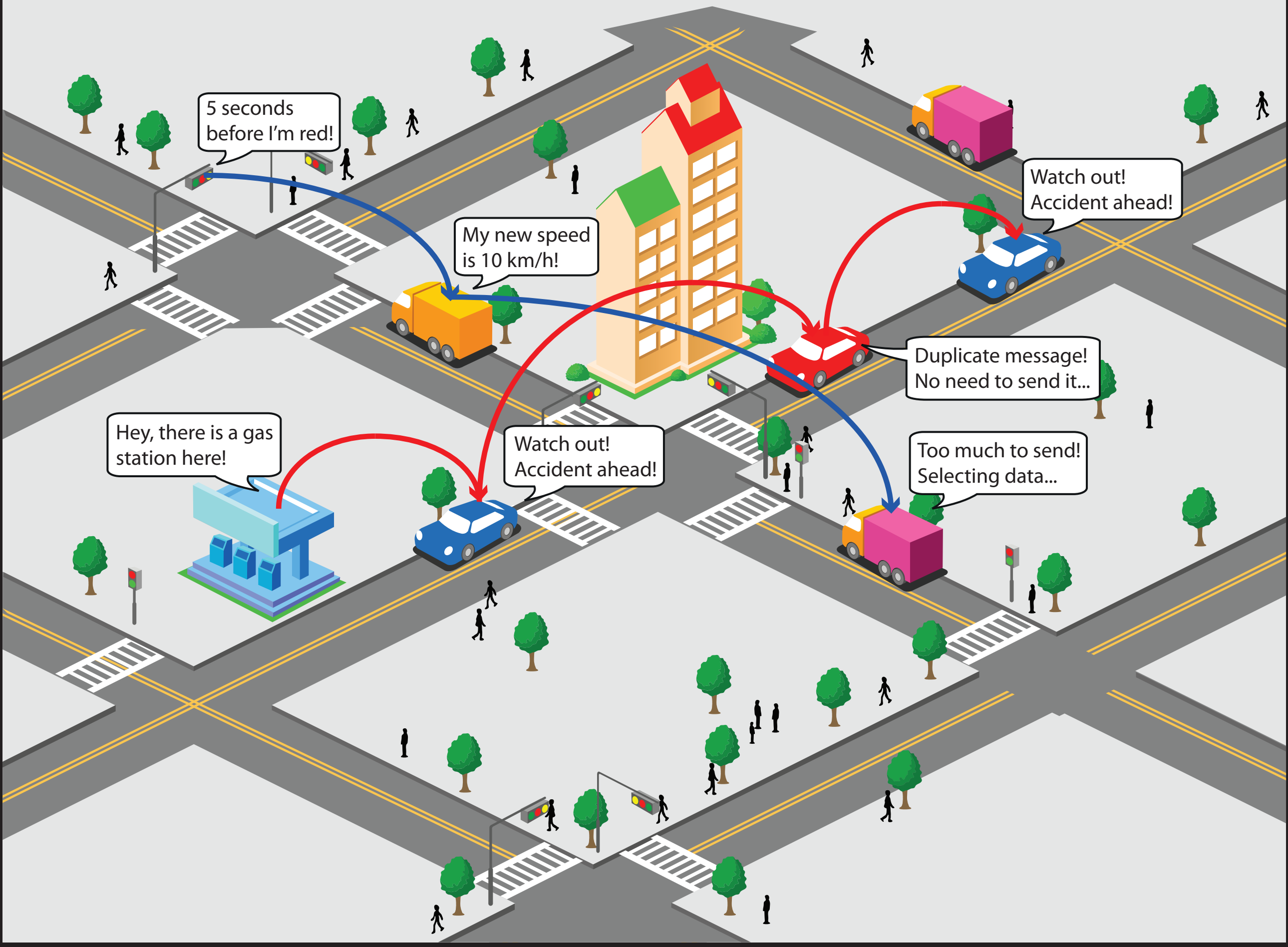
Data Dissemination in Vehicular Environments



Ramon S. Schwartz

Data Dissemination in Vehicular Environments

Ramon S. Schwartz

De promotiecommissie:

voorzitter en secretaris:	Prof. dr. ir. Anton J. Mouthaan
promotor:	Prof. dr. ing. Paul J. M. Havinga
referent:	Ir. Hans Scholten

leden:

