tracks and actively suppresses nodes deemed to be engaging in erroneous retransmissions.

Partition Handling: Network fragmentation is overcome within the dissemination area by implementing standard partition handling techniques which are coupled with the local zone connectivity mechanism in order to efficiently resume dissemination when a forwarding opportunity arises.

The discussion on metrics in §7.6 and §7.7, which quantify the comparative performance of the DDF protocol against ODAM and the simple flooding based scheme, leads to the following conclusions.

The DDF protocol was observed to provide the most consistently reliable mean coverage ratio of 100%, as both the size of the DDA and traffic flow rates increased in comparison to both flooding and ODAM whose performance is affected by variations in the traffic flow dynamics. In the instances where ODAM's performance was observed to deteriorate this was found to occur as a consequence of the restriction which limits message forwarding to one direction of traffic flow only, which caused the dissemination process to terminate prematurely. Therefore, in order to ensure reliable coverage both directions of traffic flow, as used in DDF protocol, should be utilised to increase reliability of the dissemination process.

The delivery ratio was found to be comparable between DDF, flooding and ODAM. However, DDF was found to achieve the most consistent performance level, which varied between 98 - 99% in terms of scaling with increased node density and the size of the message delivery area, whereas ODAM and flooding varied between 92 - 99%.

Achieving messaging economy and efficiency was an important goal of this research, and indeed, from the results for the forwarding and retransmission ratio in Chapter 7, it can be observed that the DDF protocol has achieved this goal. The DDF protocol was found to significantly outperform ODAM. Moreover, the forwarding ratio was found to be consistently within 2% of the theoretical forwarding ratio as both DDA size and traffic flow rate increased. DDF's consistency in performance can be attributed to its ability to adapt to local variations in both vehicular traffic dynamics and data traffic in comparison to ODAM whose deferral timing is too short, which results in nodes retransmitting unnecessarily. Moreover, ODAM's high forwarding ratio is compounded by its simple rules which govern whether a node participates in the forwarding process again, unlike DDF which tracks the location of the forwarding chain enabling more considered and intelligent decisions and hence reducing unnecessary retransmissions.

Forwarding reliability is ensured through the message deferral mechanism on the occasion that the forwarding chain has not been overheard to be progressing. In the case of DDF the level of the retransmitting ratio is found to be consistently low with increases in traffic flow rate and size of the DDA. Indeed the retransmission ratio for DDF accounts for $< 5\%$ of the total protocol overhead. In the case of ODAM retransmissions occur as a result of broadcasting periodically to overcome partitions. In ODAM forwarding reliability is implemented inadvertently through retransmissions from forwarding nodes, which provides a level of reliability at the expense of a significantly increased protocol overhead.

In terms of protocol overhead ratio, DDF is able to scale extremely well over the varying traffic dynamics tested, from sparsely through to highly congested vehicle traffic networks, with increasing sizes of *DDA*. In comparison ODAM was found to deteriorate with an increase in both traffic flow and size of *DDA*. ODAM tends to follow the performance observed for the flooding protocol, and in some instances performs worse than flooding. DDF's ability to adapt to local variability, the suppression mechanism and the forwarding participation rules based on forwarding chain tracking, all prevent unnecessary retransmissions and result in a low messaging overhead.

Both ODAM and DDF are able to overcome network partitions successfully. However, ODAM's mechanism which periodically broadcasts the safety message failed to overcome a small number of partitions.

In terms of the time taken to propagate the message from the source node towards the forwarding boundary, both ODAM and flooding had a slightly lower propagation speed than DDF. The slightly longer propagation time for DDF occurs as a result of the longer retransmission deferral time which is at the expense of achieving an efficient and significant reduction in economy of messaging. The per hop latency of each protocol was found to be approximately 1 ms for both ODAM and flooding and 2 ms for DDF. Comparing the per hop latency with the critical latency values given in Table 2.1 for the safety related applications, which ranges from 50 ms - 100 ms, it can be seen that all protocols perform satisfactorily in this respect.

From the results of the performance evaluation, it can therefore be concluded that this research has satisfactorily accomplished the goal of supporting the dissemination of messages for safety related vehicular applications. Firstly, this research has ensured reliable and timely delivery of safety messages within a geographical area, secondly it has been ensured that the protocol adapts and scales to the varying traffic dynamics and thirdly, that an economy of messaging has been achieved whilst still maintaining reliable message delivery within the addressed geographic area.

8.3 Further Work

As with all simulation-based networking research, a wider range of scenarios need to be investigated at depth before a protocol such as the proposed one can be confidently adopted for actual use. In what follows, possible improvements are divided into two categories - those that increase the realism (e.g. propagation, shadowing, etc.), scope of the simulations (e.g. more involved road geometries) and those that improve on the functionality of the proposed protocol (e.g. temporal persistence, improved termination conditions).

Radio Propagation Model

The radio propagation model used in the simulation analysis is the two-ray flat ground model, whereby a packet is successfully received within the transmission range unless collisions occur at the MAC layer. This is essentially the unit disc graph model. Hence, this model is more likely to give a more optimistic performance measure in contrast to radio propagation models which model signal attenuation caused by multi-path fading or shadowing in the vehicular environment. Consequently, further simulation analysis is advisable using a more realistic propagation model which models signal degradation on a number of length and time scales in the vehicular environment.

Irrespective of the location of the channel fade within the FZ of a node, a missed intermediate node does not present a problem for forwarding process. However, if such an intermediate node is still located in the fade when the next retransmitting node broadcasts the message, this will affect the delivery rate. If a number of forwarding nodes are positioned in a channel fade then it is possible that the end-to-end communication delay will be increased since the forwarding process will resume from an intermediate node retransmission after the deferral time has expired. Such phenomena, especially for persistent, extended fades such as the ones present in shadowing, merit further careful investigation.

IEEE 802.11p PHY Layer

The channel access model used in the simulation analysis was based on IEEE 802.11b which differs in many aspects from the IEEE 802.11p scheme currently being standardised for cooperative vehicular communication. The IEEE 802.11p OFDM based modulation scheme has been implemented to mitigate against multipath effects and therefore, the impact of employing a more realistic propagation model would counter the effects that could otherwise have been observed in the relatively simplistic simulations in this work, in terms of increasing frame error ratios and thus increasing end-to-end delay and reducing successful delivery ratios. Further improvements to the performance of the PHY layer could be achieved by employing MIMO space-time coding in conjunction with OFDM. The trade-off of increased system complexity versus reliability gains, merits further investigation.

IEEE 802.11p MAC Layer

Although IEEE 802.11p operates using channelisation (control and service channels) and priority queues, safety messages from event driven applications and beacon messages are both transmitted on the control channel. This implies that the underlying CSMA channel access protocol is still common between the two IEEE 802.11 variants for the safety critical messages, and still provides a valid comparative performance analysis of the simulated protocols[1]. However, using IEEE 802.11p in the simulations will provide a more realistic platform meeting currently accepted technology, but more importantly a multi-channel version of the DDF protocol could be created and investigated at length.

Effect on Performance of Overlapping Event-driven RZR Regions

In the simulations this research evaluated the performance metrics of the DDF protocol during the dissemination of a safety message from one source vehicle. In reality, there will be multiple vehicles originating safety messages which define *DDAs* which will partially overlap. This will mean that the areas where the *DDAs* overlap will incur increased local demand on the communications channel. To the best of our knowledge, protocol evaluations within the literature only assess protocol performance during the dissemination of event based messages from one source. Consequently, the performance of DDF and in particular the ability of the protocol to adapt the retransmission deferral time should be investigated further.

Performance in Urban Scenarios

The DDF protocol has only been evaluated so far within a highway environment. Further simulations which assess performance in the urban road network environment are recommended. This will require further refinement to the relative positioning classification algorithm in order to cope with the classification of vehicles in more complicated road geometries such as junctions and intersections. Additionally, realistic traffic flow patterns reflecting the road network in an urban environment will be required to assess the performance of the DDF protocol.

Forwarding Boundary Termination Conditions

It is recommended that the termination conditions at the forwarding boundary are further improved in the case of the DDF protocol to prevent the isolated occurrences of local collisions causing slight dips in the delivery ratio, as shown by the max-min error bars. This could be achieved by defining a region around the forwarding boundary which allows nodes within this area to retransmit the message or alternatively allow the forwarding

[1]It is worth mentioning that at the time the DDF simulation model was created IEEE 802.11p did not exist, but it is fortuitous that the simulated results are still valid for the safety critical messages that this research is interested in.

node which detects the forwarding boundary within its *LZ* to schedule a maximum of two retransmissions.

Temporal Message Persistence

Within the DDF protocol forwarding persistence was implemented which ensures that the dissemination process does not terminate prematurely prior to the message reaching the boundary of the dissemination area; the dissemination process only terminates when the boundary is reached. Message persistence was only evaluated within the addressed area for the length of time it took for the dissemination process to reach the forwarding boundary. This is because this research was only interested in evaluating performance mechanisms to cover the *DDA* in a timely and reliable manner. The DDF protocol would further benefit from temporal persistence. Temporal persistence allows messages to be delivered to vehicles entering the *DDA* whilst hazards still exist. This can be implemented using one of two methods. In the first method the source node periodically broadcasts the message and in the second method the message is kept alive at the border of the *DDA* and broadcast to new vehicles entering the *DDA*.

Adaption of Beaconing Load to Local Parameters

As can be seen from the literature survey in Chapter 3, congestion control for beacon messages generally focuses on maintaining a mean beaconing load (MBL) below a statically defined threshold level so that a bounded proportion of the bandwidth is available for event based messages. However, in the situation where there are overlapping dissemination areas, localised demand on the channel from event based messages may exceed the allocated proportion of the available bandwidth. Therefore, it is recommend that the the following mechanisms are integrated with DDF, and in each case relative performance should be evaluated for overlapping *DDAs*.

- Although investigations can be found in the literature which examine the effect of varying the beacon rate with respect to local traffic density and relative vehicle speed, it is recommended that such a mechanism is evaluated in conjunction with DDF.
- Implementing an adaptive mean beaconing level threshold which adapts to local vehicle density and event message demand. This would require an investigation into defining a minimum and maximum threshold level for beaconing messages to ensure that local connectivity data does not result in erroneous DDF decisions as a result of stale local connectivity information. An additional investigation in relation to adapting the MBL would be to allow beacon messages to use more of the bandwidth when congestion exceeds a maximum MBL under the scenario when demand on channel from event based messages is low. In this case extensive performance

evaluations should be carried out with non-overlapping *DDAs*.

Determining the Spread of Channel Access Delays in Real-time

Investigations are recommended to determine the spread of MAC access delays in real-time, calculated from observed vehicle density and data traffic intensity statistics within a moving time window. When the observed statistical sample size falls below a threshold level, which can no longer be relied upon for accurate MAC delay spread values, then values determined from the empirical study presented here should be used instead. Determining, the MAC access delay parameters in real-time as opposed to the empirical method proposed in this thesis could further improve the adaptability of the DDF retransmission timing.

Further theoretical analysis of MAC channel access delays in the vehicular environment is also an open research issue. Moreover, extending this to multichannel MAC, as in 802.11p, also merits further attention for both saturated and non-saturation conditions.

Further Development of the Dissemination/routing Schemes Proposed in the IVC Data Dissemination Framework

Implementation of the remaining protocols described in the dissemination framework in §2.4.2 will enable a comprehensive platform for the bench marking of IVC protocols. The performance of each individual protocol needs to be assessed in the presence of data traffic generated from other IVC applications.

APPENDIX A

IVC USE CASES

A.1 Use Case Actors

Name	Description
Vehicle_1	Vehicle initiating request
Vehicle_2	Vehicle immediately affected by vehicle_1
Driver_1	Driver initiating request process
Relevant_drivers	Drivers that are implicitly affected by a request and play a primary role in a request. These drivers are within the RZR.
Relevant_vehicles	Vehicles that are implicitly affected by a request and play a primary role in a request. These vehicles are within the RZR.
Infrastructure	Fixed communications network dedicated to ITS and IVC incorporates intelligence to coordinate manoeuvres, process requests and warn of incidents
Routing zone of relevance (RZR)	Variable sized region to which the message is relevant and is dependant on application requirements - region can vary upwards in size from the area covered by the transmission radius
Lead_vehicle	vehicle of a platoon of vehicles. If the control structure is decentralised the lead_vehicle coordinates and controls the running of the platoon. If the control structure is centralised then the infrastructure controls and coordinates the running of the platoon through the lead vehicle, which is the only vehicle to communicate with the infrastructure
Nth_vehicle	The last vehicle in a platoon
Temporary_lead_vehicle	Vehicle in a platoon which is temporally assigned as a lead_vehicle to allow a platoon to temporally split in order to let a vehicle either join or leave the convoy
Lead-1_vehicle	Vehicle immediately succeeding the lead_vehicle
Nth-1_vehicle	Vehicle immediately preceding the Nth_vehicle
Platoon_members	Driver initiating request process
Temporary_lead_vehicle	Vehicles that form a convoy of vehicles which are either controlled by the lead_vehicle or by the infrastructure via the lead_vehicle, depending on whether the control structure is either centralised or decentralised respectively

A.2 Use Case Titles

U1 Cooperative Driving (Major Use case)

U1.1 Lane Changing

- U1.1pad Decentralised, partly automated
- U1.1cfa Centralised, fully automated
- U1.1dfa Decentralised, fully automated

U1.2 Emergency Stop

- U1.2pad Decentralised, partly automated
- U1.2cfa Centralised, fully automated
- U1.2dfa Decentralised, fully automated

U1.3 Leaving Motorway

- U1.3pad Decentralised, partly automated
- U1.3cfa Centralised, fully automated
- U1.3dfa Decentralised, fully automated

U1.4 Warning Other Drivers of Obstructions

- U1.4pad Decentralised, partly automated
- U1.4cfa Centralised, fully automated
- U1.4dfa Decentralised, fully automated

U1.5 Driving Habits

- U1.5pad Decentralised, partly automated
- U1.5cfa Centralised, fully automated
- U1.5dfa Decentralised, fully automated

U2 Convoy Driving (Major Use Case)

U2.1 Convoy Formation

- U2.1.1 On the Motorway
  - U2.1.1cfa Centralised, fully automated
  - U2.1.1dfa Decentralised, fully automated
- U2.1.2 On the Slip Road
  - U2.1.2cfa Centralised, fully automated item U2.1.2dfa Decentralised, fully automated

U2.2 Joining a convoy

- U2.2.1 From an adjacent Lane
  - U2.2.1cfa Centralised, fully automated
  - U2.2.1dfa Decentralised, fully automated
- U2.2.2 From a Slip Road
  - U2.2.2cfa Centralised, fully automated
  - U2.2.2dfa Decentralised, fully automated
- U2.2.3 Same lane as Convoy
  - U2.2.3cfa Centralised, fully automated
  - U2.2.3dfa Decentralised, fully automated
- U2.2.4 Convoys Merging
  - U2.2.4cfa Centralised, fully automated
  - U2.2.4dfa Decentralised, fully automated

U2.3 Leaving a Convoy

- U2.3.1 Lead Vehicle
  - U2.3.1cfa Centralised, fully automated
  - U2.3.1dfa Decentralised, fully automated
- U2.3.2 Last vehicle in Convoy
  - U2.3.2cfa Centralised, fully automated
  - U2.3.2dfa Decentralised, fully automated
- U2.3.3 Vehicle Between Lead and Nth Vehicle
  - U2.3.3cfa Centralised, fully automated
  - U2.3.3dfa Decentralised, fully automated

U2.4 Convoy Break-up

- U2.4.1 Lead Vehicle signals
  - U2.4.1cfa Centralised, fully automated
  - U2.4.1dfa Decentralised, fully automated
- U2.4.2 Roadside Beacon Signals
  - U2.4.2cfa Centralised, fully automated
- U2.4.3 Another Platoon signals
  - U2.4.3cfa Centralised, fully automated
  - U2.4.3dfa Decentralised, fully automated

A.3 Use Cases

USE CASE # U1.1pad	**Lane changing**	
Goal	Driver wishes to make manoeuvre into the adjacent lane.	
Control Structure	Decentralised, partly automated.	
Scope & Level	Secondary case	
Successful End Condition	End condition achieved when driver_1 broadcasts 'manoeuvre complete' message.	
Failed End Condition	Vehicle_1 determines that manoeuvre is not feasible Request for manoeuvre declined	
Actors	Driver_1, Relevant_vehicles , Relevant_drivers, RZR	
ID Type	Will require communication with all vehicles within the RZR of vehicle_1, and the position of these vehicles in relation to the requesting vehicle. Unique **address ID** and **location information** required.	
Data priority	Medium	
Driving environment	Motorway, Rural and Urban Roads	
Trigger	Driver_1 requesting lane change	
Use Case Steps	**Step**	**Action**
	1	Driver_1 initiates request to change lane, through the man machine interface (MMI).
	2	Vehicle_1 broadcasts information to vehicles in its RZR.
	3	Each vehicle within the RZR of driver_1 will transmit an acknowledgement message containing; vehicle dynamics and location information.
	4	Vehicle_1 will determine the vehicles of immediate concern to the manoeuvre and whether the manoeuvre is feasible from the acknowledgement data.
	5	Vehicle_1 will transmit a broadcast addressed to all relevant_ vehicles including the necessary control information for each vehicle.
	6	Relevant_drivers will be asked to accept or decline manoeuvre.
	7	All relevant vehicles will broadcast their acceptance or rejection of participating in the manoeuvre to both vehicle_1 and their RZR (request message from vehicle_1 will contain a list of all relevant vehicles ID)
	8	Vehicle_1 will wait for all acknowledgements (within a specific timeframe)
	9	Vehicle_1 broadcasts a start command if all acknowledgements are received successfully.
	10	Each vehicles MMI displays/commands manoeuvre parameters.
	11	Vehicle_1 broadcasts manoeuvre start message
	12	Relevant_vehicles adjust speed accordingly
	13	Driver_1 begins manoeuvre
	14	Vehicle_1 broadcasts 'manoeuvre_complete' message
EXTENSIONS	**Step**	**Branching Action**
	1a	Potential message clash if a vehicle relevant to a manoeuvre (located in the middle lane) receives more than one messages from both adjacent lanes.
	1b	Accepts manoeuvre based on time stamp
	9a	If any of the relevant_vehicles decline taking part in the manoeuvre , Vehicle_1 will broadcast a cancellation message to abort the manoeuvre.
SUB-VARIATIONS		**Branching Action**
	1	Tbc
RELATED INFORMATION	Lane changing	
Priority:	Medium	
Process length	Tbc	
Frequency	<how often it is expected to happen>	

OPEN ISSUES	How many times is request for manoeuvre transmitted if; Driver/ECU determines manoeuvre not feasible Transmission failure occurs Any of the relevant_vehicles does not accept manoeuvre
...any other system control information...	Tbc
Subordinates	Tbc
Superordinates	Tbc

USE CASE # 1.1cfa	**Lane changing**	
Goal	Driver wishes to make manoeuvre into the adjacent lane.	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	End condition achieved when driver_1 broadcasts 'manoeuvre complete' message.	
Failed End Condition	Vehicle_1determines that manoeuvre is not feasible or request for manoeuvre declined.	
Actors	Vehicle_1, Relevant_vehicles, Infrastructure, RZR	
ID Type	Will require communication with all adjacent vehicles, and the position of these vehicles in relation to the requesting vehicle. **Vehicle ID** and **location information** required.	
Data priority	Medium - High (messages will be assigned high priority status during the manoeuvre).	
Driving environment	Motorway, Urban and Rural Roads	
Trigger	Driver_1 requesting lane change	
Use Case Steps	**Step**	**Action**
	1	Vehicle_1 transmits request to change lane to the infrastructure.
	2	Infrastructure transmits acknowledgement to vehicle_1.
	3	Infrastructure transmits a message addressed to vehicles in the geographical RZR to vehicle_1, requesting them to transmit vehicle dynamics information.
	4	Vehicles transmit the requested data to the infrastructure
	5	Infrastructure determines manoeuvre parameters such as speed, Yaw rate etc. for each vehicle, relevant to the manoeuvre.
	6	Infrastructure broadcasts control information to the relevant_vehicles.
	7	Relevant_vehicles send an acknowledgement back to the infrastructure.
	8	Infrastructure broadcasts start command when all acknowledgements have been received from the relevant_vehicles.
	9	MMI warns driver that the vehicle is about to make a manoeuvre
	10	Each relevant_vehicle broadcasts messages to the infrastructure when they start the manoeuvre.
	11	Each time the infrastructure receives a manoeuvre start message it transmits this information back to the relevant_vehicles (this enables the infrastructure to monitor and control the manoeuvre process)
	12	Each relevant_vehicle sends a manoeuvre complete message to the infrastructure when it has reached it's target position, which will enable other manoeuvre requests to be coordinated, by the infrastructure in the RZR to the current manoeuvre.
		Vehicle_1 sends an acknowledgement that it has completed the manoeuvre successfully to the infrastructure
EXTENSIONS	**Step**	**Branching Action**
	1a	Clash of messages if one of the relevant vehicles in middle lane receives messages from both adjacent lanes.

	5a	If the infrastructure determines that one vehicle declines or senses that not all vehicles in the RZR have communicated back, then the manoeuvre will not proceed with the manoeuvre
SUB-VARIATIONS		**Branching Action**
	1	<list of variation s>
RELATED INFORMATION	Lane changing – centralised, fully automated	
Priority:	Medium - high	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	How many times is request for manoeuvre transmitted if; a)) Infrastructure determines manoeuvre not feasible b)) Transmission failure occurs c)) Any of the relevant_vehicles do not accept to make the manoeuvre	
...any other system control information...	Tbc	
Subordinates	Tbc	
Superordinates	Tbc	

USE CASE # 1.1dfa	**Lane changing**	
Goal in Context	Driver wishes to make manoeuvre into the adjacent lane.	
Control	Decentralised, fully automated	
Scope & Level	Secondary case	
Success End Condition	End condition achieved when driver_1 broadcasts 'manoeuvre complete' message.	
Failed End Condition	Vehicle_1 determines that manoeuvre is not feasible or Request for manoeuvre declined	
Primary, Secondary Actors	Vehicle_1, RZR, Relevant_vehicles	
ID Type	Will require communication with all adjacent vehicles, and the position of these vehicles in relation to the requesting vehicle. **Vehicle ID** and **location information** required.	
Data priority	Medium - high (messages will be assigned high priority status during the manoeuvre).	
Driving environment	Motorway, Urban and Rural Roads	
Trigger	Driver_1 requesting lane change	
Use Case Steps	**Step**	**Action**
	1	Vehicle_1 broadcasts request to change lane to its RZR.
	2	Each vehicle within the RZR of vehicle_1 will transmit an acknowledgement back to vehicle_1.
	3	Vehicle_1 will determine from the acknowledgements, which vehicles are relevant to the manoeuvre, and transmit a message addressed to the relevant_vehicles requesting an acknowledgment of their ability to take part in the process.
	4	Relevant_vehicles will transmit acknowledgements to vehicle_1 including acceptance or rejection to take part in the manoeuvre.
	5	When vehicle_1 receives all the requested acknowledgements, Vehicle_1 will determine whether the manoeuvre is feasible
	6	Vehicle_1 broadcasts a message informing vehicles whether the manoeuvre will be executed or not (acknowledgement of this information is requested).
	7	When all acknowledgements are received, vehicle_1 will determine the relevant control information for each relevant_vehicle
	8	Vehicle_1 will transmit control information addressed to each of the relevant_vehicles.

	9	When vehicle_1 has received an acknowledgment from each of the relevant_vehicles it will transmit a manoeuvre start command
	10	As each vehicle begins it's manoeuvre it will transmit its intention addressed to each relevant_vehicle including vehicle_1.
	11	After each relevant_vehicle has made its manoeuvre it will transmit its status to all relevant_vehicles including vehicle_1
	12	When vehicle_1 has completed the manoeuvre it will transmit a manoeuvre process over message to its RZR.
EXTENSIONS	**Step**	**Branching Action**
	1a	Clash of messages if any relevant_vehicle in middle lane and receives messages from both adjacent lanes.
	1b	Accepts manoeuvre request based on time stamp
	6a	If vehicle_1 determines that the manoeuvre is not feasible the process ends here with vehicle_1 transmitting a cancellation message to its RZR.
SUB-VARIATIONS		**Branching Action**
	1	<list of variation s>
RELATED INFORMATION	Lane changing	
Priority:	Medium - high	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	How many times is request for manoeuvre transmitted if; a)) vehicle determines manoeuvre not feasible b)) transmission failure occurs c)) vehicle does not accept manoeuvre d)) vehicle critical to the manoeuvre has a faulty system, how will requesting vehicle determine that it has not received communication from this vehicle depend on other vehicles for this information depend on proximity sensors	
...any other system control information...	Tbc	
Subordinate	Tbc	
Superordinates	Tbc	

USE CASE U1.2pad	**Emergency Stop**	
Goal	To warn drivers/vehicles of potential accident	
Control Structure	Decentralised, partly automated	
Scope & Level	Secondary case	
Successful End Condition	Necessary avoidance tactics deployed	
Failed End Condition	Communication not received and processed in time, collision occurs	
Actors	Vehicle_1, vehicle_2, RZR, infrastructure	
ID Type	Will require communication with all adjacent vehicles, and the position of the sending vehicle. **Address ID** and location **information** necessary.	
Data priority	High	
Driving Environment	Motorway, Rural and Urban roads	
Trigger	Vehicle malfunction, accident, obstruction	
Use Case Steps	**Step**	**Action**
	1	Vehicle_ 1 executes an emergency stop
	2	Vehicle_1 broadcasts a warning to its RZR. The message will contain driver ID and location information.
	3	Vehicles in the RZR will process the message and determine if any preventative measures need to be taken.
	4	Vehicle_2 will warn the driver to take necessary action.

	5	Vehicle_ 2 will determine the distance to vehicle_1, if it determines that an impact will occur a signal will be sent to the airbag control module to pre-arm the airbags.
	6	Vehicle_2 will also broadcast a warning message to the vehicles in its RZR.
	7	If vehicle_2 calculates an imminent impact, the airbags will be deployed.
	8	Vehicle_2 will broadcast an impact warning message
EXTENSIONS	**Step**	**Branching Action**
	1a	Transmission of drivers intentions U3.2
	6a	May cause potential obstruction U3.1
SUB-VARIATIONS		**Branching Action**
	3a	If vehicle_2 is fitted with an ACC system then it will sense that the vehicle's speed has changed and activate the brakes automatically. The system may not know that an emergency stop has been executed and will need to increase the brake pressure or indicate to that driver intervention is required.
RELATED INFORMATION	Emergency stop	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Subordinates	Tbc	
Superordinates	Tbc	

USE CASE U1.2cfa	**Emergency Stop**	
Goal	To warn drivers/vehicles of potential accident	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Necessary avoidance tactics deployed	
Failed End Condition	Communication not received and processed in time, collision occurs	
Actors	Vehicle_1, vehicle_2, RZR, infrastructure	
ID Type	Will require communication with all adjacent vehicles, and the position of the sending vehicle. **Address ID, speed, and vehicle type** and **location information** necessary.	
Data priority	High priority	
Driving Environment	Motorways, Urban and Rural Roads	
Trigger	Vehicle malfunction, accident, obstruction	
Use Case Steps	**Step**	**Action**
	1	Vehicle_ 1 executes an emergency stop
	2	Vehicle_1 sends an Emergency message to the infrastructure of action taken
	3	Infrastructure broadcasts a warning to the RZR of vehicle_1 warning vehicles. The message will contain driver ID and location information.
	4	Vehicles will transmit their positions to the infrastructure.
	3	Vehicles in the RZR will process the message and determine if any preventative measures need to be taken.
	4	Vehicle_2 will warn the driver that intervention maybe required.
	5	Vehicle_ 2 will determine the distance to vehicle_1, if it determines that an impact will occur a signal will be sent to the airbag control module to pre-arm the airbags.

	6	If vehicle_2 calculates an imminent impact, the airbags will be deployed.
	7	Vehicle_2 will broadcast an impact warning message to the infrastructure
	8	Infrastructure will notify the emergency services and broadcast emergency information the vehicles in the RZR of Vehicle_2
EXTENSIONS	**Step**	**Branching Action**
	4a	The infrastructure may determine any potential impacts and the necessary avoidance information for each vehicle instead of the vehicle. Each vehicle would have to wait for the control information from the infrastructure before taking any action.
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Emergency stop	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Subordinates	Tbc	
Superordinates	Tbc	

USE CASE 1.2dfa	**Emergency Stop**	
Goal	To warn drivers/vehicles of potential accident	
Control Structure	Decentralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Necessary avoidance tactics deployed	
Failed End Condition	Communication not received and processed in time, collision occurs	
Actors	Vehicle_1, vehicle_2, RZR, infrastructure	
ID Type	Will require communication with all adjacent vehicles, and the position of the sending vehicle. **Address ID** and location **information** necessary.	
Data priority	High priority	
Driving Environment	Motorways, Urban and Rural Roads	
Trigger	Vehicle malfunction, accident, obstruction	
Use Case Steps	**Step**	**Action**
	1	Vehicle_ 1 executes an emergency stop
	2	Vehicle_1 broadcasts a warning to the RZR. The message will contain driver ID and location information.
	3	Vehicles in the RZR will process the message and determine if any preventative measures need to be taken.
	4	Vehicle_2 will warn the driver of situation
	5	Vehicle_ 2 will determine the distance to vehicle_1, if it determines that an impact will occur a signal will be sent to the airbag control module to pre-arm the airbags.
	6	Vehicle_2 will also broadcast a message to the vehicles in its RZR warning drivers of any action it will take.
	7	If vehicle_2 calculates an imminent impact, the airbags will be deployed.
	8	Vehicle_2 will broadcast an impact warning message emergency message.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Emergency stop	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	

Channels to actors	Tbc
OPEN ISSUES	Tbc
...any other system control information...	Tbc
Superordinates	Tbc
Subordinates	Tbc

USE CASE 1.3pad	**Leaving motorway**	
Goal	Exit motorway	
Control Structure	Decentralised, partly automated	
Scope & Level	Secondary case	
Successful End Condition	Requesting vehicle enters slip road	
Failed End Condition	Vehicle not able to exit motorway at required junction	
Actors	Vehicle_1, RZR, infrastructure	
ID Type	ID not necessary, location information required	
Data priority	Medium - low dependant on the position of the vehicle in relation to the exit	
Driving Environment	Motorway and Dual carriage ways	
Trigger	Driver wishes to exit at the next junction	
Use Case Steps	**Step**	**Action**
	1	Vehicle following route already pre-programmed by the driver, vehicle will know when the exit junction is approaching.
		Vehicle_1 will inform the driver that he should exit at next junction
	2	Vehicle_1 will transmit its intention to the vehicles within its RZR, of its intention to leave at the next exit.
	3	Vehicle _1 will coordinate any necessary manoeuvres to achieve end condition
	4	Vehicle_1 will inform the driver when to start exiting the motorway.
EXTENSIONS	**Step**	**Branching Action**
	3a	If driver is not currently in the outside lane uses Lane Changing use case # 1.2
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Leaving the motorway	
Priority:	Medium	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Superordinates	1.2pad	

USE CASE 1.3cfa	**Leaving motorway**
Goal	Exit motorway
Control Structure	Centralised, fully automated
Scope & Level	Secondary case
Successful End Condition	Requesting vehicle enters slip road
Failed End Condition	Vehicle not able to exit motorway at required junction

Actors	Vehicle_1, RZR, infrastructure	
ID Type	ID not necessary, location information required	
Data priority	Medium - low dependant on the position of the vehicle in relation to the exit	
Driving Environment	Motorways and dual carriage ways	
Trigger	Driver wishes to exit at the next junction	
Use Case Steps	**Step**	**Action**
	1	If the route has been pre-programmed via the Internet etc and journey information is being transmitted to the vehicle, as it passes each roadside beacon, vehicle_1 will be told when to leave the motorway. The MMI would interact with driver to leave at next junction. A message would be broadcast to other vehicles at this point
	2	Vehicle_1 will transmit its intention to leave at the next exit.
	3	The infrastructure requests ID information (inc. speed & location) of all vehicles within the RZR of vehicle_1.
	4	The infrastructure will transmit the necessary control information which will allow vehicle_1 to make the manoeuvre off the motorway
	5	The infrastructure will transmit any necessary control information to the vehicles in the RZR of vehicle_1, which will enable vehicle_1 to exit at the requested junction.
EXTENSIONS	**Step**	**Branching Action**
	4a	If driver is not currently in the outside lane uses Lane Changing use case # 1.2
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Leaving the motorway	
Priority:	Medium	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...anyother system control information...	Tbc	
Subordinates	1.2cfa	
Superordinates	Tbc	

USE CASE # 1.3dfa	**Leaving motorway**	
Goal	Exit motorway	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Requesting vehicle enters slip road	
Failed End Condition	Vehicle not able to exit motorway at required junction	
Actors	Vehicle_1, RZR, infrastructure	
ID Type	ID not necessary, location information required	
Data priority	Medium - low dependant on the position of the vehicle in relation to the exit	
Driving Environment	Motorways and dual carriage ways	
Trigger	Driver wishes to exit at the next junction	
Use Case Steps	**Step**	**Action**
	1	If the route has been pre-programmed via the Internet etc and journey information is being transmitted to the vehicle, as it passes each roadside beacon, vehicle_1 will be told when to leave the motorway. The MMI would interact with driver to leave at next junction. A message would be broadcast to other vehicles at this point
	2	Vehicle_1 will transmit its intention to leave at the next exit.

	3	The infrastructure requests ID information (inc. speed & location) of all vehicles within the RZR of vehicle_1.
	4	The infrastructure will transmit the necessary control information which will allow vehicle_1 to make the manoeuvre off the motorway
	5	The infrastructure will transmit any necessary control information to the vehicles in the RZR of vehicle_1, which will enable vehicle_1 to exit at the requested junction.
EXTENSIONS	**Step**	**Branching Action**
	4a	If driver is not currently in the outside lane Lane Changing use case # 1.2dfa
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Leaving the motorway	
Priority:	Medium	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES		
...any other system control information...	Tbc	
Subordinates	1.2dfa	
Superordinates	Tbc	

USE CASE 1.4pad	**Warning Drivers of obstructions**	
Goal	To warn drivers of any obstructions	
Control Structure	Decentralised, partly automated	
Scope & Level	Secondary case	
Successful End Condition	Obstruction avoided or warning received Evasive action taken.	
Failed End Condition	Obstruction becomes a problem or accident caused	
Primary, Secondary Actors	vehicle_1, vehicle_2, Vehicle, RZR, infrastructure	
ID Type	Vehicle_ID, location information, speed, vehicle type	
Data priority	High	
Driving Environment	Motorway, Rural and Urban Roads	
Trigger	Received warning, driver sees obstruction.	
	Step	**Action**
	1	Vehicle_1 receives a warning message.
	2	Vehicle_1 determines that the obstruction is in its path.
	3	Vehicle_1 requests Vehicle_ID, Vehicle_type, speed and location information from the vehicles within its RZR.
	3	Vehicles in the RZR transmit requested data.
	4	Vehcle_1 determines necessary action it needs to take in order to avoid the obstruction.
	5	Vehicle_1 broadcasts its manoeuvre information to the vehicles within its RZR.
	6	Driver informed when and what action he/she is required to take.
	7	If manoeuvre not successful vehicle_1 will transmit an emergency warning message.
EXTENSIONS	**Step**	**Branching Action**
	4a	Infrastructure may determine that use cases U1.1pad, U1.2pad, U1.3pad need to be employed in order to avoid the obstruction.
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Warning drivers of obstructions	

Priority:	High
Process length	Tbc
Frequency	Tbc
OPEN ISSUES	Tbc
...any other system control information...	Tbc
Subordinates	1.1pad, 1.2pad, 1.3pad
Superordinates	Tbc

USE CASE 1.4cfa	**Warning Drivers of obstructions**	
Goal	To warn drivers of any obstructions	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Obstruction avoided or warning received Evasive action taken.	
Failed End Condition	Obstruction becomes a problem or accident caused	
Primary, Secondary Actors	vehicle_1, RZR, infrastructure	
ID Type	Vehicle_ID, location information, speed, vehicle type	
Data priority	High	
Driving Environment	Motorway, Urban and Rural Roads	
Trigger	Received warning, driver sees obstruction.	
	Step	**Action**
	1	Vehicle_1 receives a warning message from the infrastructure
	2	Vehicle_1 determines that the obstruction is in its path and transmits vehicle_ID, location information, speed and vehicle type to the infrastructure.
	3	Infrastructure determines the necessary control information for vehicle_1 to avoid obstruction
	4	Vehcle_1 acknowledges receipt of control message
	5	Infrastructure transmits message to vehicle_1's RZR warning of manoeuvre
	6	Vehicle_1 broadcasts status after control data executed
EXTENSIONS	**Step**	**Branching Action**
	3a	Infrastructure may determine that use cases, 1.1cfa, 1.2cfa, 1.3cfa need to be employed in order to avoid the obstruction
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Warning drivers of obstructions	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Subordinates	Tbc	
Superordinates	Tbc	

USE CASE 1.4dfa	**Warning Drivers of obstructions**
Goal	To warn drivers of any obstructions

Control Structure	Decentralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Obstruction avoided or warning received Evasive action taken.	
Failed End Condition	Obstruction becomes a problem or accident caused	
Primary, Secondary Actors	vehicle_1, RZR, infrastructure	
ID Type	Vehicle_ID, location information, speed, vehicle type	
Driving Environment	Motorway, Urban and Rural Roads	
Data priority	High	
Trigger	Received warning, driver sees obstruction.	
	Step	**Action**
	1	Vehicle_1 receives a warning message.
	2	Vehicle_1 determines that the obstruction is in its path.
	3	Vehicle_1 requests Vehicle_ID, Vehicle_type, speed and location information from the vehicles within its RZR.
	3	Vehicles in the RZR transmit requested data.
	4	Vehcle_1 determines necessary action it needs to take in order to avoid the obstruction.
	5	Vehicle_1 broadcasts its manoeuvre information to the vehicles within its RZR.
	6	Driver informed of action about to be taken.
	7	If manoeuvre not successful vehicle_1 will transmit an emergency warning message.
EXTENSIONS	**Step**	**Branching Action**
	4a	Infrastructure may determine that use cases, 1.1dfa, 1.2dfa, 1.3dfa need to be employed in order to avoid the obstruction.
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Warning drivers of obstructions	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Subordinates	1.1dfa, 1.2dfa, 1.3dfa	
Superordinates	Tbc	

USE CASE 1.5pad	**Driving habits**
Goal	To communicate erratic driving behaviour to other drivers
Control Structure	Decentralised, partly automated
Scope & Level	Secondary case
Success End Condition	Accidents avoided, safe driving maintained
Failed End Condition	Collision occurs, driving becomes hazardous
Primary, Secondary Actors	Vehicle_1, Driver_1, Relevant_vehicles, RZR
ID Type	**Vehicle_ID**, **Location coordinates**, **Vehicle_type**, **Vehicle_speed**
Driving Environment	Motorway, Urban and Rural Roads
Data priority	Medium - high