Prof. dr. Frank E. Kargl	Universiteit Twente / Universität Ulm
Prof. dr. Falko Dressler	Universität Innsbruck
Prof. dr. ir. Bart van Arem	Technische Universiteit Delft
Dr. ir. Geert Heijenk	Universiteit Twente

CTIT Ph.D.-thesis Series No. 13-278
Centre for Telematics and Information Technology
University of Twente
P.O. Box 217, NL – 7500 AE Enschede

ISSN 1381-3617
ISBN 978-90-365-3571-7
DOI: 10.3990/1.9789036535717

Printed by Gildeprint Drukkerijen - Enschede
Cover design: Ramon S. Schwartz

DATA DISSEMINATION IN VEHICULAR ENVIRONMENTS

PROEFSCHRIFT

ter verkrijging van
de graad van doctor aan de Universiteit Twente,
op gezag van de rector magnificus,
Prof. dr. H. Brinksma,
volgens besluit van het College voor Promoties,
in het openbaar te verdedigen
op vrijdag 22 november 2013 om 12.45 uur

door

Ramon de Souza Schwartz

geboren op 17 maart 1984
te Vitória, Espírito Santo, Brazilië

Dit proefschrift is goedgekeurd door:
Prof. dr. ing. Paul J. M. Havinga (promotor)
Ir. Hans Scholten (referent)

Acknowledgments

Time flies… this is the best description of how I feel after having spent the last four years working towards a PhD degree. This work has only been possible thanks to the help that I received from various people throughout these years.

First of all, I would like to thank my supervisors Hans Scholten and Paul Havinga for guiding me and, most importantly, for giving me the priceless freedom to choose my own path. I would also like to thank Hylke for the fruitful discussions we had in the beginning of my PhD.

To all Pervasive Systems group members, I would like to leave here my gratitude for providing such a nice international working environment. I have learned uncountable new things during our PS discussion meetings, events, and lunch times.

Having contributed directly to my thesis, all the co-authors of papers published and used as basis for this thesis deserve a special acknowledgement: Anthony E. Ohazulike, Rafael R.R. Barbosa, Hylke W. van Dijk, Kallol Das, Geert Heijenk, Nirvana Meratnia, Christoph Sommer, and Falko Dressler.

I would like to thank Geert Heijenk, Frank E. Kargl, Falko Dressler and Bart van Arem for accepting being part of my committee. I feel honored to have such experts in my defense.

To my friends, I'm grateful for the numerous good moments we shared during this time and I hope we can have many others in the near future.

To my family: gostaria de agradecê-los muito pelo apoio incondicional em deixar-me buscar meu próprio caminho mesmo que isto signifique vivermos um pouco mais longe uns dos outros.

Finally, I want to thank my wife Rita for being every day on my side giving unconditional support, sharing all the good and bad moments during all these intense years and many more that are to come in our lives.

Ramon S. Schwartz
Juan-les-Pins, October 2013

Abstract

In the last few decades, Intelligent Transportation Systems (ITS) have been deployed to reduce congestion, enhance mobility, and help save lives. Among the various technologies incorporated is vehicular communication which consists in equipping vehicles with inexpensive wireless devices to enable a decentralized network composed by vehicles and infrastructure points. Such a vehicular network allows vehicles to extend their horizon of awareness to events that are beyond those that on-board sensors alone are able to detect.

In this context, one crucial task is the dissemination of data generated by a wide range of applications. On the one hand, safety applications are mostly related to hazardous situations. Therefore, they require a low dissemination delay and reliable delivery to all vehicles in the surroundings. On the other hand, non-safety applications, related to transport efficiency and infotainment, tolerate higher levels of delay, however, they also generate larger data volumes. Due to the limited channel capacity, the data must be selected prior to broadcasting according to the current level of interest of neighboring vehicles. This can be defined based on the current context such as the vehicles' direction and the age of the data being disseminated. In both categories, applications share the challenges raised by unique characteristics of vehicular networks such as the continual variation in density and predominant intermittent connectivity between vehicles. This thesis focuses on the development of data disseminating solutions that address these challenges while fulfilling the requirements of both safety and non-safety applications.

The main contributions of this thesis can be summarized as follows:

- ***A directional data dissemination protocol for highway scenarios*** that copes with disconnected highway scenarios while preventing the broadcast storm problem in dense networks. To achieve this goal, we propose a straightforward store-carry-forward algorithm for sparse networks and an optimized delay-based suppression technique for dense networks.
- ***A scalable directional data dissemination protocol for dense highway scenarios*** to tackle scalability issues in terms of number of transmissions when

increasing network densities are taken into account. To this end, we exploit the information contained in beacons to select the best available vehicles to forward messages.

- ***A scalable data dissemination protocol for both highway and urban scenarios*** which elaborates on aspects of multi-directional dissemination. We present an infrastructure-less protocol that combines a generalized delay-based suppression technique based on directional sectors and a store-carry-forward algorithm to support multi-directional data dissemination.

- ***A comparative study between fairness and efficiency as goals for data selection*** when the connectivity time or available bandwidth is not large enough for all data to be broadcast. Such data selection aims to maximize the utility (importance) gain of all vehicles. For this study, we propose a basic protocol to exchange messages between a pair of vehicles.

- ***A fair data dissemination protocol via synchronous broadcasting*** that distributes data utility fairly among vehicles in the neighborhood. To achieve this goal, synchronous broadcasting is used to prioritize messages according to a fairness criteria. This mechanism is also able to suppress the least relevant data, given a defined maximum network load allowed.

- ***A fair and adaptive data dissemination protocol*** that distributes data utility fairly over vehicles while adaptively controlling the network load. The protocol dynamically adjusts the intervals between consecutive broadcasts based on both data priority and network load. Both real-world experiments and simulations of realistic large-scale networks are used for validation.

Samenvatting

In de afgelopen decennia zijn Intelligente Transportsystemen (Intelligent Transport Systems, ITS) ingezet om verkeersopstoppingen te verminderen, mobiliteit te verbeteren en levens te redden. Een van de technologieën die gebruikt worden bij ITS is voertuigcommunicatie, waarbij voertuigen worden uitgerust met goedkope apparatuur voor draadloze communicatie. Daarmee kan een gedecentraliseerd netwerk worden gemaakt tussen voertuigen en infrastructuur. Zo'n netwerk stelt een voertuig in staat verder te kijken dan met de eigen sensoren mogelijk zou zijn geweest.

Het verspreiden van informatie voor uiteenlopende toepassingen is daarbij van cruciaal belang. Enerzijds zijn er de veiligheidstoepassingen waarbij informatie over gevaarlijke situaties zonder grote vertraging betrouwbaar moet worden verspreid naar voertuigen in de omgeving. Anderzijds zijn er niet aan veiligheid gerelateerde toepassingen, zoals "infotainment" en efficiëntie van het verkeer, die deze eisen niet stellen aan de communicatie, maar waarbij wel grotere hoeveelheden informatie wordt geproduceerd. Omdat de capaciteit van de draadloze communicatie beperkt is, moet een afweging worden gemaakt welke informatie verzonden zal worden. Deze afweging is afhankelijk van de vraag van andere voertuigen, die kan worden bepaald op basis van de context van de betrokken voertuigen. Voorbeelden zijn plaats van vertrek en aankomst en de mate van actualiteit van de informatie. Beide categorieën van toepassingen hebben te maken met de uitdagingen die voortkomen uit eigenschappen van netwerken van voertuigen, zoals voortdurende veranderingen in de dichtheid van het verkeer en onstabiele communicatie met onderbrekingen.

Dit proefschrift richt zich op oplossingen voor het verspreiden van informatie in voertuignetwerken die voldoen aan de eisen van veiligheids- en niet-veiligheidstoepassingen.

De belangrijkste bijdragen van dit proefschrift kunnen als volgt worden samengevat:

- ***Een directioneel data disseminatieprotocol voor snelweg scenario's*** dat bij

netwerken met hoge dichtheid het probleem van "broadcast storm" vermijdt. Om dit te bereiken introduceren we een "store-carry-forward" algoritme voor netwerken met lage dichtheid en een geoptimaliseerde vertraging-gebaseerde suppressietechniek voor dichte netwerken.

- ***Een schaalbaar directioneel data disseminatieprotocol voor hoge dichtheid snelweg scenario's*** dat het aantal uitzendingen beperkt bij toenemende dichtheid van het netwerk. Daartoe wordt gebruik gemaakt van de informatie die bakens uitzenden om uit de beschikbare voertuigen het beste voertuig te kiezen om berichten door te sturen.