Trigger	Erratic driving (driver uses either the brake, clutch or throttle erratically) Vehicle reversing Driver using incorrect indicators contrary to vehicle Yaw rate etc. Head-up display senses driver fatigue Vehicle diagnostic system senses system malfunction	
	Step	**Action**
	1	Vehicle_1 dynamics controller (VDC(monitoring system parameters)) decides that brake, clutch or throttle is used erratically or driver operates the vehicle in opposition to external indicators.
	2	Vehicle_1 indicates to driver_1 through the MMI that they are driving erratically.
	3	If driving behaviour does not change, then vehicle_1 will broadcast a message to its RZR.
	4	Relevant_vehicles will process the message and display warning through the MMI to the driver.
	5	Each relevant_vehicle will determine if any action is required
	6	Each relevant vehicle displays to the driver any actions he/she may be required to take.
EXTENSIONS	**Step**	**Branching Action**
	1a	Vehicle may contain a head-up sensor, which senses driver fatigue. If the system senses that the driver is falling asleep the system will transmit a message to its RZR warning drivers to be aware.
	1b	The vehicle may have malfunctioned or breakdown anticipated.
	1b.1	Vehicle will switch to manual mode
	1b.2	Broadcast warning message to the RZR to alert brake down services.
	5a	A collision avoidance manoeuvre may need to be executed using one of the following use cases: 1.1pad, U1.2pad 1.4pad
SUB-VARIATIONS		**Branching Action**
	1	
RELATED INFORMATION	Driving habits	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Subordinates	1.1pad, U1.2pad 1.4pad	
Superordinates	Tbc	

USE CASE 1.5cfa	**Driving habits**
Goal	To communicate erratic driving behaviour to other drivers
Control Structure	Centralised, fully automated
Scope & Level	Secondary case
Successful End Condition	Accidents avoided, safe driving maintained
Failed End Condition	Collision occurs, driving becomes hazardous
Actors	Vehicle_1, RZR, Infrastructure
ID Type	Vehicle_ID, Location coordinates, Vehicle_type, Vehicle_speed
Data priority	Medium - high
Driving Environment	Motorway, Rural and Urban Roads

Trigger	Erratic driving (driver uses either the brake, clutch or throttle erratically) Vehicle reversing Driver using incorrect indicators contrary to vehicle Yaw rate etc. Head-up display senses driver fatigue Vehicle diagnostic system senses system malfunction	
	Step	**Action**
	1	Vehicle_1 dynamics controller (VDC(monitoring system parameters)) that the vehicle is being driven in an erratic manor by the automatic drive system.
	2	Vehicle_1 attempts to diagnose the fault.
	3	If system detects a malfunction, vehicle_1 will transmit a warning message to the infrastructure.
	4	Infrastructure will transmit a message to the vehicles in the RZR of vehicle_1 warning them of the vehicle and its location.
	5	If vehicle_1 decides that it needs to make a particular manoeuvre to avoid an incident, it will broadcast this information to the infrastructure.
	6	Infrastructure will relay vehicle_1's action to the vehicle's in its RZR
	5	Vehicles within the RZR will take necessary action to avoid an incident.
EXTENSIONS	**Step**	**Branching Action**
	1a	Vehicle_1 may have received a message warning from the infrastructure that it is driving erratically or externally indicators are inoperative.
	1b	Vehicle may contain a head-up sensor, which senses driver fatigue. If the system senses that the driver is falling asleep the system will transmit a message to the infrastructure who will relay the message to the vehicles in its RZR warning drivers to be aware.
	1c	The vehicle may have malfunctioned or breakdown anticipated.
	1c.1	Vehicle will switch to manual mode .
	1c.2	Broadcast warning message to the RZR Brake down services alerted.
	5a	A collision avoidance manoeuvre may need to be executed using on of the following use cases: 1.1cfa, U1.2cfa, U1.4cfa
SUB-VARIATIONS		**Branching Action**
	1	If vehicle is about to make a manoeuvre it will broadcast its intention to its RZR, i.e change lane, reverse etc
RELATED INFORMATION	Driving habits	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Superordinates	1.1cfa, U1.2cfa, U1.4cfa	

USE CASE 1.5dfa	**Driving habits**
Goal	To communicate erratic driving behaviour to other drivers
Control Structure	Decentralised, fully automated
Scope & Level	Secondary case
Successful End Condition	Accidents avoided, safe driving maintained
Failed End Condition	Collision occurs, driving becomes hazardous
Actors	Vehicle_1, RZR, Infrastructure
ID Type	Vehicle_ID, Location coordinates, Vehicle_type, Vehicle_speed
Data priority	Medium - high
Driving Environment	Motorway, Urban and Rural Roads

Trigger	Erratic driving (driver uses either the brake, clutch or throttle erratically) Vehicle reversing Driver using incorrect indicators contrary to vehicle Yaw rate etc. Head-up display senses driver fatigue Vehicle diagnostic system senses system malfunction	
	Step	**Action**
	1	Vehicle_1 dynamics controller (VDC(monitoring system parameters)) that the vehicle is being driven in an erratic manor by the automatic drive system.
	2	Vehicle_1 attempts to diagnose the fault
	3	If system detects a malfunction, vehicle_1 will warn the vehicles within its RZR that it is experiencing problems and to drive with caution.
	4	Relevant_vehicles will process the message.
	5	Each relevant_vehicle will determine if any action is required.
EXTENSIONS	**Step**	**Branching Action**
	1a	Vehicle_1 may have received a message warning the vehicle that it is driving erratically or externally indicators are inoperative.
	1b	Vehicle may contain a head-up sensor, which senses driver fatigue. If the system senses that the driver is falling asleep the system will transmit a message to its RZR warning drivers to be aware.
	1c	The vehicle may have malfunctioned or breakdown anticipated.
	1c.1	vehicle will switch to manual mode
	1c.2	Broadcast warning message to the RZR Brake down services alerted.
	5a	A collision avoidance manoeuvre may need to be executed using on of the following use cases: 1.1dfa, 1.2dfa, 1.4dfa
SUB-VARIATIONS		**Branching Action**
	1	If vehicle is about to make a manoeuvre it will broadcast its intention to its RZR, i.e change lane, reverse etc
RELATED INFORMATION	Driving habits	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
Channels to actors	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Superordinates	Tbc	
Subordinates	1.1dfa, 1.2dfa, 1.4dfa	

USE CASE 2.1.1cfa	**Convoy formation on the motorway**	
Goal	Formation of a fully automated train of vehicles	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Train of vehicles formed	
Failed End Condition	Formation did not occur	
Actors	Lead_vehicle, Nth_vehicle, Infrastructure	
ID Type	Vehicle_ID, location coordinates, vehicle_type	
Data priority	Requesting formation - Medium Inter-platoon communication - High Infrastructure to lead vehicle - High	
Driving Environment	Motorways initially	
Trigger	Infrastructure assigns a lead vehicle	
Use Case Steps	**Step**	**Action**
	1	Vehicles transmit their intention to join a convoy to the infrastructure.
	2	Infrastructure determines which vehicles within a particular geographic area are able to form a convoy. Infrastructure coordinates the movements of the vehicles to bring them into the same lane and ready to form a convoy.
	3	Infrastructure attempts to assign a lead vehicle and transmits a message to this vehicle.
	4	The elected lead_vehicle either accepts or declines the position of lead vehicle.
	5	Infrastructure assigns an inter-platoon communication channel to the lead vehicle and member vehicles.
	6	Infrastructure transmits convoy-operating parameters to the lead vehicle and member vehicles.
	7	Vehicles now in convoy position. Lead vehicle transmits a message containing operational speed and convoy operating parameters to the vehicle behind.
	8	Next_vehicle adds its own vehicle_ID and operational parameters to the message than transmit this to the next vehicle. This process continues until the nth vehicle is reached.
	9	Nth_vehicle forwards the message to the lead vehicle, reversing the path (vehicles forward the message only).
	10	Lead_vehicle processes received information to determine platoon members and transmits this to the infrastructure.
	11	Functioning of the platoon is maintained by repeating steps 7 – 11.
	12	Any changes in the operating speed of the convoy is controlled from the infrastructure (who coordinates all platoon movements) and communicated to the lead_vehicle
	13	Any change is communicated to the platoon members at step 7.
EXTENSIONS	**Step**	**Branching Action**
	1a	
SUB-VARIATIONS		**Branching Action**
	1a	Vehicle may decide it wants to lead a convoy and transmits it's request to the infrastructure.
	1b	Infrastructure will then broadcast a message addressed to vehicles in a restricted geographic region of the requesting vehicle.
	1c	Vehicles will transmits their acceptance to the infrastructure.
	4a	If the elected vehicle rejects the position the infrastructure will elect another vehicle.
RELATED INFORMATION	Convoy formation	
Priority:	High	
Process length	Tbc	

Frequency	Tbc
Channels to actors	Tbc
OPEN ISSUES	Tbc
...any other system control information...	Tbc
Superordinates	Tbc
Subordinates	Tbc

USE CASE 2.1.1dfa	**Convoy formation on the motorway**	
Goal	Formation of a fully automated train of vehicles	
Control Structure	Decentralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Train of vehicles formed	
Failed End Condition	Formation did not occur	
Actors	Lead_vehicle, nth_vehicle	
ID Type	Vehicle_ID, location coordinates, vehicle_type	
Data priority	Requesting formation - Medium Inter-platoon communication - High	
Driving Environment	Motorways Initially	
Trigger	Infrastructure assigns a lead vehicle	
Use Case Steps	**Step**	**Action**
	1	Vehicle transmits its intention to function as a convoy lead_vehicle to its RZR.
	2	Any accepting vehicles transmit a message back to the lead vehicle (message contains vehicle_ID, vehicle_type, location coordinates, vehicle speed.
	3	The lead vehicle determines the manoeuvres that each of these vehicles needs to make in order to bring them into the same lane to form the convoy. Message also contains the inter-platoon communication channel frequency. (Each vehicle will have been brought into the convoy in a sequence determined by its original location)
	4	As each vehicle moves into its convoy position it will send a message to the lead vehicle (via the vehicle in-front) informing it that it has joined the convoy.
	5	Each vehicle joining the convoy will receive a message containing the operational parameters of the convoy.
	6	A communications cycle will be set up between the lead vehicle and the nth vehicle forming a "round robin" cycle. Lead vehicle will send a message containing platoon speed, distance to be maintained between vehicles, each following vehicle will adjust speed and distance accordingly add its own information then forward to the next vehicle. This cycle will repeat until the message is received by the nth vehicle.
	7	Nth vehicle will forward this information to the lead vehicle
	8	Steps 6 to 7 are repeated periodically to maintain convoy speeds and distance.
EXTENSIONS	**Step**	**Branching Action**
	1a	
SUB-VARIATIONS		**Branching Action**
	3a	Use case U1.1 maybe used
RELATED INFORMATION	Convoy formation	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	

OPEN ISSUES	Tbc
...any other system control information...	Tbc
Subordinates	1.1dfa
Superordinates	Tbc

USE CASE 2.1.2cfa	**Convoy formation - on the same slip road**	
Goal	Formation of a fully automated train of vehicles before entry onto the motorway	
Control Structure	Centralised	
Scope & Level	Secondary case	
Successful End Condition	Train of vehicles formed on the slip road	
Failed End Condition	Formation did not occur	
Actors	Lead_vehicle, nth_vehicle, vehicle, Infrastructure	
ID Type	ID necessary, location information required	
Data priority	High priority	
Driving Environment	Motorway Slip Road Initially	
Trigger	Infrastructure assigns a lead vehicle	
	Step	**Action**
	1	Vehicles who wish to join a convoy are kept in a dedicated lane on the slip road. Infrastructure assigns the lead vehicle. The lead vehicle is chosen to be the vehicle at the front of the dedicated lane, every vehicle there after which is a multiple of the maximum platoon limit is assigned as a lead_vehicle.
	2	On receipt of the acknowledgment from the lead vehicles the infrastructure assigns an inter-platoon communication channel to each lead vehicle.
	3	The lead vehicle set's up a chain of communication to the nth_ vehicle to obtain the vehicle_ID and location of each member vehicle.
	4	Lead vehicle communicates convoy members location and vehicle ID to the infrastructure.
	5	Infrastructure determines the location of platoons/vehicles within range of the slip road
	6	Infrastructure will adjust the speed of any approaching platoons/vehicles to allow the platoon on the slip road to enter.
	7	Infrastructure will inform the lead_vehicle when to enter the motorway and what the platoon functional parameters will be.
	8	Lead_vehicle will communicate the functional operating parameters to the platoon.
	9	Lead_vehicle waits to receive a message from the nth vehicle which contains an acknowledgement from each vehicle that it has received the platoon functional parameters
	10	Infrastructure commands lead_vehicle to enter motorway
	11	Each vehicle will communicate both lead vehicle and it's own functional information to the following vehicle.
	12	When the nth vehicle receives the lead vehicle's information, it transmits the message back to the lead vehicle
	13	Cycle of communication from the lead vehicle to the nth vehicle will occur periodically as in steps 11 – 12
	14	Infrastructure will coordinate speed and any manoeuvres which need to be made with the lead_vehicle
EXTENSIONS	**Step**	**Branching Action**
	6a	Infrastructure will instruct the platoon to wait until the next platoon goes by.

SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Convoy formation	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Subordinates	Tbc	
Superordinates	Tbc	

USE CASE 2.1.2dfa	**Convoy formation - on the same slip road**	
Goal	Formation of a fully automated train of vehicles before entry onto the motorway	
Control Structure	Decentralised	
Scope & Level	Secondary case	
Successful End Condition	Train of vehicles formed on the slip road and successfully enters the motorway	
Failed End Condition	Formation did not occur	
Actors	Lead_vehicle, Nth_vehicle, Infrastructure	
ID Type	Vehicle_ID, location coordinates, vehicle_speed, Vehicle_type	
Data priority	High priority	
Driving Environment	Motorway slip road initially	
Trigger	Vehicle assigns itself as the lead vehicle	
	Step	**Action**
	1	Vehicles who wish to join a convoy are kept in a dedicated lane on the slip road. The lead vehicle who is at the front of the dedicated lane will automatically assign itself as the lead vehicle every vehicle thereafter which is a multiple of the maximum platoon limit is assigned as a lead_vehicle.
	2	Vehicle broadcasts it's intention to operate as the lead vehicle including information on intra_platoon communication channel.
	3	Vehicles acknowledge message and confirms membership of the convoy.
	4	Communication cycle from the lead vehicle to the nth vehicle is initiated.
	5	Lead_vehicle interrogates passing vehicles or platoons to determine the current traffic flow on the motorway.
	6	Lead_vehicle determines when the platoon can enter the motorway.
	7	Lead_vehicle transmits the intra-platoon operational parameters.
	8	Lead_vehicle waits to receive the acknowledgement from the nth vehicle before commanding platoon to enter the motorway.
	9	Platoon enters motorway.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
	5a	Beacon located on each slip road each of which is connected to the TCC. The TCC through it's network of sensors will know the whereabouts of the traffic in relation to the slip road
	5b	The TCC could inform vehicles through VMS to change speed or lane.
RELATED INFORMATION	Convoy formation	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	

OPEN ISSUES	Is it possible for the lead vehicle to coordinate the movement of the platoon onto the motorway from the information gathered from interrogating passing vehicles This use case may have to relay on some sort of infrastructure to enable safe coordination of vehicles onto the motorway and therefore may not be achievable using a centralised control structure .
...any other system control information...	Tbc
Subordinates	Tbc
Superordinates	Tbc

USE CASE 2.2.1cfa	**Joining Convoy – from the adjacent lane**	
Goal	Vehicle in adjacent lane to platoon joins platoon	
Control Structure	Centralised	
Scope & Level	Secondary case	
Successful End Condition	Vehicle in adjacent lane to platoon joins platoon	
Failed End Condition	Vehicle not able to join platoon	
Actors	Lead_vehicle, Nth_vehicle, Infrastructure, Vehicle_1	
ID Type	Vehicle_ID, location coordinates, vehicle_speed, Vehicle_type	
Data priority	Medium priority	
Driving Environment	Motorways initially	
Trigger	Vehicle sends request to join platoon	
	Step	**Action**
	1	Vehicle_1 transmits message to the infrastructure requesting to join the nearest convoy.
	2	Infrastructure acknowledges message and determines where the nearest platoon is in relation to vehicle_1 and which platoons are not currently at their maximum limit.
	3	Infrastructure will transmit a message to vehicle_1, either accepting or declining request.
	4	Infrastructure will transmit a cruising speed to vehicle_1 (including any further manoeuvres), which needs to be maintained until it either catches up with the platoon or vice versa.
	5	Infrastructure periodically interrogates vehicle_1 for location and speed data.
	6	Infrastructure compares position of vehicle_1 and position of the platoon to the computed trajectory. If the infrastructure calculates that the intercept point is no longer viable it will alter the cruising speed of either the platoon or vehicle_1.
	7	When vehicle_1 approaches the intercept point the infrastructure will transmit the platoon functional information to vehicle_1.
	8	Infrastructure will instruct the platoon lead_vehicle to coordinate the platoon to allow for vehicle_1 to join the platoon at the computed position.
	8a	In order for a vehicle to join between the lead and nth vehicle the infrastructure will need to assign a temporary lead vehicle in order to allow the platoon to split and create a gap to allow the vehicle to enter the platoon. Infrastructure transmits platoon operational information including the ID of the temp_lead_vehicle, to the lead_vehicle.
	9	Lead vehicle forwards request to the elected team_lead_vehicle
	10	Temp_lead vehicle forwards acknowledgement message to the lead_vehicle.