- ***Een schaalbaar data disseminatieprotocol voor snelweg en stedelijke scenario's*** waarbij aspecten van multi-directionele disseminatie aan de orde komen. Het geïntroduceerde infrastructuurloze protocol combineert een algemene vertraging-gebaseerde suppressietechniek en een "store-carry-forward" algoritme om multi-directionele data disseminatie te ondersteunen.

- ***Een vergelijkende studie tussen "fairness" en efficiëntie bij data selectie*** wanneer de tijdsduur van een verbinding of de beschikbare bandbreedte niet voldoende zijn om alle data te versturen. Het doel van data selectie is het nut ("utility") van de informatie voor alle betrokken voertuigen te maximaliseren.

- ***Een eerlijk ("fair") data disseminatieprotocol door middel van synchrone "broadcasts"*** dat informatie verspreid naar voertuigen in de buurt waarbij het nut van de informatie eerlijk wordt verdeeld. Om dit te bewerkstelligen wordt op basis van eerlijkheidscriteria synchrone "broadcasts" gebruikt om berichten een prioriteit te geven. Gegeven de maximale netwerkbelasting, zullen de minst belangrijke berichten (met een lage prioriteit) worden onderdrukt.

- ***Een eerlijk ("fair") en adaptief data disseminatieprotocol*** dat informatie verspreid naar voertuigen in de buurt, waarbij het nut van de informatie eerlijk wordt verdeeld en de netwerkbelasting adaptief wordt aangepast. Op basis van de prioriteit van de informatie en de netwerkbelasting wordt het interval tussen opeenvolgende uitzendingen aangepast. Het protocol wordt gevalideerd door zowel experimenten als simulaties van realistische grootschalige netwerken.

Contents

CHAPTER 1

Introduction

The number of vehicles operating on the roads in the world has passed in 2010 the impressive mark of 1 billion units, just 24 years after reaching 500 million in 1986 [1]. Such an immense road network has brought comfort to numerous new drivers but also accounted for approximately 1.24 million deaths in 2010 [2]. Along with these numbers come the increasing level of CO_2 emission and billions of hours wasted in traffic congestion [3].

In view of these problems, Intelligent Transportation Systems (ITS) have been deployed with the ultimate goal of reducing congestion, enhancing mobility, and helping save lives [4]. These systems incorporate a broad range of wireless and wire line communications, information processing, advanced computing, and electronics technologies. One of the most prominent technologies is vehicular communication [5, 6]. Both industry and academia advocate equipping vehicles with inexpensive wireless devices to enable not only the communication between vehicles but also between vehicles and infrastructure. Such a decentralized network, known as Vehicular Ad-hoc Network (VANET), allows vehicles to extend their horizon of awareness to events that are beyond those that on-board sensors alone are able to detect.

Vehicular ad-hoc networks are expected to support the development of a wide range of applications related to safety, transport efficiency, and even infotainment [7]. In its basic form, vehicles periodically broadcast *beacons* that are essentially status messages containing information such as the vehicle's position and speed [8]. These messages serve as heartbeat in order for each vehicle to be aware of other neighboring vehicles in the vicinity. On top of that, more complex applications exploit the local awareness acquired by these *beacons* to *disseminate* their produced data to potentially interested vehicles that are situated in much farther locations within the road network. In this way, a multi-hop network is formed, where each vehicle continuously gathers, processes, and disseminates data to other vehicles in the neighborhood.

This thesis is motivated by the challenges that arise when disseminating data in vehicular environments. The aim is to design data dissemination solutions that fulfill the requirements of a wide variety of applications. In the remainder of this chapter, we elaborate on the characteristics of vehicular ad-hoc networks and on the key points and limitations of the underlying wireless technology in Section 1.1. In Section 1.2, we outline application requirements for data dissemination. Section 1.3 describes the research objective of this thesis and how we address our research questions. Next, we summarize the main contributions of this work in Section 1.4. Finally, an overview of the thesis is given in Section 1.5.

1.1 Vehicular ad-hoc network

Vehicular ad-hoc networks, or simply vehicular networks, consist of vehicles and infrastructure points (roadside units) equipped with wireless devices. Figure 1.1 shows an example where both vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications take place. Two flows of data are disseminated in a multi-hop fashion through a few vehicles before being sent to a road side unit placed either in a smart traffic light or gas station.