	11	Lead_vehicle forwards message containing functional parameters of manoeuvre to be made.
	12	Lead_vehicle waits to receive acknowledgement from Nth_vehicle (message includes ack's from all platoon members).
	13	Lead_vehicle transmits ready message to the infrastructure
	14	Infrastructure requests that veicle_1 takes up intercept position (includes parameters to maintain intercept position if changed)
	15	Temp_lead_vehicle commands following vehicles to change to the temporally assigned communications channel.
	16	When temp_lead_vehicle receives ack from the its Nth vehicle it transmits platoon operational parameters to it s members
	17	Infrastructure transmits start command to both the Lead_vehicle and the temp_lead_vehicle.
	18	Both the lead_vehicle and the temp_lead_vehicle adjust platoon speeds
	19	When both platoons reach their target position and speed, they transmit an acknowledgement to the infrastructure.
	20	Infrastructure commands vehicle_1 to make manoeuvre into platoon (message includes platoon comm's channel & platoon functional parameters).
	21	Vehicle_1 sends message to preceding vehicle of its presence, which is forwarded to the lead_vehicle.
	22	Lead_vehicle will inform infrastructure that vehicle_1 has successfully joined the platoon.
	23	Infrastructure transmits new operational parameters to temp_lead_vehicle.
	24	Temp_lead_vehicle transmits parameters to the platoon
	25	Temp_lead_vehicle coordinates manoeuvre to join onto original platoon (comm's channel changed)
	26	Vehicle_1 senses communication from temp_lead_vehicle
	27	Vehicle_1 forwards message to lead_vehicle
	28	Lead_vehicle informs infrastructure that the manoeuvre has been completed successfully.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
	2a	Infrastructure will transmit a message to vehicle_1 declining the request if: Platoons are too far away from the requesting vehicle and the current traffic flow will not allow vehicle_1 to catch up or slow down in order to join a convoy. All platoons in the vicinity of vehicle_1 are at their maximum allowable limit.
	6a	If infrastructure now determines that it is impossible for vehicle_1 to merge with the target platoon, it will transmit a message to vehicle_1 aborting the manoeuvre. Vehicle_1 will need to make the request to join a platoon again.
	8a	Vehicle_1 joining end of platoon
	b	Vehicle_1 transmits a message to the infrastructure when it reaches the intercept point.
	c	Infrastructure will transmit message to the lead_vehicle informing it that vehicle_1 is ready to make the manoeuvre (any platoon operational parameters will be included in the message)
	d	Lead_vehicle transmits manoeuvre details and platoon operational parameters to the platoon.
	e	When Lead_vehicle receives ack from the Nth_vehicle (includes ack's from all platoon members), it transmits a ready message to the infrastructure.
	f	Infrastructure commands vehicle_1 to make the manoeuvre to join the platoon (includes any updated platoon functional information)

	g	When vehicle_1 is in position it sends a message to the platoon Nth vehicle informing of it's presence, which is forwarded to the lead_vehcicle.
		Lead_vehicle transmits message to the infrastructure that vehicle_1 has successfully joined the convoy and manoeuvre is accomplished.
	10a	If vehicle cannot accept role, lead_vehicle will notify infrastructure who will nominate another vehicle & provide new functional parameters.
	1	Infrastructure will transmit to the lead_vehicle, platoon functional information including the vehicle_ID of the temporary_lead_vehicle, temporary lead vehicle communication channel and any change in platoon speed.
	2	Lead_vehicle will forward manoeuvre
RELATED INFORMATION	Joining Convoy – from adjacent lane	
Priority:	Medium High	
Process length	Tbc	
Frequency	<how often it is expected to happen>	
OPEN ISSUES	Assumed that the requesting vehicle will merge into the platoon from an adjacent lane. Unnecessary amount of control information required for vehicle to join as the lead vehicle as the platoon reaches the requesting vehicle.	
...any other system control information...	Tbc	
Subordinates	Tbc	
Superordinates	Tbc	

USE CASE # 2.2.2cfa	**Joining Convoy – from the slip road**	
Goal	Vehicle on slip road joins platoon on motorway	
Control Structure	Centralised, fully automated	
Scope & Level	Tbc	
Successful End Condition	Vehicle in joins platoon	
Failed End Condition	Vehicle not able to join platoon	
Actors	Lead_vehicle, Nth_vehicle, Infrastructure, Vehicle_1, temp_lead_vehicle	
ID Type	Vehicle_ID, location coordinates, vehicle_speed, Vehicle_type	
Data priority	Medium priority	
Driving Environment	Motorway initially	
Trigger	Vehicle sends request to join platoon	
	Step	**Action**
	1	Vehicle_1 transmits message to the infrastructure requesting to join the a convoy.
	2	Infrastructure acknowledges message and determines where the nearest platoon is in relation to vehicle_1 and if any platoons are already at their maximum limit.
	3	Infrastructure will transmit a message to vehicle_1, either accepting or declining request.
	4	Infrastructure will transmit a message to the platoon which it has computed vehicle_1 is able to join.
	5	Usecase 2.1.1cfa steps 4 to 8 are followed to complete the manoeuvre.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**

	2a	Infrastructure will transmit a message to vehicle_1 declining the request if: There are no passing Platoons within a specific range All platoons in the vicinity of vehicle_1 are at their maximum allowable limit. Vehicle_1 will however be given the opportunity to either enter the motorway then re-request it joins a platoon using either use case U2.1.1cfa or U2.2.1cfa join the platoon holding lane to form a platoon using use case 2.1.1cfa
RELATED INFORMATION	Joining Convoy – from slip road	
Priority:	Medium	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Assumed that the requesting vehicle will merge into the platoon from an adjacent lane. Unnecessary amount of control information required for vehicle to join as the lead vehicle as the platoon reaches the requesting vehicle.	
...any other system control information...	Tbc	
Subordinates	2.1.1cfa, 2.2.1cfa	
Superordinates	Tbc	

USE CASE # 2.2.3cfa	**Joining Convoy – nth position**	
Goal	Vehicle joins platoon	
Control Structure	Centralised	
Scope & Level	Secondary case	
Successful End Condition	Vehicle joins platoon in the nth position	
Failed End Condition	Vehicle not able to join platoon	
Actors	Lead_vehicle, Nth_vehicle, Infrastructure, Vehicle_1, temp_lead_vehicle	
ID Type	Vehicle_ID, location coordinates, vehicle_speed, Vehicle_type	
Data priority	Medium priority	
Driving Environment	Motorways Initially	
Trigger	Vehicle sends request to join platoon	
	Step	**Action**
	1	Vehicle_1 transmits message to the infrastructure requesting to join a convoy.
	2	Infrastructure acknowledges message and determines where the nearest platoon is in relation to vehicle_1 and if any platoons are already at their maximum limit.
	3	Infrastructure will transmit a message to vehicle_1, either accepting or declining request.
	4	Infrastructure will compute any manoeuvres including the speed of vehicle_1, in order to reach the computed intercept point with the platoon. The infrastructure will transmit this data to vehicle_1.
	5	Infrastructure will update the platoons functional parameters in order to achieve the computed intercept position..
	6	Vehicle_1 transmits a message to the infrastructure when it reaches the intercept point.
	7	Infrastructure will transmit message to the lead_vehicle informing it that vehicle_1 is ready to make the manoeuvre (any platoon operational parameters will be included in the message)
	8	Lead_vehicle transmits manoeuvre details and platoon operational parameters to the platoon - if any.

	9	When Lead_vehicle receives ack from the Nth_vehicle (includes ack's from all platoon members), it transmits a ready message to the infrastructure.
	10	Infrastructure commands vehicle_1 to make the manoeuvre to join the platoon (includes any updated platoon functional information)
	11	When vehicle_1 is in position it sends a message to the platoon Nth_ vehicle informing it, of it's presence, which is forwarded to the lead_vehcicle.
	12	Lead_vehicle transmits message to the infrastructure that vehicle_1 has successfully joined the convoy and manoeuvre is accomplished.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
	3a	Infrastructure will transmit a message to vehicle_1 declining the request if: There are no passing Platoons within a specific range All platoons in the vicinity of vehicle_1 are at their maximum allowable limit.
RELATED INFORMATION	Joining Convoy – in the nth position	
Priority:	Medium	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Assumed that the requesting vehicle will merge into the platoon from an adjacent lane. Unnecessary amount of control information required for vehicle to join as the lead vehicle as the platoon reaches the requesting vehicle.	
...any other system control information...	Tbc	
Subordinates	Tbc	
Superordinates	Tbc	

USE CASE # 2.2.4cfa	**Platoons Merging**	
Goal	Platoons merge	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Platoons merge	
Failed End Condition	Platoons are not able to merge	
Actors	Lead_vehicle, Nth_vehicle, Infrastructure, Vehicle_1,	
ID Type	Vehicle_ID, location coordinates, vehicle_speed, Vehicle_type	
Data priority	Medium priority	
Driving Environment	Motorways initially	
Trigger	Vehicle sends request to join platoon	
	Step	**Action**
	1	Lead_vehicle transmits message to the infrastructure requesting to join with another convoy.
	2	Infrastructure acknowledges message and determines where the nearest platoon is in relation to the lead_vehicles convoy.
	3	Infrastructure will transmit a message to the requesting lead_vehicle, either accepting or declining request.

	4	Infrastructure will compute any manoeuvres including the speed of the requesting lead_vehicles platoon, in order to reach the computed intercept point. The infrastructure will transmit this data to the lead_vehicle.
	5	Infrastructure will update the target platoons functional parameters in order to achieve the computed intercept position..
	6	Requesting lead_vehicle transmits a message to the infrastructure when it reaches the intercept point.
	7	Infrastructure will transmit message to the lead_vehicle informing it that the requesting platoon is ready to join the platoon (any platoon operational parameters will be included in the message)
	8	Lead_vehicle transmits manoeuvre details and platoon operational parameters to the platoon - if any.
	9	When Lead_vehicle receives ack from the Nth_vehicle (includes ack's from all platoon members), it transmits a ready message to the infrastructure.
	10	Infrastructure commands the requesting lead_vehicle to make the manoeuvre to join the platoon (includes any updated platoon functional information)
	11	Requesting lead_vehicle will forward message to its members commanding them to change communication channel to the target platoons
	12	Requesting lead_vehicle informs infrastructure that it has changed communication channel and is about to adopt platoon member status.
	13	When the requesting lead_vehicle is in position it sends a message to the target platoon Nth_ vehicle informing it, of it's presence, which is forwarded to the lead_vehcicle.
	14	Any messages that are forwarded to the nth vehicle are now forwarded as the nth vehicles now detects that there are platoon members behind it.
	15	Lead_vehicle transmits message to the infrastructure that the requesting platoon has successfully joined the convoy and manoeuvre is accomplished.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
	3a	Infrastructure will transmit a message to vehicle_1 declining the request if: There are no passing Platoons within a specific range All platoons in the vicinity of vehicle_1 are at their maximum allowable limit.
RELATED INFORMATION	Convoys merging	
Priority:	Medium	
Process length	Tbc	
Frequency	<how often it is expected to happen>	
OPEN ISSUES	It is assumed that there would be a logistics problem in order to coordinate the motorway traffic to allow a platoon to merge between the lead and nth vehicles. Therefore the case where the platoon is manoeuvred behind the target platoon is considered only at this stage..	
...any other system control information...	Tbc	
Subordinates	Tbc	
Superordinates	Tbc	

USE CASE 2.3.1cfa	**Leaving convoy - lead vehicle**	
Goal	Lead vehicle leaves the platoon	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	New lead vehicle assigned	
Failed End Condition	(i) () Lead vehicle did not leave platoon when requested (ii) () platoon falls apart	
Actors	Lead_vehicle, lead-1_vehicle, Infrastructure	
ID Type	ID necessary, location coordinates, timing, vehicle dynamics	
Data priority	High priority	
Driving Environment	Motorway	
Trigger	Lead vehicle requests to leave platoon.	
	Step	**Action**
	1	Lead vehicle sends a message to the infrastructure requesting to leave the platoon.
	2	Infrastructure sends an acknowledgment to the lead vehicle.
	3	Infrastructure sends a message to the lead_vehicle which is forwarded to the lead-1_vehicle requesting it to take over the lead_vehicle position.
	4	Lead-1_ vehicle acknowledges request either accepting or declining the new role.
	5	Infrastructure sends lead_vehicle & lead-1_ vehicle manoeuvre information and platoon functional information respectively, to enable the lead_ vehicle to break away from the platoon.
	6	Both Lead-1_vehicle & lead _vehicle acknowledge receipt of functional information.
	7	Lead-vehicle transmits functional information to the platoon.
	8	Lead_vehicle waits for an acknowledgement from the nth vehicle then transmits a ready command to the infrastructure.
	9	Infrastructure sends a command to the lead vehicle and the lead-1_vehicle to start the manoeuvre.
	10	Lead_vehicle and lead-1_vehicle acknowledge request,
	11	Lead-1_vehicle transmits message to platoon of its new position
	12	Lead-1_vehicle waits to receive acknowledgement from Nth vehicle before informing infrastructure that it has successfully taken over as the lead_vehicle
	13	Infrastructure commands lead_vehicle to breakaway from the platoon.
EXTENSIONS	**Step**	**Branching Action**
	4a	If lead-1_vehicle declines role then the infrastructure will send a request for the platoon to break_up U2.5.2c
	5a	The infrastructure could send the platoon functional information to each vehicle in the platoon then wait for an acknowledgment from each platoon member before sending a start command.
	5a	Lead_vehicle may exit motorway U1.3
	5b	Lead_vehicle may change lane using U1.1
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Leaving convoy - lead vehicle	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	platoon manoeuvre use case needs to be addressed	
...any other system control information...	Tbc	
Superordinates	<optional, name of use case(s) that includes this one>	

Subordinates	U1.1cfa, U1.3cfa

USE CASE 2.3.1dfa	**Leaving convoy - lead vehicle**	
Goal	Lead vehicle leaves the platoon	
Control Structure	Decentralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	New lead vehicle assigned	
Failed End Condition	(i) () Lead vehicle did not leave platoon when requested (ii) () Platoon falls apart	
Actors	Lead_vehicle, lead-1_vehicle, Zonal_drivers	
ID Type	ID necessary, location coordinates, vehicle_dynamics	
Data priority	Medium to High priority	
Driving Environment	Motorways initially	
Trigger	Vehicle broadcasts it's intention to leave the platoon	
	Step	**Action**
	1	Lead vehicle broadcasts its intention to leave the platoon.
	2	Lead vehicle transmits its intention to handover lead status to the lead-1_vehicle.
	3	lead-1_vehicle acknowledges and accepts status.
	4	Lead vehicle transmits platoon control data and its intended manoeuvre to lead-1_vehicle.
	5	Lead_vehicle determines platoon functional information (which will allow it to make the desired manoeuvre). Lead_vehicle may use either use case U1.1, U1.2, U1.3, U1.4 to coordinate its manoeuvre and determine platoon functional information that will allow it to make the manoeuvre.
	6	Lead_vehicle forwards platoon functional information to the platoon.
	7	Lead_vehicle waits for an acknowledgment from the Nth_vehicle.
	8	Lead_vehicle forwards message to lead-1_vehicle to take over as the lead_vehicle.
	9	Lead-1_vehicle adjusts the platoons's speed (if required) to allow the vehicle to makes its manoeuvre.
	10	Lead_vehicle breaks away from the platoon
EXTENSIONS	**Step**	**Branching Action**
	3a	If lead-1_vehicle does not accept lead status then the lead vehicle will command a platoon break-up U2.3.1
	7a	Lead-1_vehicle may command the lead_vehicle to make its manoeuvre after it has received an acknowledgement from the nth vehicle.
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Leaving convoy - lead vehicle (decentralised control)	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	platoon manoeuvre use case needs to be addressed	
...any other system control information...	Tbc	
Superordinates	Tbc	
Subordinates	U1.1dfa, U1.3dfa	

USE CASE 2.3.2cfa	**Leaving convoy - Last vehicle**
Goal	Last vehicle leaves the platoon

Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Last vehicle in convoy leaves platoon	
Failed End Condition	(i) () Lead vehicle did not leave platoon when requested (ii) () platoon falls apart	
Actors	Lead_vehicle, Nth_vehicle, Nth-1_vehicle, platoon, Infrastructure	
ID Type	ID necessary, location coordinates, vehicle_dynamics	
Data priority	Medium to high priority	
Driving Environment	Motorway initially	
Trigger	Last vehicle in the platoon requests to leave platoon	
	Step	**Action**
	1	Nth_vehicle forwards message to the lead_vehicle requesting to leave the platoon.
	2	Lead_vehicle forwards message to the infrastructure
	3	Infrastructure sends acknowledgement to the lead_vehicle.
	4	Lead_vehicle forwards acknowledgement to the Nth_vehicle.
	5	Infrastructure broadcasts a message addressed to vehicles that are within the Nth_vehicles RZR, to determine if requested manoeuvre to leave platoon is possible. may coordinate its manoeuvre using either use cases U1.1 or U1.3.
	6	Infrastructure will forward message via the lead_vehicle to the Nth_vehicle either accepting or declining manoeuvre
	7	Nth_vehicle will send an acknowledgment via the lead_vehicle to the infrastructure.
	8	Infrastructure sends a message to the lead_vehicle which includes platoon functional information and the manoeuvre parameters for the Nth_vehicle
	9	Lead_vehicle acknowledges receipt of message
	10	Lead_vehicle forwards message to Nth_ containing parameters, which will allow it to make its manoeuvre.
	11	Lead_vehicle forwards platoon functional parameters to platoon excluding the Nth_vehicle
	12	When the lead_vehicle receives acknowledgments from all vehicles it will send a message to the infrastructure informing it that it is ready to release the Nth_vehicle.
	12	Infrastructure commands platoon to start manoeuvre
	11	The Nth_vehicle will start it's manoeuvre when it receives the start command from the lead_vehicle
	12	When the Nth-1_vehicle senses no communication from the Nth vehicle it sends a message to the lead_vehicle
	13	Lead_vehicle informs the infrastructure that the Nth vehicle has left the platoon.
EXTENSIONS	**Step**	**Branching Action**
	5a	May coordinate requested manoeuvre using use cases U1.1 or U1.3
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Leaving convoy - last vehicle (centralised control)	
Priority:	Medium to High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Tbc	
...any other system control information...	Tbc	
Superordinates	Tbc	
Subordinates	U1.1cfa, U1.3cfa	

USE CASE 2.3.2dfa	**Leaving convoy - Last vehicle**	
Goal	Last vehicle leaves the platoon	
Control Structure	Decentralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Last vehicle in convoy leaves platoon	
Failed End Condition	(i) () Lead vehicle did not leave platoon when requested (ii) () platoon falls apart	
Actors	Lead_vehicle, Nth_vehicle, Nth-1_vehicle, platoon	
ID Type	ID necessary, location coordinates	
Data priority	Medium to high priority	
Driving Environment	Motorway initially	
Trigger	Last vehicle in the platoon requests to leave platoon	
	Step	**Action**
	1	Nth_vehicle forwards message to the lead_vehicle requesting to leave the platoon.
	2	Lead_vehicle acknowledges request.
	3	Nth_vehicle may coordinate its manoeuvre using either use cases U1.1 or U1.3. It will determine manoeuvre parameters and forward them to the lead_vehicle
	4	Lead_vehicle will acknowledge receipt of message
	5	Lead_vehicle will transmit to the platoon any change in platoon operational parameters in order to help the Nth_vehicle to leave the platoon.
	6	When the lead vehicle receives replies back from the nth-1_vehicle it forwards a message to the platoon, which commands them to change speed etc.
	3	Platoon adjusts it's speed to allow the vehicle to makes its manoeuvre
	4	Nth-1_vehicle after receiving command to change parameters sends message to the Nth_vehicle commanding it to start its manoeuvre
	5	Nth_vehicle breaks away from the convoy
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Leaving convoy - last vehicle (decentralised control)	
Priority:	Medium to High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	platoon manoeuvre use case needs to be addressed	
...any other system control information...	Tbc	
Superordinates	Tbc	
Subordinates	U1.1dfa, U1.3da	

USE CASE # 2.3.3cfa	**Leaving Convoy – vehicle between the lead and nth vehicle.**
Goal	Vehicle between the lead vehicle and nth vehicle leaves the platoon
Control Structure	Centralised, fully automated
Scope & Level	Secondary case
Successful End Condition	Vehicle leaves platoon
Failed End Condition	Vehicle not able to leave the platoon
Actors	Lead_vehicle, Nth_vehicle, Infrastructure, Vehicle_1, temporary_lead_vehicle, RZR.
ID Type	Vehicle_ID, location coordinates, vehicle_speed, Vehicle_type,

Data priority	Medium to high priority	
Driving Environment	Motorway initially	
Trigger	Vehicle sends request to join platoon	
	Step	**Action**
	1	Vehicle_1 sends a message to the lead_vehicle requesting to leave the platoon.
	2	The lead_vehicle forwards request to the infrastructure
	3	Infrastructure will broadcast message addressed to vehicles within the RZR of vehicle_1 in order to coordinate requested manoeuvre.
	4	Infrastructure will send a message to vehicle_1 via the lead_vehicle either accepting or declining manoeuvre.
	5	In order for a vehicle to leave between the lead and nth vehicle the infrastructure will need to assign a temporary lead vehicle in order to allow the platoon to split and create a gap to allow the vehicle to leave the platoon. Infrastructure transmits platoon operational information including the ID of the temporary_lead_vehicle, to the lead_vehicle.
	6	Lead vehicle forwards request to the elected temporary_lead_vehicle
	7	Temp_lead vehicle forwards acknowledgement message to the lead_vehicle.
	8	Lead_vehicle forwards message containing functional parameters of manoeuvre to be made.
	9	Lead_vehicle waits to receive acknowledgement from Nth_vehicle (message includes ack's from all platoon members).
	10	Lead_vehicle transmits ready message to the infrastructure
	11	Infrastructure transmits command via the lead_vehicle to the temporary_lead_vehicle to take up its role and change communication channel
	12	Temporary_lead_vehicle commands following vehicles to change to the temporally assigned communications channel.
	13	When temporary_lead_vehicle receives ack from the its Nth vehicle it transmits platoon operational parameters to its members
	14	Infrastructure transmits start command to both the Lead_vehicle and the temporary_lead_vehicle.
	15	Both the lead_vehicle and the temporary_lead_vehicle adjust platoon speeds
	16	When both platoons reach their target position and speed, they transmit an acknowledgement to the infrastructure.
	17	Infrastructure commands vehicle_1 to make manoeuvre out of the platoon (message includes platoon comm's channel & platoon functional parameters).
	18	When the vehicle preceding vehicle_1 senses no communication from vehicle_1 it forwards a message to the lead vehicle.
	19	Lead_vehicle will inform infrastructure that vehicle_1 has successfully left the platoon.
	20	Infrastructure transmits new operational parameters to temporary_lead_vehicle.
	21	Tempoary_lead_vehicle transmits parameters to the platoon
	22	Temporary_lead_vehicle coordinates manoeuvre to join onto original platoon (comm's channel changed)
	23	Nth_vehicle senses communication from temp_lead_vehicle
	24	Nth_vehicle forwards message to lead_vehicle
	25	Lead_vehicle informs infrastructure that vehicle_1 has successfully left the platoon and the platoon has rejoined.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**

	3a	Infrastructure may need to use, use cases 1.1 or 1.3 in order to coordinate manoeuvre.
	4a	Infrastructure will transmit a message to vehicle_1 declining the request if it is not able to coordinate the manoeuvre with the vehicles in the RZR of vehicle_1. The infrastructure will periodically interrogate both the lead_vehicle and the vehicles, which are relevant to vehicle_1's manoeuvre. If it detects that there are any problems the infrastructure will abort the manoeuvre.
RELATED INFORMATION	Joining Convoy – from adjacent lane	
Priority:	Medium High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Assumed that the requesting vehicle will merge into the platoon from an adjacent lane. Unnecessary amount of control information required for vehicle to join as the lead vehicle as the platoon reaches the requesting vehicle.	
...any other system control information...	<...as needed>	
Subordinates	1.1cfa, 1.3cfa	
Superordinates	Tbc	

USE CASE 2.3.3dfa	**Leaving Convoy – vehicle between the lead and nth vehicle.**	
Goal	Vehicle between the lead vehicle and nth vehicle leaves the platoon	
Control Structure	Decentralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	Vehicle leaves platoon	
Failed End Condition	Vehicle not able to leave the platoon	
Actors	Lead_vehicle, Nth_vehicle, Vehicle_1, temporary_lead_vehicle, RZR.	
ID Type	Vehicle_ID, location coordinates, vehicle_speed, Vehicle_type,	
Data priority	Medium to high priority	
Driving Environment	Motorway Initially	
Trigger	Vehicle sends request to join platoon	
	Step	**Action**
	1	Vehicle_1 sends a message to the lead_vehicle requesting to leave the platoon.
	3	Lead_vehicle will broadcast message addressed to vehicles within the RZR of vehicle_1 in order to coordinate requested manoeuvre.
	4	Lead_vehicle will send a message to vehicle_1 via the either accepting or declining manoeuvre.
	5	In order for a vehicle to leave between the lead and nth vehicle the lead_vehicle will need to assign a temporary_ lead_ vehicle in order to allow the platoon to split and create a gap to allow vehicle_1 to leave the platoon. Lead_vehicle transmits platoon operational information including to the vehicle immediately following vehicle_1 who will act assume the role of temporary_lead_vehicle.
	7	Temp_lead vehicle forwards acknowledgement message to the lead_vehicle.
	8	Lead_vehicle forwards message containing functional parameters of manoeuvre to be made.
	9	Lead_vehicle waits to receive acknowledgement from Nth_vehicle (message includes ack's from all platoon members).

	11	lead_vehicle transmits command to the temporary_lead_vehicle to take up its role and change communication channel.
	12	Temporary_lead_vehicle commands following vehicles to change to the temporally assigned communications channel.
	13	When temporary_lead_vehicle receives ack from its Nth_vehicle it transmits platoon operational parameters to its members.
	14	Lead_vehicle transmits start command to the temporary_lead_vehicle.
	15	Both the lead_vehicle and the temporary_lead_vehicle adjust platoon speeds
	16	When the original platoon reaches its target speed and position the lead_vehicle sends a request to vehicle_1 asking if it is still hearing communication from the temporary_ lead_vehicle.
	17	When vehicle_1 no longer hears communication from the temporary_lead_vehicle it sends a message to vehicle_1 to make manoeuvre out of the platoon.
	18	When the vehicle preceding vehicle_1 senses no communication from vehicle_1 it forwards a message to the lead vehicle.
	19	The lead_vehicle slows the platoon operating speed down and requests that the Nth vehicle forwards a message as soon as it senses communication from the temporary lead vehicle.
	20	Nth_vehicle senses temporary lead_vehicle and forwards message to the lead_vehicle.
	21	The Lead_vehicle will transmit platoon functional information addressed to the temporary lead_vehicle
	22	The temporary lead vehicle will acknowledge message and transmit platoon functional information to its members.
	23	The temporary lead vehicle when it has received an acknowledgment from its Nth vehicle will change command its members to change back to the original communication channel.
	19	The temporary_lead_vehicle will send a message to the lead_vehicle informing it that it has successfully re-joined the platoon with all its members and will then resume its position as a member of the platoon.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
	3a	Infrastructure may need to use, use cases 1.1 or 1.3 in order to coordinate manoeuvre.
	4a	Infrastructure will transmit a message to vehicle_1 declining the request if it is not able to coordinate the manoeuvre with the vehicles in the RZR of vehicle_1. The infrastructure will periodically interrogate both the lead_vehicle and the vehicles, which are relevant to vehicle_1's manoeuvre. If it detects that there are any problems the infrastructure will abort the manoeuvre.
	16	If vehicle_1 still detects communication from the temporary_lead_vehicle, the lead_vehicle will keep interrogating vehicle_1 until it receives no communication. If after a certain period of time vehicle_1 is still receiving communication the lead_vehicle will abort the manoeuvre.
RELATED INFORMATION	Leaving Convoy – between lead and nth vehicle.	
Priority:	Medium to High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	Assumed that the requesting vehicle will merge into the platoon from an adjacent lane. Unnecessary amount of control information required for vehicle to join as the lead vehicle as the platoon reaches the requesting vehicle.	
...any other system control information...	Tbc	

Subordinates	1.1dfa, 1.3dfa
Superordinates	Tbc

USE CASE # 2.4.1cfa	**Platoon break-up - signalled by lead vehicle**	
Goal	Convoy breaks up with each vehicle operating individually	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	The platoon successfully breaks up.	
Failed End Condition	(i) () platoon did not break-up when request transmitted by the lead vehicle.	
Actors	Lead_vehicle, nth_vehicle, Zonal_drivers, Infrastructure	
ID Type	ID necessary, location information required	
Data priority	High priority	
Driving Environment	Motorway Initially	
Trigger	Lead vehicle senses communication failure	
	Step	**Action**
	1	Lead_vehicle detects communication failure
	2	Lead vehicle broadcasts message to communications infrastructure
	3	Simultaneously a message is transmitted to the platoon to change into self-automated mode.
	4	Infrastructure transmits the break-up of platoon to the platoons within the RZR of the old platoon.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Platoon break-lead vehicle signals break-up (centralised control)	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
Channels to actors	<e.g. interactive, static files, database, timeouts>	
OPEN ISSUES	Infrastructure could communicate to each vehicle in the platoon to transfer into self automated mode. This would ensure that each vehicle received the instruction simultaneously.	
...any other system control information...	Tbc	
Superordinates	Tbc	
Subordinates	Tbc	

USE CASE 2.4.1dfa	**Platoon break-up - signalled by lead vehicle**
Goal	Convoy breaks up with each vehicle operating individually
Control Structure	Decentralised, fully automated
Scope & Level	Secondary case
Successful End Condition	The platoon successfully breaks up.
Failed End Condition	(i) () platoon did not break-up when request transmitted by the lead vehicle.
Actors	Lead_vehicle, nth_vehicle, Zonal_drivers

ID Type	ID necessary, location information required	
Data priority	High priority	
Driving Environment	Motorway	
Trigger	Lead vehicle senses communication failure	
	Step	**Action**
	1	Lead_vehicle detects communication failure
	2	Lead vehicle broadcasts message to communications infrastructure
	3	simultaneously a message is transmitted to the platoon to change into self-automated mode.
	4	infrastructure transmits the break-up of platoon to the platoons within the RZR of the old platoon.
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Platoon break-lead vehicle signals break-up (centralised control)	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
Channels to actors	<e.g. interactive, static files, database, timeouts>	
OPEN ISSUES	Infrastructure could communicate to each vehicle in the platoon to transfer into self automated mode. This would ensure that each vehicle received the instruction simultaneously.	
...any other system control information...	Tbc	
Superordinates	Tbc	
Subordinates	Tbc	

USE CASE 2.4.2cfa	**Platoon break-up - signalled by infrastructure**	
Goal	platoon breaks up	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	The platoon successfully breaks up.	
Failed End Condition	(i) () platoon did not break-up when request transmitted by the infrastructure.	
Actors	Lead_vehicle, nth_vehicle, Zonal_drivers	
Control	Centralised/ Decentralised	
ID Type	ID necessary, location information required	
Data priority	High priority	
Driving Environment	Motorway initially	
Trigger	obstruction, emergency, coordination problem, platoon exciting onto motorway .	
	Step	**Action**
	1	Infrastructure transmits message to the lead vehicle requesting the platoon to break-up
	2	lead vehicle acknowledges message
	3	lead vehicle transmits message to the platoon n times
	4	nth vehicle transmits message back to the lead vehicle
	5	Lead vehicle transmits message back to the infrastructure - all_platoon_members_ ready
	6	Infrastructure transmits command to the lead vehicle
	7	lead vehicle transmits command to platoon

	8	vehicles transfer to self-automated mode
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Platoon break-up - infrastructure signals platoon break-up	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	For the decentralised case, assuming minimal infrastructure exists for emergency supervisory issues, the same three steps apply.	
...any other system control information...	Tbc	
Superordinates	Tbc	
Subordinates	Tbc	

. **USE CASE # 2.4.3cfa**	**Platoon break-up - signalled by another platoon**	
Goal	Platoon breaks up	
Control Structure	Centralised, fully automated	
Scope & Level	Secondary case	
Successful End Condition	The platoon successfully breaks up.	
Failed End Condition	(i) () platoon did not break-up when request transmitted by another platoon.	
Actors	Lead_vehicle, nth_vehicle, Zonal_drivers, Infrastructure	
ID Type	ID necessary, location information required	
Data priority	High priority	
Driving Environment	Motorway Initially	
Trigger	obstruction, emergency, coordination problem (platoon exciting onto motorway), infrastructure unable to communicate directly with a specific platoon.	
	Step	**Action**
	1	Infrastructure broadcasts a platoon break-up message
	2	No acknowledgement message is received within the accepted time-frame
	3	Infrastructure broadcasts request vehicle/platoon to transmit the platoon break-up control information
	4	Platoon or vehicle that is within communication range of the target platoon will broadcast the message to the platoon.
	5	If the message is received by a member of the platoon, it will be forwarded to the lead vehicle
	6	Either the lead vehicle, platoon member or transmitting platoon/vehicle will transmit an acknowledgement to the infrastructure.
	7	lead vehicle will transmit message to the platoon
	8	Platoon members transfer into self-automated mode
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
RELATED INFORMATION	Platoon break-up - signalled by outside platoon/vehicle	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	

OPEN ISSUES	As the message from the infrastructure contains emergency information an acknowledgement is compulsory and the timeout window to receive the acknowledgement will be as short as possible.
...any other system control information...	Tbc
Superordinates	Tbc
Subordinates	Tbc

USE CASE 2.4.3dfa	Platoon break-up - signalled by another platoon	
Goal	Platoon breaks up	
Control Structure	Decentralised, fully automated	
Scope & Level	Secondary case	
Preconditions	<what we expect is already the state of the world>	
Success End Condition	The platoon successfully breaks up.	
Failed End Condition	(i) () platoon did not break-up when request transmitted by another platoon.	
Primary, Secondary Actors	Lead_vehicle, nth_vehicle, Zonal_drivers	
Control	Centralised/ Decentralised	
ID Type	ID necessary, location information required	
Data priority	High priority	
Driving Environment	Motorway initially	
Trigger	obstruction, emergency, infrastructure unable to communicate directly with a specific platoon, platoon functioning problems.	
	Step	**Action**
	1	Vehicle broadcasts emergency information
	2	Message received by a platoon member
	3	Message forwarded onto lead vehicle
	4	Lead_vehicle processes message and decides on action
	5	Lead_vehicle transmits control information to the platoon
	6	Vehicles transfer into self-automated mode
EXTENSIONS	**Step**	**Branching Action**
SUB-VARIATIONS		**Branching Action**
	2a	message received by lead vehicle
RELATED INFORMATION	Platoon break-up - signalled by outside platoon/vehicle	
Priority:	High	
Process length	Tbc	
Frequency	Tbc	
OPEN ISSUES	As the message from the infrastructure contains emergency information an acknowledgement is compulsory and the timeout window to receive the acknowledgement will be as short as possible.	
...any other system control information...	Tbc	
Superordinates	Tbc	
Subordinates	Tbc	

APPENDIX B

DERIVATION OF MEAN AND STANDARD DEVIATION FOR THEORETICAL HALF-GAUSSIAN MAC DELAY DISTRIBUTION

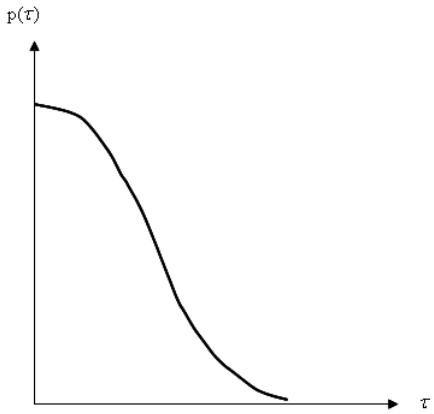


Figure B.1: Half-Gaussian MAC delay distribution, $p(\tau) = \frac{2}{\sigma\sqrt{2\pi}} e^{-\frac{\tau^2}{2\sigma^2}}$

By definition, the mean delay is given by,

$$\bar{\tau} = \langle \tau \rangle = \int_0^\infty \tau \cdot P(\tau) d\tau \tag{B.1}$$

$$\bar{\tau} = \int_0^\infty \frac{1}{\sigma} \sqrt{\frac{2}{\pi}} \tau e^{-\frac{\tau^2}{2\sigma^2}} d\tau \tag{B.2}$$

Changing the variable of integration to $u = \frac{\tau^2}{2\sigma^2}$, yields, $du = \frac{2\tau d\tau}{2\sigma^2}$, which simplifies to, $\sigma du = \frac{1}{\sigma}\tau d\tau$. Thus,

$$\bar{\tau} = \sqrt{\frac{2}{\pi}} \int_0^\infty \sigma e^{-u} du \tag{B.3}$$

$$\bar{\tau} = \sigma\sqrt{\frac{2}{\pi}}\left[-e^{-u}\right]_0^{\infty} = \sigma\sqrt{\frac{2}{\pi}}[0+1] \tag{B.4}$$

$$\bar{\tau} = \sigma\sqrt{\frac{2}{\pi}} \approx 0.798\sigma \tag{B.5}$$

Similarly, the second moment of the distribution is given by,

$$\langle\tau^2\rangle = \int_0^{\infty} \tau^2 \cdot p(\tau) d\tau \tag{B.6}$$

$$\langle\tau^2\rangle = \int_0^{\infty} \frac{1}{\sigma}\sqrt{\frac{2}{\pi}}\tau^2 e^{-\frac{\tau^2}{2\sigma^2}} d\tau \tag{B.7}$$

Using integration by parts, $\int u\frac{dv}{dx}dx = uv - \int v\frac{du}{dx}dx$, where $x = \tau$, $u = \tau$, $\frac{dv}{dx} = \tau e^{-\frac{\tau^2}{2\sigma^2}}$ and hence $v = -\sigma 2e^{-\frac{\tau^2}{2\sigma^2}}$, gives,

$$\langle\tau^2\rangle = \left[-\sigma^2\tau e^{-\frac{\tau^2}{2\sigma^2}}\frac{1}{\sigma}\sqrt{\frac{2}{\pi}}\right]_0^{\infty} + \int_0^{\infty}\frac{1}{\sigma}\sqrt{\frac{2}{\pi}}\sigma^2 e^{-\frac{\tau^2}{2\sigma^2}} d\tau \tag{B.8}$$

$$\langle\tau^2\rangle = 0 + \sigma\sqrt{\frac{2}{\pi}}\frac{1}{2}\sqrt{\frac{\pi}{\frac{1}{2\sigma^2}}} \tag{B.9}$$

This finally simplifies to,

$$\langle\tau^2\rangle = \sigma^2 \tag{B.10}$$

Thus the variance of τ can be determined through $var(\tau) = \langle\tau^2\rangle - \langle\tau\rangle^2$, to give,

$$var(\tau) = \sigma^2 - (\sigma\sqrt{\frac{2}{\pi}})^2 = \sigma^2\left(1 - \frac{2}{\pi}\right) \tag{B.11}$$

The standard deviation, $std(\tau)$ is finally given by,

$$std(\tau) = \sqrt{var(\tau)} = \sigma\sqrt{1 - \frac{2}{\pi}} \approx 0.6036 \tag{B.12}$$