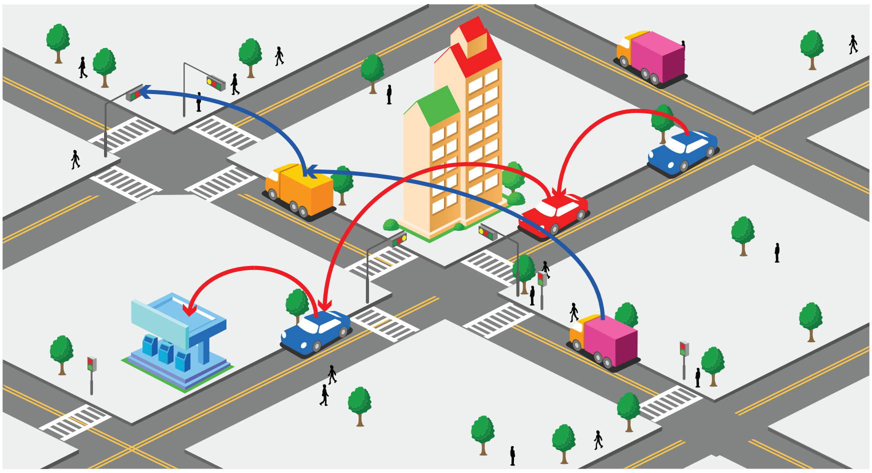


Figure 1.1: Example of vehicular ad-hoc network

1.1.1 Characteristics

Given their dimension and high mobility of vehicles, vehicular networks present the following unique characteristics [9, 10]:

- ***Density variation***: vehicular networks are in a constant state of flux. The network density varies from being very sparse (e.g., free-flow traffic) to very dense (e.g., traffic jams) in a very short period of time.

- ***Intermittent connectivity***: the highly dynamic nature of vehicular networks leads to a predominant intermittent connectivity between vehicles. Due to the high speed of vehicles, the connectivity duration time varies from a few seconds to a few minutes.

- ***Data locality***: for several applications, the data produced by vehicles is usually associated and relevant to a certain geographical region of the road network. Each vehicle is assumed to be equipped with means to derive its own geographical location, e.g., with a GPS device.

- ***Predictable pattern***: vehicles move along known paths, often in a predictable manner. Therefore, applications can leverage contextual information such as the vehicle's direction and speed to deliver information to target regions.

- ***No power contraints***: in contrast to traditional wireless mobile ad-hoc networks, energy is not of primary concern. Vehicles can be used as a source of electric power continually recharged by fuel.

- ***Broadcast***: since the acquired data is usually of interest to a number of vehicles in the region, e.g., data about accidents, broadcasting becomes the predominant communication paradigm for most applications.

1.1.2 Overview of underlying wireless technology

Due to the specific characteristics of vehicular networks, efforts in the United States, Europe and Japan have been put to establish a new set of communication standards exclusively meant for vehicular communication. Such standards are key to promote interoperability between equipment developed by distinct groups and countries. In the U.S., 75 MHz of bandwidth in the 5.9 GHz band has been allocated with the specific goal of supporting dedicated short-range communications (DSRC) for Intelligent Transportation Systems (ITS) [11].

In Europe, different ranges of bandwidth also in the 5 GHz band have been allocated for ITS applications [12].

In both American and European standards, one radio channel within the bandwidth allocated is dedicated exclusively for safety applications. The reason for such separation lies in guaranteeing that messages related to hazardous situations are not hindered by messages generated by non-safety applications, thereby allowing for an effective prevention of accidents. In the U.S., the bandwidth is divided into seven channels of 10 MHz, where one is the control channel and the remaining are service channels. The control channel is used for the exchange of control and safety messages, whereas service channels are used for the exchange of messages generated by non-safety applications after coordinating their use in the control channel. A similar strategy is adopted in the European standard. The bandwidth is divided into ITS-G5A (30 MHz) reserved for safety applications and ITS-G5B (20 MHz) reserved for non-safety applications. Another class of bandwidth is IT-G5C (255 MHz) reserved for other ITS applications. However, IT-G5C is only meant for the communication between infrastructure and mobile nodes, thereby excluding vehicle-to-vehicle communication.