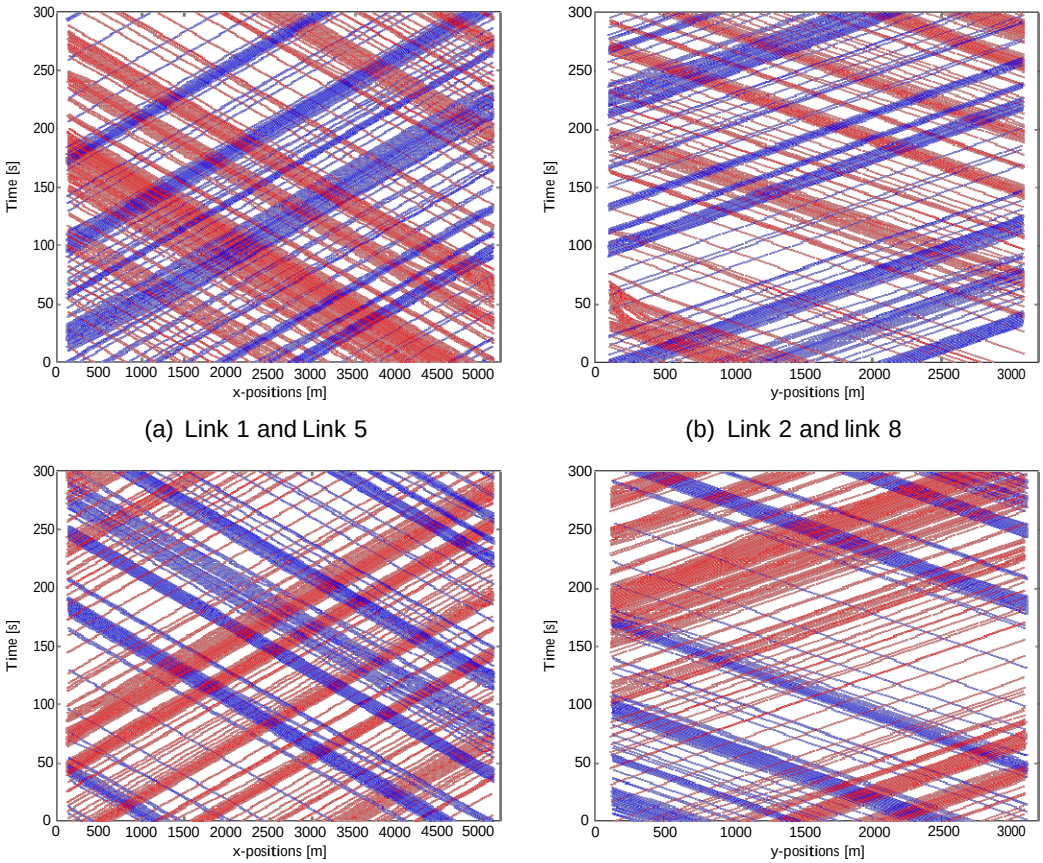
The ratio of mean channel access delay to its standard deviation for this one-sided Gaussian distribution is then a constant, independent of σ,

$$\frac{\bar{\tau}}{std(\tau)} \approx 1.324 \tag{B.13}$$

APPENDIX C

VEHICLE TRAJECTORY SPACE TIME PLOTS

C.1 Low Density



(a) Link 1 and Link 5

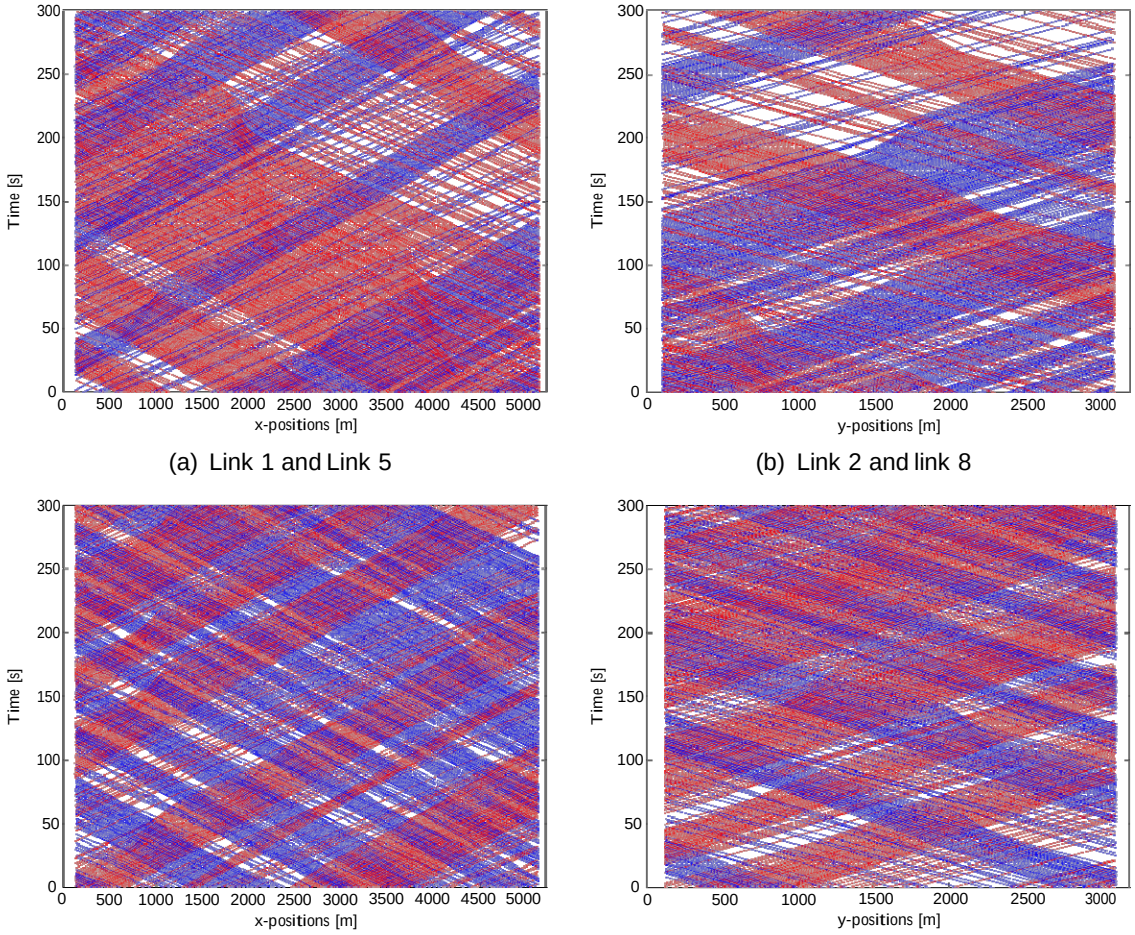
(b) Link 2 and link 8

(c) Link 3 and Link 7

(d) Link 6 and link 4

Figure C.1: Space-time plots of vehicle positions for low density traffic at a rate of 600 veh/lane/hr. Vehicles circulating in a clockwise direction are shown in blue, whilst vehicles circulating in an anti-clockwise direction as shown in red. Each plot shows the vehicle positions for both lanes on each carriageway link.

C.2 Low/Medium Density



(a) Link 1 and Link 5

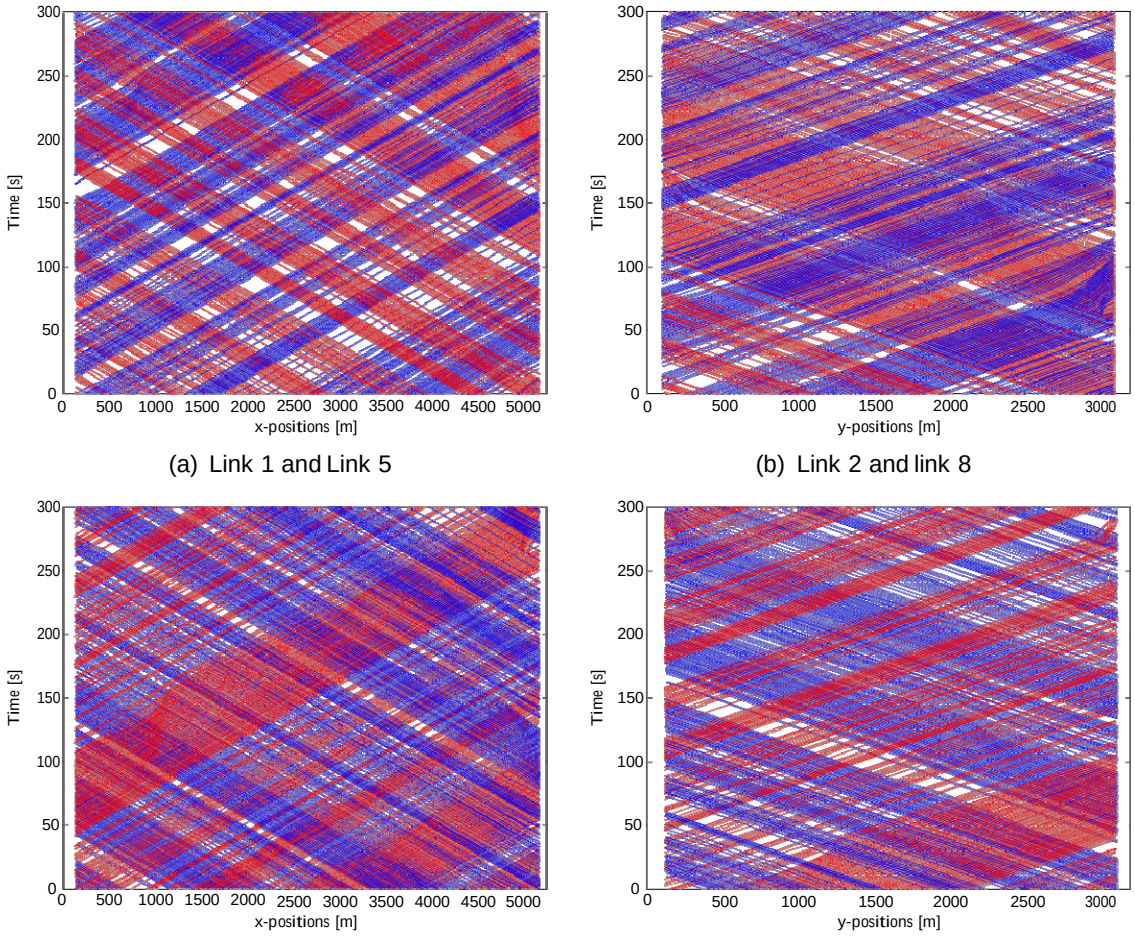
(b) Link 2 and link 8

(c) Link 3 and Link 7

(d) Link 6 and link 4

Figure C.2: Space and time plot of vehicle positions for low-medium density at a rate of 800 veh/lane/hr. Vehicles circulating in a clockwise direction are shown in blue, whilst vehicles circulating in an anticlockwise direction as shown in red. Each plot shows the vehicle positions for both lanes on each carriageway link

C.3 Medium Density



(a) Link 1 and Link 5

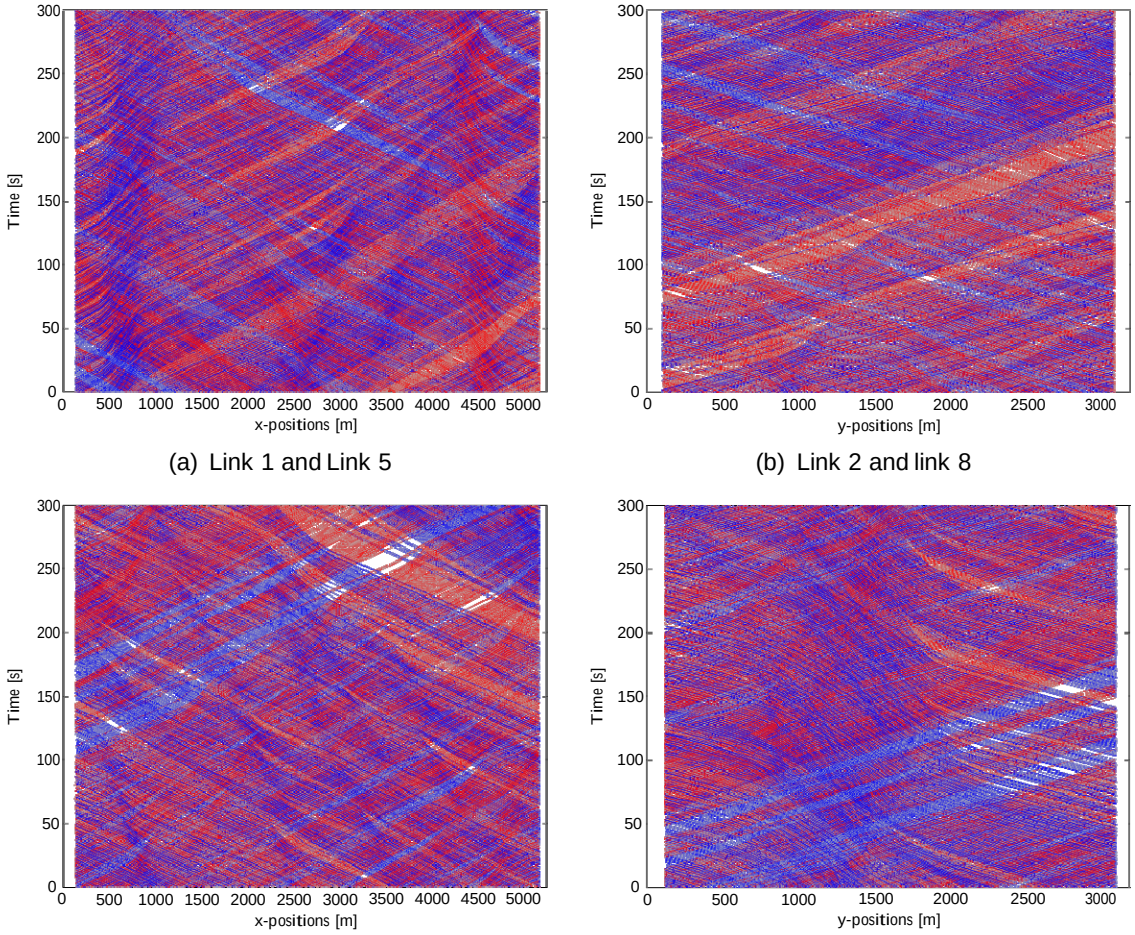
(b) Link 2 and link 8

(c) Link 3 and Link 7

(d) Link 6 and link 4

Figure C.3: Space and time plot of vehicle positions for medium density at a rate of 1100 veh/lane/hr. Vehicles circulating in a clockwise direction are shown in blue, whilst vehicles circulating in an anti-clockwise direction as shown in red. Each plot shows the vehicle positions for both lanes on each carriageway link.

C.4 Medium/High Density



(a) Link 1 and Link 5

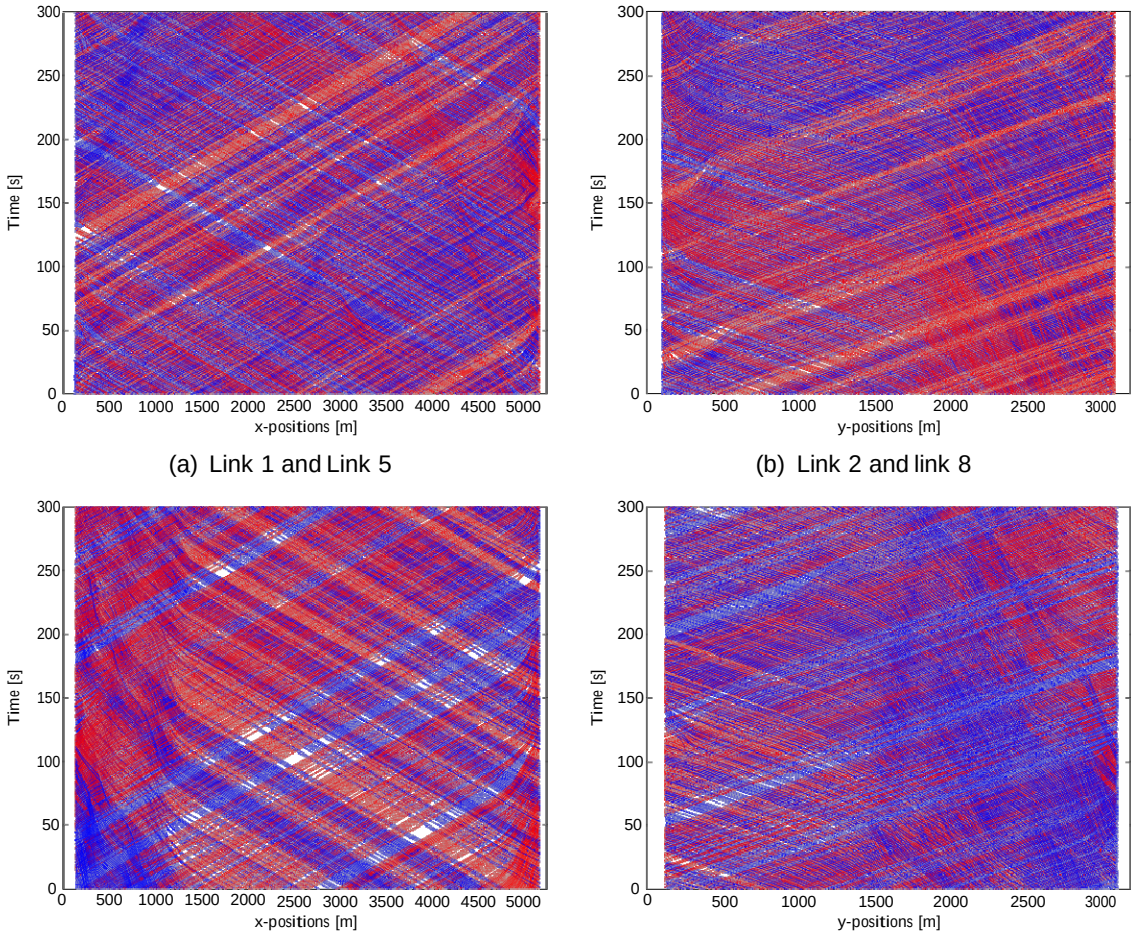
(b) Link 2 and link 8

(c) Link 3 and Link 7

(d) Link 6 and link 4

Figure C.4: Space-time plot of vehicle positions for medium/high density at a rate of 1500 veh/lane/hr. Vehicles circulating in a clockwise direction are shown in blue, whilst vehicles circulating in an anticlockwise direction as shown in red. Each plot shows the vehicle positions for both lanes on each carriageway link

C.5 High Density



(a) Link 1 and Link 5

(b) Link 2 and link 8

(c) Link 3 and Link 7

(d) Link 6 and link 4

Figure C.5: Space and time plot of vehicle positions for high density traffic at a rate of 1700 veh/lane/hr. Vehicles circulating in a clockwise direction are shown in blue, whilst vehicles circulating in an anti-clockwise direction as shown in red. Each plot shows the vehicle positions for both lanes on each carriageway link

APPENDIX D

RADIO TRANSCEIVER PIPELINE STAGES

Stage 0: Receiver Group

This stage is executed once only at the start of a simulation for each transmitter and receiver channels to determine the feasibility of communication. It is not executed on a per transmission basis. It assigns receiver groups for all transmitter channels. The default receiver group stage model where all receivers are in the receive group of all the transmitters is used in this research.

Stage 1: Transmission delay

This is the first stage of the radio pipeline to be invoked dynamically at the beginning of a packet transmission and calculates the amount of time required for the entire packet to complete transmission. The result is used to schedule an end-of-transmission event, during this time interval the transmitter stops transmitting new packets. The transmission delay is calculated using equation (D.1) using channel data rate and packet size. The result from the transmission delay is used in conjunction with the propagation delay stage to calculate the time at which the packet completes reception at the links destination. This model is executed once only for each packet transmitted and the resulting computation shared be all resulting receiver destinations. **The default transmission delay model is used.**

$$trx\_delay = \frac{pckt\_size}{data\_rate} \tag{D.1}$$

Stage 2: Link Closure

This stage determines if the transmission can physically reach a receiver channel or if occlusion occurs due to the curvature of the earth. If the transmitted packet can physically attain the candidate receiver channel then the packet continues transmission through the remaining pipeline stages. Otherwise, if the receiver cannot be reached, execution of the pipeline stage is stopped for this particular receiver and the process moves onto the next receiver in the receiver group determined during stage 0. This stage is invoked once for each receiver channel included in the transmitting channels destination channel set (i.e. transmitters receiver group set).

Stage 3: Channel Match

This stage is invoked once for each receiver channel that satisfies the criteria of stage 2. The purpose of this stage is to classify the transmission with respect to the receiver channel based on Frequency, bandwidth, data rate, modulation, etc. A packet is assigned as valid if the receiver channel is determined to be compatible with the transmission. When incompatibilities occur between the transmitter and receiver

channel configurations then the packet is classed as noise and may cause interference at the receiver. However, if a transmitter and receiver channel configurations do not overlap and the transmission cannot affect the receiver channel whatsoever, then the packet is ignored and further execution of the pipeline stage a the receiver for this transmission ceases.

Stage 4: Transmitter Antenna Gain

The purpose of the transmitter antenna gain stage is to compute the gain provided by the transmitters associated antenna. This is determined from the direction of the vector leading from the transmitter to the receiver. The simulation Kernel does not use the result directly, but it is typically factored into the received power calculation performed by stage 7. This stage is executed separately for each destination channel, except those that failed at stages 2 an 3. As discussed in §5.7.3 an isotropically radiating antenna is used in this work, and therefore the antenna gain is $0dBi$ in all directions uniformly.

Stage 5: Propagation delay

The purpose of this stage is to calculate the amount of time required for the packet to travel from the transmitter to the receiver. The result is dependant on the distance between the source of the transmission and the receiver. The kernel uses this result to schedule a beginning-of-reception event for the receiver channel that the packet is destined for. In addition, the propagation delay result is used in conjunction with the result of stage 1 to compute the time at which the packet completes reception. This stage was modified so that the status of the packet was changed to noise if the distance exceeded the required transmission range.

Stage 6: Receiver Antenna Gain

This is the first stage directly associated with the receiver. The purpose of this stage is to compute the gain provided by the receivers associated antenna, based on the direction of the vector leading from the receiver to the transmitter. Again as in stage 4, the kernel does not use this result, instead it is factored into the calculation received power during stage 7. The default model was used.

Stage 7: Received power

The purpose of this stage is to compute the received power of the arriving packet's signal. In general, the calculation of received power is based on factors such as the power of the transmitter, the path loss over the distance separating the transmitter an receiver and antenna gains. The receiver will lock onto the first arriving packet any packets received after this packet whilst the receiver is locked will be classed as Noise. The default model uses free space path propagation, the path loss equation was modified as described in §5.7.3.

Stage 8: Background Noise

The purpose of this stage is to represent the effect of all noise sources, with the exception of inter-packet interference (which is modelled by stage 9).The expected result is the sum of the power of other noise sources measured at the receivers location and in the receiver channel band. Tropical background noise sources include thermal or galactic noise, emissions from neighbouring electronics and otherwise unmodelled radio transmissions.

The background noise N_b is characterised by an effective background temperature T_{bk} which is added to the effective device temperature of the receiver T_{rx}. The receiver temperature, T_{rx}, is determined from the receiver noise figure NF assuming an operating temperature of 290 K. The sum of these temperatures, each accounting for a separate source of noise is multiplied by the bandwidth of the receiver B_{rx} and Boltzman's constant, k, to obtain the added noise contributed by the receiver to the processed signal. The

noise is added to the ambient noise N_a to model overall effect of inexplicitly modelled noise sources. The value of NF can be set ny the user and N_b is assigned a default value of 10^{-26}.

$$T_{rx} = (N_f - 1) \times 290.0 \tag{D.2}$$

$$N_b = (T_{rx} + T_{bk}) B_{rx} k \tag{D.3}$$

$$N_a = B_{rx}(1.0E^{-26}) \tag{D.4}$$

$$N = N_b + N_a \tag{D.5}$$

Stage 9: Interference noise

The computation of interference noise occurs when two packets are simultaneously present at the same radio receiver, which could occur under the following circumstances: the packet arrives at its destination channel while another packet is already being received; or the packet is already being received when another packet arrives. The first circumstance can occur at most once for each packet and the second may occur any number of times depending upon the transmission activities of other transmitters in the model.

For each packet arriving concurrently the process increments the number of collisions. The process determines if the packets are either noise or valid packets, and retrieves the received power level determined from previous pipeline stages. Computations of noise contribution are only necessary for affected valid packets, because invalid packets are not considered for reception and the their signal quality need not be determined. Each packet has a noise accumulator associated with it that is augmented with the received power of each interfering packet as shown in Figure D.1. When a current packet completes reception, the kernel automatically subtracts its received noise power from the noise accumulator of all valid packets that are still arriving at the channel. In this manner, the noise accumulator reflects only the current noise level.

$$P_i = sum_{t=t_{rcvd}}^{t=t_{comp}} P_{rcv} \tag{D.6}$$

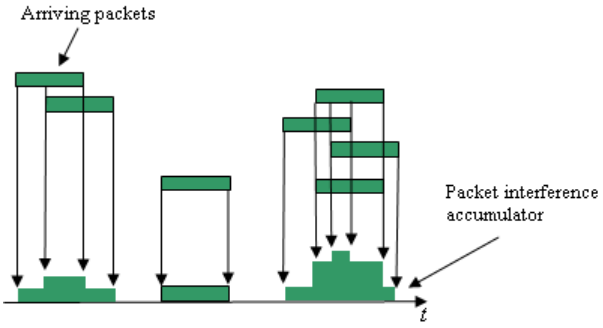


Figure D.1: Principle of accumulation of interference in the OPNET radio transceiver

Stage 10: Signal_to_noise Ratio (SNR)

The SNR stage is invoked for valid packets under three conditions only: the packet arrives at its destination channel, or the packet is being received and another valid or noise packet arrives or the packet is already being received and another valid or noise packet completes reception. The above mentioned invocations define SNR update intervals over which a packet's average power SNR is taken to be constant (which is an approximation when the nodes are mobile). Background noise is calculated once for each transmission (at the time reception starts), new interference sources can become active or inactive many times during a packets reception. Therefore, SNR may need to be reevaluated many times during the packet's reception.

The portion of the packet arriving at the receiver between SNR updates is termed a packet segment. During any given packet segment, the value of SNR is constant. The computation of SNR depends on the average background noise power P_b and the average interference noise power P_i from other transmissions, determined from stages 8 and 9 respectively. SNR is calculated using equation (D.7).

$$SNR = 10 \log_{10}[\frac{P_r}{(P_b + P_i)}] \tag{D.7}$$

Stage 11: Bit Error Rate (BER)

The BER of a packet is evaluated according to each segment of the packet defined by the SNR update intervals in stage 10, when the value of the SNR changes as described for the three invocation conditions described for stage 10. The purpose of the BER stage is to derive the probability of bit errors during the past interval of constant SNR. This is not the empirical rate of bit errors, but the expected rate, usually based on previously computed average power SNR. An effective SNR SNR_{eff} is determined from the addition of SNR to the processing gain of the receiver G_p. The BER is then derived using a modulation look-up table based on the value of SNR_{eff}.

Stage 12: Error Allocation

The error allocation stage estimates the number of bit errors in a packet segment where the bit error probability has been calculated and is constant. This segment may be the entire packet if no changes in bit error probability occur over the course of the packet's reception. Bit error count estimation is based on the bit error probability derived in stage 11 and the length of the affected segment. This stage maintains an error accumulator for each packet which holds the accumulation of the number of bit errors for each segment.

Stage 13: Error Correction

This stage is invoked one for each packet completing reception. The purpose of this stage is to determine whether or not the packet should be accepted at the receiver. The resulting value of accumulated errors determined in stage 12 is then compared with the error correction threshold of the receiver, if it less than this value and the packet has not been truncated due to errors at the transmitter than the packet is accepted.

APPENDIX E

COMPARISON OF IEEE 802.11p WITH IEEE802.11b SIMULATION SETTINGS

Parameter	802.11p	802.11b (simulated)
Frequency band	5.9 GHz	2.4 GHz
Channel bandwidth	10 MHz	20 MHz
Data Rate(s)	3 to 27 Mbps	11 Mbps
Slot time	16 μs	20 μs
SIFS time	32 μs	10 μs
Air propagation time	4 μs	1 μs
PLCP Preamble length	32 μs	192 μs
CW_{min}	15	31
CW_{min}	1023	1023
Receiver threshold	-95 dBm	-101 dBm
Transmit power(max)	33 dBm	0 dBm

BIBLIOGRAPHY

[1] U.S Department of Tansporation. Research and Innovative Technology Administration: Transportation Vision for 2030; 2008. http://www.rita.dot.gov/publications/transportation_vision_2030/pdf/entire.pdf. Last Accessed 31/10/11.

[2] Europe's Information Society Thematic Portal;. http://ec.europa.eu/information_society/activities/esafety/index_en.htm. Last accessed 4/11/11.

[3] COMeSafety Website. Communciations for eSafety;. http://www.comesafety.org/. Last accessed 20/10/11.

[4] U.S Deperatment of Transportation. Research and Innovotive Technology Adminstration (RITA). Connected Vehicle Research (Previously referred to as IntelliDrive program);. http://www.its.dot.gov/connected_vehicle/connected_vehicle.htm. Last Accessed: 4/11/11.

[5] Hartenstein H, Laberteaux KP. A tutorial survey on vehicular ad hoc networks. Communications Magazine, IEEE. 2008;46(6):164–171.

[6] Ward D, Topham D, Constantinou C, Arvanitis T. Developments in Vehicle-to-vehicle Communications. In: Valldorf J, Gessner W, editors. Advanced Microsystems for Automotive Applications 2005. VDI-Buch. Springer Berlin Heidelberg; 2005. p. 353–370.

[7] ETSI Technical Commitee on ITS;. http://www.etsi.org/WebSite/Technologies/DSRC.aspx: Last Accessed 19/09/2011.

[8] CEN TC 278. Developing Standards for ITS;. http://www.compumax.nl/tc278/. Last Accessed 3/11/11.

[9] Car-to-Car Communications Consortium (C2C-CC). CAR 2 CAR Communication Consortium Manifesto, V1.1; 2007. http://www.car-to-car.org. Last Accessed: 19/09/2011.

[10] Torrent-Moreno M. Inter-vehicle communications: Achieving safety in a distributed wireless environment, PhD Thesis. Inst. of Telematics, University Karlsruhe (TH), Karlsruhe, Germany; 2007.

[11] Torrent-Moreno M, Santi P, Hartenstein H. Distributed Fair Transmit Power Adjustment for Vehicular Ad Hoc Networks. In: Proc. 3rd Annual IEEE Communications Society Sensor and Ad Hoc Communications and Networks SECON '06. vol. 2; 2006. p. 479–488.

[12] Willke TL, Tientrakool P, Maxemchuk NF. A survey of inter-vehicle communication protocols and their applications. IEEE Communications Surveys & Tutorials. 2009;11(2):3–20.

[13] Molisch A, Tufvesson F, Karedal J, Mecklenbrauker C. A survey on vehicle-to-vehicle propagation channels. IEEE Wireless Communications Magazine. 2009;16(6):12–22.

[14] IEEE 802.11p. Amendment 6: Wireless Acces in Vehicular Environments (WAVE); 2010.

[15] Topham DA, Ward D, Arvanitis TN, Constantinou CC. Inter-vehicle communications based on mobile ad hoc networks, , 135-140, 2003. In: Proceedings of the 1st IEE International Conference on Sensors, Navigation and Communications for Vehicle Telematics (VehCom 2003); 2003.

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

[17] ETSI TC ITS, Intetelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions. ETSI TR 102 V1.1.1; June 2009.

[18] Car-to-Car Communications Consortium (C2C-CC);. http://www.car-to-car.org. Last Accessed: 19/09/2011.

[19] Vehicle Safety Communications Consortium, Vehicle Safety Communications Project. Task 3. Final Report: Identify Intelligent Vehicle Safety Applications Enabled by DSRC. DOT HS 809 859. US Department of Transport; March 2005.

[20] IEEE 802.11a. Amendment 1: High Speed Physical Layer in the 5 GHz Band; 1999.

[21] IEEE 802.11. IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications; 2007.

[22] Uzcategui R, Acosta-Marum G. WAVE: A tutorial. IEEE Communications Magazine. 2009;47(5):126–133.

[23] ETSI: Cooperative ITS. Website for Memorandom activties between ETSI andCEN;. http://www.etsi.org/WebSite/Technologies/CooperativeITS.aspx. Last accessed 21/10/11.

[24] CEN and ETSI. Joint CEN and ETSI Response to Mandate M/453; 2010. http://www.ertico.com/assets/News-images/EU-news/pdf/cenetsiresponsetoprogrammerelatedtomandatem453.pdf. Last accessed 21/10/11.

[25] ISO TC WG 16: CALM Website;. http://www.isotc204wg16.org/. Last acessed 20/10/11.

[26] IEEE 1609 WG. Official Website of the DSRC/WAVE WG;. http://standards.ieee.org/develop/wg/1609_WG.html. Last Accessed 19/09/2011.

[27] ETSI TR 102 638 V1.1.1. Intelligent Transportation Systems (ITS); European Profile Standard for the Physical and Medium Acces Control Layer of Intelligent Transportation Systems Operating in the 5 GHz Frequency Band; 2009-06.

[28] Kasemann M, Hartenstein H, Fubler H, Mauve M. Analysis of a Location Service for Position-Based Routing in Mobile Ad Hoc Networks. In: Proc. of the 1st German Workshop on Mobile Ad-hoc Networking (WMAN 2002), Ulm, Germany; 2002. .

[29] Perkins CE, Bhagwat P. Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers. In: Proceedings of the conference on Communications architectures, protocols and applications. SIGCOMM '94. New York, NY, USA: ACM; 1994. p. 234–244. Available from: http://doi.acm.org/10.1145/190314.190336.

[30] Royer EM, Toh CK. A review of current routing protocols for ad hoc mobile wireless networks. IEEE Personal Communications. 1999;6(2):46–55.

[31] Perkins CE. Ad Hoc Networking. Addison-Wesley Professional; 2000.

[32] Johnson DB, Maltz DA. Dynamic source routing in ad hoc wireless networks. In: Mobile Computing. Kluwer Academic Publishers; 1996. p. 153–181.

[33] Haas ZJ, Pearlman MR. The performance of query control schemes for the zone routing protocol. In: Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications. DIALM '99. New York, NY, USA: ACM; 1999. p. 23–29. Available from: http://doi.acm.org/10.1145/313239.313271.

[34] Pearlman MR, Haas ZJ. Determining the optimal configuration for the zone routing protocol. IEEE Journal on Selected Areas in Communications. 1999;17(8):1395–1414.

[35] Tian J, Stepanov I, Rothermel K. Spatial aware geographic Forwarding for mobile ad hoc networks. University of Stuttgart; 2002. Technical Report. Available from: http://elib.uni-stuttgart.de/opus/volltexte/2002/1160.

[36] Mauve M, Widmer A, Hartenstein H. A survey on position-based routing in mobile ad hoc networks. IEEE Network. 2001;15(6):30–39.

[37] Karp B, Kung HT. GPSR: greedy perimeter stateless routing for wireless networks. In: Proceedings of the 6th annual international conference on Mobile computing and networking. MobiCom '00. New York, NY, USA: ACM; 2000. p. 243–254. Available from: http://doi.acm.org/10.1145/345910.345953.

[38] Camp T, Boleng J, Davies V. A survey of mobility models for ad hoc network research. Wireless Communications and Mobile Computing. 2002;2(5):483–502. Available from: http://dx.doi.org/10.1002/wcm.72.

[39] Tian J, Hahner J, Becker C, Stepanov I, Rothermel K. Graph-based mobility model for mobile ad hoc network simulation. In: Proc. 35th Annual Simulation Symp; 2002. p. 337–344.

[40] Briesemeister L, Hommel G. Overcoming Fragmentation in Mobile Ad Hoc Networks. Journal of Communications and Networks. 2000;2:182–187.

[41] Kremer W. Vehicle density and communication load estimation in mobile radio local area networks (MR-LANs). In: Proc. IEEE 42nd Vehicular Technology Conf.; 1992. p. 698–704.

[42] Ko YB, Vaidya NH. Location-aided routing (LAR) in mobile ad hoc networks. Wirel Netw. 2000 July;6:307–321. Available from: http://dx.doi.org/10.1023/A:1019106118419.

[43] Haas ZJ, Tabrizi S. On some challenges and design choices in ad-hoc communications. In: Military Communications Conference, 1998. MILCOM 98. Proceedings., IEEE. vol. 1; 1998. p. 187 –192 vol.1.

[44] Stojmenovic I. Position-based routing in ad hoc networks. IEEE Communications Magazine. 2002;40(7):128–134.

[45] Briesemeister L, Schafers L, Hommel G. Disseminating messages among highly mobile hosts based on inter-vehicle communication. In: Proc. IEEE Intelligent Vehicles Symp. IV 2000; 2000. p. 522–527.

[46] Steenstrup M, editor. Routing in communications networks. Prentice Hall International (UK) Ltd.; 1995.

[47] Ni SY, Tseng YC, Chen YS, Sheu JP. The broadcast storm problem in a mobile ad hoc network. In: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking. MobiCom '99. New York, NY, USA: ACM; 1999. p. 151–162. Available from: http://doi.acm.org/10.1145/313451.313525.

[48] Tseng YC, Ni SY, Chen YS, Sheu JP. The broadcast storm problem in a mobile ad hoc network. WirelessNetworks, ACM. 2002 March;8:153–167. Available from: http://dx.doi.org/10.1023/A:1013763825347.

[49] Wisitpongphan N, Tonguz OK, Parikh JS, Mudalige P, Bai F, Sadekar V. Broadcast storm mitigation techniques in vehicular ad hoc networks. IEEE Wireless Communications Magazine. 2007;14(6):84–94.

[50] Sun MT, Feng WC, Lai TH, Yamada K, Okada H, Fujimura K. GPS-based message broadcast for adaptive inter-vehicle communications. In: Proc. 52nd Vehicular Technology Conf. IEEE VTS-Fall VTC 2000. vol. 6; 2000. p. 2685–2692.

[51] Briesemeister L, Hommel G. Role-based multicast in highly mobile but sparsely connected ad hoc networks. In: Proc. MobiHOC Mobile and Ad Hoc Networking and Computing 2000 First Annual Workshop; 2000. p. 45–50.

[52] Ko YB, Vaidya NH. Geocasting in mobile ad hoc networks: location-based multicast algorithms. In: Proc. Second IEEE Workshop Mobile Computing Systems and Applications WMCSA '99; 1999. p. 101–110.

[53] Briesemeister L. Group Membership and Communication in Highly Mobile Ad Hoc Networks [PhD Thesis]. Technical University of Berlin; 2001.

[54] Bachir A, Benslimane A. A multicast protocol in ad hoc networks inter-vehicle geocast. In: Proc. VTC 2003-Spring Vehicular Technology Conf. The 57th IEEE Semiannual. vol. 4; 2003. p. 2456–2460.

[55] Benslimane A. Optimized Dissemination of Alarm Messages in Vehicular Ad-Hoc Networks (VANET). In: Mammeri Z, Lorenz P, editors. High Speed Networks and Multimedia Communications. vol. 3079 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg; 2004. p. 655–666.

[56] Joshi HP, Sichitiu ML, Kihl M. Distribured Robust Geocast. Multicast Routing for Inter-Vehicle Communication. In: Proc. First WEIRD Workshop on WiMAx, Wireless and Mobility, 2007, pp.9-21; 2007. p. 9–21.

[57] Tonguz O, Wisitpongphan N, Bai F, Mudalige P, Sadekar V. Broadcasting in VANET. In: Proc. Mobile Networking for Vehicular Environments; 2007. p. 7–12.

[58] Tonguz OK, Wisitpongphan N, Bai F. DV-CAST: A distributed vehicular broadcast protocol for vehicular ad hoc networks. Wireless Communications, IEEE. 2010 april;17(2):47 –57.

[59] Kim KW, Kim KK, Han CM, Lee MMO, Kim YK. An Enhanced Broadcasting Algorithm in Wireless Ad-hoc Networks. In: Information Science and Security, 2008. ICISS. International Conference on; 2008. p. 159 –163.

[60] Korkmaz G, Ekici E, Özgüner F, Özgüner U. Urban multi-hop broadcast protocol for inter-vehicle communication systems. In: Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks. VANET '04. New York, NY, USA: ACM; 2004. p. 76–85. Available from: `http://doi.acm.org/10.1145/1023875.1023887`.

[61] Korkmaz G, Ekici E, Ozguner F. Black-Burst-Based Multihop Broadcast Protocols for Vehicular Networks. Vehicular Technology, IEEE Transactions on. 2007 sept;56(5):3159 –3167.

[62] Fasolo E, Zanella A, Zorzi M. An Effective Broadcast Scheme for Alert Message Propagation in Vehicular Ad hoc Networks. In: Communications, 2006. ICC '06. IEEE International Conference on. vol. 9; 2006. p. 3960 –3965.

[63] Palazzi CE, Ferretti S, Roccetti M, Pau G, Gerla M. How Do You Quickly Choreograph Inter-Vehicular Communications? A Fast Vehicle-to-Vehicle Multi-Hop Broadcast Algorithm, Explained. In: Proc. 4th IEEE Consumer Communications and Networking Conf. CCNC 2007; 2007. p. 960–964.

[64] Campelli L, Cesana M, Fracchia R. Directional broadcast forwarding of alarm messages in VANETs. In: Wireless on Demand Network Systems and Services, 2007. WONS '07. Fourth Annual Conference on; 2007. p. 72 –79.

[65] Torrent-Moreno M, Mittag J, Santi P, Hartenstein H. Vehicle-to-Vehicle Communication: Fair Transmit Power Control for Safety-Critical Information. Vehicular Technology, IEEE Transactions on. 2009 sept;58(7):3684 –3703.

[66] Torrent-Moreno M. Inter-vehicle communications: assessing information dissemination under safety constraints. In: Proc. Fourth Annual Conf. Wireless Demand Network Systems and Services WONS '07; 2007. p. 59–64.

[67] Schmidt-Eisenlohr F, Torrent-Moreno M, Mittag J, Hartenstein H. Simulation platform for inter-vehicle communications and analysis of periodic information exchange. In: Proc. Fourth Annual Conf. Wireless Demand Network Systems and Services WONS '07; 2007. p. 50–58.

[68] Osafune T, Lin L, Lenardi M. Multi-Hop Vehicular Broadcast (MHVB). In: Proc. Conf. 6th Int ITS Telecommunications; 2006. p. 757–760.

[69] Mariyasagayam MN, Osafune T, Lenardi M. Enhanced Multi-Hop Vehicular Broadcast (MHVB) for Active Safety Applications. In: Proc. 7th Int. Conf. ITS Telecommunications ITST '07; 2007. p. 1–6.

[70] Ibrahim K, Weigle MC, Abuelela M. p-IVG: Probabilistic Inter-Vehicle Geocast for Dense Vehicular Networks. In: Proc. IEEE 69th Vehicular Technology Conf. VTC Spring 2009; 2009. p. 1–5.

[71] Chiasserini CF, Gaeta R, Garetto M, Gribaudo M, Sereno M. Efficient broadcasting of safety messages in multihop vehicular networks. In: Proc. 20th Int. Parallel and Distributed Processing Symp. IPDPS 2006; 2006. .

[72] Alshaer H, Horlait E. An optimized adaptive broadcast scheme for inter-vehicle communication. In: Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st. vol. 5; 2005. p. 2840 – 2844 Vol. 5.

[73] Mylonas Y, Lestas M, Pitsillides A. Speed adaptive probabilistic flooding in cooperative emergency warning. In: Proceedings of the 4th Annual International Conference on Wireless Internet. WICON '08. ICST, Brussels, Belgium.; 2008. p. 81:1–81:7. Available from: `http://dl.acm.org/citation.cfm?id=1554126.1554228`.

[74] Jiang H, Guo H, Chen L. Reliable and Efficient Alarm Message Routing in VANET. In: Proc. 28th Int. Conf. Distributed Computing Systems Workshops ICDCS '08; 2008. p. 186–191.

[75] Durresi M, Durresi A, Barolli L, Hsu F. Intervehicle communication protocol for emergency situations. In: Parallel Architectures,Algorithms and Networks, 2005. ISPAN 2005. Proceedings. 8th International Symposium on; 2005. p. 6 pp.

[76] Chen W, Cai S. Ad hoc peer-to-peer network architecture for vehicle safety communications. Communications Magazine, IEEE. 2005;43(4):100–107.

[77] Chisalita L, Shahmehri N. A peer-to-peer approach to vehicular communication for the support of traffic safety applications. In: Proc. IEEE 5th Int Intelligent Transportation Systems Conf; 2002. p. 336–341.

[78] Blum J, Eskandarian A, Hoffman L. Mobility management in IVC networks. In: Proc. IEEE Intelligent Vehicles Symp; 2003. p. 150–155.

[79] Little TDC, Agarwal A. An information propagation scheme for VANETs. In: Proc. IEEE Intelligent Transportation Systems; 2005. p. 155–160.

[80] Torrent-Moreno M, Jiang D, Hartenstein H. Broadcast reception rates and effects of priority access in 802.11-based vehicular ad-hoc networks. In: Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks. VANET '04. New York, NY, USA: ACM; 2004. p. 10–18. Available from: `http://doi.acm.org/10.1145/1023875.1023878`.

[81] Füßler H, Mauve M, Hartenstein H, Ksemann M, Vollmer D. A Comparison of Routing Strategies for Vehicular Ad-Hoc Networks, TR-02-003. Department of Computer Science, University of Mannheim; 2002.

[82] Reumerman H, Ruffini M. Distributed Power Control for Reliable Broadcast in Inter-Vehicle Communication Systems. In: Proc. of the 2nd International Worshop on Intelligent Transportation (WIT 2005), March 2005; 2005. p. 153–157. Available from: http://www.prevent-ip.org/download/Events/20050315-16_WIT05/WIT05%20Willwarn.pdf.

[83] Artimy MM, Robertson W, Phillips WJ. Assignment of dynamic transmission range based on estimation of vehicle density. In: Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks. VANET '05. New York, NY, USA: ACM; 2005. p. 40–48. Available from: http://doi.acm.org/10.1145/1080754.1080761.

[84] Artimy M. Local Density Estimation and Dynamic Transmission-Range Assignment in Vehicular Ad Hoc Networks. Intelligent Transportation Systems, IEEE Transactions on. 2007 sept;8(3):400 –412.

[85] Yang L, Guo J, Wu Y. Channel Adaptive One Hop Broadcasting for VANETs. In: Proc. 11th Int. IEEE Conf. Intelligent Transportation Systems ITSC 2008; 2008. p. 369–374.

[86] Mittag J, Schmidt-Eisenlohr F, Killat M, Härri J, Hartenstein H. Analysis and design of effective and low-overhead transmission power control for VANETs. In: Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking. VANET '08. New York, NY, USA: ACM; 2008. p. 39–48. Available from: http://doi.acm.org/10.1145/1410043.1410051.

[87] Bouassida MS, Shawky M. A Cooperative Congestion Control Approach within VANETs: Formal Verification and Performance Evaluation. EURASIP J Wireless Comm and Networking. 2010;.

[88] Fallah YP, Huang C, Sengupta R, Krishnan H. Congestion Control Based on Channel Occupancy in Vehicular Broadcast Networks. In: Proc. IEEE 72nd Vehicular Technology Conf. Fall (VTC 2010-Fall); 2010. p. 1–5.

[89] He J, Chen HH, Chen TM, Cheng W. Adaptive congestion control for DSRC vehicle networks. Communications Letters, IEEE. 2010 February;14(2):127 –129.

[90] Mughal BM, Wagan AA, Hasbullah H. Efficient congestion control in VANET for safety messaging. In: Proc. Int Information Technology (ITSim) Symp. vol. 2; 2010. p. 654–659.

[91] Wischhof L, Rohling H. Congestion control in vehicular ad hoc networks. In: Vehicular Electronics and Safety, 2005. IEEE International Conference on; 2005. p. 58 – 63.

[92] Zang Y, Stibor L, Cheng X, Reumerman HJ, Paruzel A, Barroso A. Congestion Control in Wireless Networks for Vehicular Safety Applications. In: Proceedings of The 8th European Wireless Conference. Paris, France; 2007. p. 7. Available from: http://www.comnets.rwth-aachen.de.

[93] Zhang W, Festag A, Baldessari R, Le L. Congestion control for safety messages in VANETs: Concepts and framework. In: Proc. 8th Int. Conf. ITS Telecommunications ITST 2008; 2008. p. 199–203.

[94] Zhang Z. Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges. IEEE Communications Surveys & Tutorials. 2006;8(1):24–37.

[95] Chen ZD, Kung HT, Vlah D. Ad hoc relay wireless networks over moving vehicles on highways. In: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing. MobiHoc '01. New York, NY, USA: ACM; 2001. p. 247–250. Available from: http://doi.acm.org/10.1145/501449.501451.

[96] Maihofer C, Eberhardt R. Geocast in vehicular environments: caching and transmission range control for improved efficiency. In: Proc. IEEE Intelligent Vehicles Symp; 2004. p. 951–956.

[97] Topham DA, Ward D, Arvanitis TN, Constantinou CC. Routing Framework for Vehicular Ad Hoc Networks: Regional Dissemination of Data Using a Directional Restricted Broadcasting Technique. In: Proceedings of 2nd International Workshop on Intelligent Transportation (WIT05), pp.23-28; 2005. .

[98] Papadimitratos P, Buttyan L, Holczer T, Schoch E, Freudiger J, Raya M, et al. Secure vehicular communication systems: design and architecture. Communications Magazine, IEEE. 2008 November;46(11):100 –109.

[99] Harsch C, Festag A, Papadimitratos P. Secure Position-Based Routing for VANETs. In: Vehicular Technology Conference. VTC-2007 Fall. IEEE 66th; 2007. p. 26 –30.

[100] SeVeCom Project;. http://www.test.org/doe/. Last Accessed: 19/09/2011.

[101] eSafety Support. eSafety Working Groups: eSecurity;. http://www.esafetysupport.org/en/esafety_activities/esafety_working_groups/esecurity.htm. Last Accessed 7/11/11.

[102] Festag A, Papadimitratos P, Tielert T. Design and Performance of Secure Geocast for Vehicular Communication. IEEE Transactions on Vehicular Technology. 2010 June;59(5):2456–2471. Available from: http://www.ee.kth.se/~papadim/publications/fulltext/secure-geocast-position-based-routing-vanet.pdf.

[103] Schoch E, Kargl F. On the efficiency of secure beaconing in VANETs. In: Proceedings of the third ACM conference on Wireless network security. WiSec '10. New York, NY, USA: ACM; 2010. p. 111–116. Available from: http://doi.acm.org/10.1145/1741866.1741885.

[104] Raya M, Papadimitratos P, Hubaux JP. Securing Vehicular Communications. Wireless Communications, IEEE. 2006;13(5):8–15.

[105] Calandriello G, Papadimitratos P, Hubaux JP, Lioy A. On the Performance of Secure Vehicular Communication Systems. IEEE Transactions on Dependable and Secure Computing. 2011;8:898–912.

[106] Parno B, Perrig A. Challenges in Securing Vehicular Networks. In: Fourth Workshop on Hot Topics in Networks (HotNets-IV); 2005. Available from: http://conferences.sigcomm.org/hotnets/2005/papers/parno.pdf.

[107] Maihofer C, Eberhardt R. Geocast in vehicular environments: caching and transmission range control for improved efficiency. In: Intelligent Vehicles Symposium, 2004 IEEE; 2004. p. 951 – 956.

[108] Floyd S, Jacobson V. The synchronization of periodic routing messages. Networking, IEEE/ACM Transactions on. 1994 apr;2(2):122 –136.

[109] FleetNet - Internet On the Road;. http://www.neclab.eu/Projects/fleetnet.htm. Last Accessed 10/11/11.

[110] PATH Project;. http://www.path.berkeley.edu/. Last Accessed 10/11/11.

[111] SAFESPOT Integrated Project;. http://www.safespot-eu.org/. Last Accessed 10/11/11.

[112] PReVENT Project;. http://www.prevent-ip.org/.

[113] Perrone LF, Yuan Y, Nicol DM. Simulation of large scale networks II: modeling and simulation best practices for wireless ad hoc networks. In: WSC '03: Proceedings of the 35th conference on Winter simulation. Winter Simulation Conference; 2003. p. 685–693.

[114] Jardosh A, Belding-Royer EM, Almeroth KC, Suri S. Towards realistic mobility models for mobile ad hoc networks. In: MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking. New York, NY, USA: ACM; 2003. p. 217–229.

[115] Balci O. Validation, verification, and testing techniques throughout the life cycle of a simulation study. In: WSC '94: Proceedings of the 26th conference on Winter simulation. San Diego, CA, USA: Society for Computer Simulation International; 1994. p. 215–220.

[116] Pawlikowski K, Jeong HDJ, Lee JSR. On credibility of simulation studies of telecommunication networks. Communications Magazine, IEEE. Jan 2002;40(1):132–139.

[117] Kurkowski S, Camp T, Colagrosso M. MANET simulation studies: the incredibles. SIGMOBILE Mob Comput Commun Rev. 2005;9(4):50–61.

[118] Lighthill MJ, Whitham GB. On Kinematic Waves. II. A Theory of Traffic Flow on Long Crowded Roads. Proceedings of the Royal Society of London Series A Mathematical and Physical Sciences. 1955;229(1178):317–345. Available from: http://rspa.royalsocietypublishing.org/content/229/1178/317.abstract.

[119] Hoogendoorn SP, Bovy PHL. State-of-the-art of vehicular traffic flow modelling. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering. 2001;215(4):283–303. Available from: http://pii.sagepub.com/content/215/4/283.abstract.

[120] Highways Agency, UK. Validation of Micro-simulation Models: Software Review; 2006. http://www.highways.gov.uk/knowledge_compendium/assets/documents/Portfolio/Software%20Review%20-%20901.pdf. Last accessed 9/11/11.

[121] Corridor Simulation: CORSIM. U.S Department of Transporation: Federal Highways Administration (FHWA).; Website accessed January 2008. http://ops.fhwa.dot.gov/trafficanalysistools/corsim.htm.

[122] Paramics online: Traffic microsimulation software. Quadstone Paramics;. http://www.paramics-online.com/. Last accessed 10/11/11.

[123] VISSIM. PTV Traffic Mobility Logistics;. http://www.vissim.de/. Last Accessed 09/10/11.

[124] Transportation Research Group (TRG), Southampton University;. http://www.trg.soton.ac.uk/index.htm. Last Accessed: 08/11/11.

[125] Wu J, Brackstone M, McDonald M. Fuzzy sets and systems for a motorway microscopic simulation model. Fuzzy Sets and Systems. 2000;116(1):65–76.

[126] Wu J, McDonald M, Brackstone M. Effects of convoy driving on motorway flow stability and capacity. Road Transport Information and Control, 2000 Tenth International Conference on (Conf Publ No 472). 2000;p. 91–95.

[127] OPNET Technologies: Network R & D. OPNET Modeler;. http://www.opnet.com/solutions/network_rd/modeler.html. Last Accessed 10/11/11.

[128] NS-2 Main Page;. http://nsnam.isi.edu/nsnam/index.php/Main_Page. Last Accessed 10/11/11.

[129] GloMoSim: Global Mobile Information Systems Simulation Library;. http://pcl.cs.ucla.edu/projects/glomosim/. Last Accessed 10/11/11.

[130] Scalable Network Technologies: QualNet Simulator;. http://www.qualnet.com/. Last Accessed: 10/11/11.

[131] JiST/SWANS: Java in Simulation Time/ Scalable Wireless Ad hoc Network Simulator;. http://jist.ece.cornell.edu/. Last Accessed: 10/11/11.

[132] OMNeT++ Discrete Event Simulation System;. http://www.omnetpp.org/. Last Accessed: 10/11/11.

[133] CSIM;. http://www.csim.com. Last Accessed: 10/11/11.

[134] OPNET Technologies: Wireless Module User Guide for OPNET Modeler. Radio Transceiver Pipeline Manual;. opnet.com.

[135] MathWorks. MATLAB Overview;. http://www.mathworks.co.uk/products/matlab/index.html. Last accessed: 10/11/11.

[136] IEEE Std 802.11b. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band; 1999.

[137] Torrent-Moreno M, Schmidt-Eisenlohr F, Fubler H, Hartenstein H. Effects of a realistic channel model on packet forwarding in vehicular ad hoc networks. In: Proc. IEEE Wireless Communications and Networking Conf. WCNC 2006. vol. 1; 2006. p. 385–391.

[138] Kleinrock L, Tobagi F. Packet Switching in Radio Channels: Part I–Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics. Communications, IEEE Transactions on. 1975 Dec;23(12):1400 – 1416.

[139] Tobagi F, Kleinrock L. Packet Switching in Radio Channels: Part II–The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution. Communications, IEEE Transactions on. 1975 Dec;23(12):1417 – 1433.

[140] Bianchi G. Performance analysis of the IEEE 802.11 distributed coordination function. Selected Areas in Communications, IEEE Journal on. 2000 mar;18(3):535 –547.

[141] P Chatzimisios VV, Boucouvlalas AC. Throughtput and Delay analysis of IEEE 802.11 Protocol. In: Proceedings of the 5th Int. Workshop on Networked Appliances; 2002. .

[142] Foh CH, Zukerman M. Performance Analysis of the IEEE 802.11 MAC Protocol. In: Proceedings of the EW 2002 Conference; 2002. .

[143] Carvalho MM, Garcia-Luna-Aceves JJ. Delay analysis of IEEE 802.11 in single-hop networks. In: Proc. 11th IEEE International Conference on Network Protocols; 2003. p. 146–155.

[144] Banchs A. Analysis of the Distribution of the Backoff Delay in 802.11 DCF: A Step Towards End-to-End Delay Guarantees in WLANs. In: Lecture Notes in Computer Science, Proc. of the International Conference on Information and Networking (ICOIN '05). Springer Berlin / Heidelberg;. .

[145] Tickoo O, Sikdar B. Modeling Queueing and Channel Access Delay in Unsaturated IEEE 802.11 Random Access MAC Based Wireless Networks. IEEE/ACM Transactions on Networking. 2008;16(4):878–891.

[146] Oliveira R, Bernardo L, Pinto P. Performance Analysis of the IEEE 802.11 Distributed Coordination Function with Unicast and Broadcast Traffic. In: Proc. IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications; 2006. p. 1–5.

[147] Jong-Mu Choi JS, Young-Bae Ko F. Numerical Analysis of IEEE 802.11 Broadcast Scheme in Multihop Wireless Ad Hoc Networks. Lecture Notes in Computer Science, Proc of the International Conference on Information and Networking (ICOIN '05). 2005;.

[148] Ma X, Chen X, Refai HH. Unsaturated Performance of IEEE 802.11 Broadcast Service in Vehicle-to-Vehicle Networks. In: Proc. VTC-2007 Fall Vehicular Technology Conf. 2007 IEEE 66th; 2007. p. 1957–1961.

[149] Feller W. An Introduction to Probability Theory and Its Applications, Vol. 2. Wiley; 1971.

[150] Sedletsky M. Position-Based Highway-Optimised Geocast. Computer Science Department, Bar Ilan University, Israel; 2009.