The de facto and approved physical (PHY) and medium access control (MAC) layers for vehicular communication are specified in the IEEE 802.11p standard [13]. The standard defines data rates from 3 to 27 Mbps and transmission power values that could reach up to a theoretical 1 km of range. IEEE 802.11p is an amendment of the IEEE 802.11 family of standards with specific modifications to cope with the highly dynamic environment that vehicular networks present. In the MAC layer, modifications are mainly focused on reducing the overhead to allow vehicles to immediately communicate without having to join a Basic Service Set (BSS). Also, the MAC layer includes the Enhanced Distributed Channel Access (EDCA) mechanism for Quality of Service (QoS) differentiation of messages, which is similar to the mechanism described in the IEEE 802.11e amendment. The PHY layer is essentially based on the OFDM PHY defined for IEEE 802.11a, however, with a 10 MHz wide channel instead of 20 MHz in order to prevent inter-symbol interferences within the vehicle's own transmissions in vehicular environments [14]. In addition, some optional enhanced channel rejection requirements are specified to improve the immunity of the communication system to out-of-channel interferences.

Efforts on the standardization of additional layers include the IEEE 1609 set of standards that specify multichannel operation, networking services, resource manager and security services [11]. The combination of IEEE 802.11p and the IEEE 1609 protocol suite is denoted as WAVE (Wireless Access in Ve-

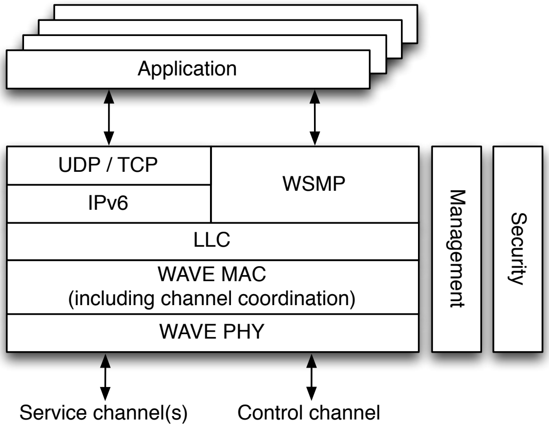


Figure 1.2: Overview of the WAVE protocol stack

hicular Environments). Figure 1.2 gives an overview of the WAVE protocol stack. In addition to the traditional IEEE 802.11 stack components and Internet protocols, the WAVE Short Message Protocol (WSMP) is included. WSMP is meant to enable high-priority, time-sensitive communication by allowing applications to directly control certain parameters of the radio resource to maximize the probability that all the implicated parties will receive the messages in time. The WSMP protocol is meant to handle safety messages whereas non-safety messages can be sent with either WSMP or with the typical UDP or TCP/IP protocols.

1.1.3 Limitations of the technology

Although designed to cope with specific characteristics of vehicular networks, the IEEE 802.11p standard inherits limitations present in other amendments of the 802.11 family of standards. Challenges arise especially when relying on broadcast communication, which is the predominant communication paradigm in vehicular environments. Broadcasting is highly unreliable due to the lack of acknowledgment in the carrier sense multiple access with collision avoidance (CSMA/CA) mechanism. The *hidden terminal problem* (shown in Figure 1.3) is also predominant due to the lack of mechanisms such as the Request to Send (RTS) and Clear to Send (CTS) used to reduce the effects of the hidden terminal problem in unicast communication.

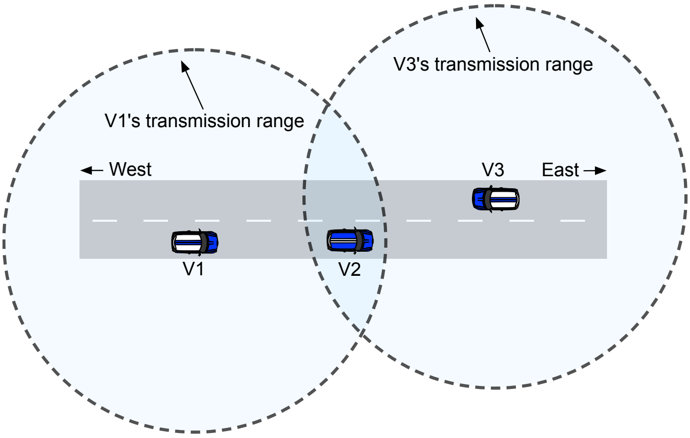


Figure 1.3: The hidden terminal problem. In this example, v_1 and v_3 can communicate with v_2 but are *hidden* from each other. The hidden terminal problem occurs when v_1 and v_3 sense the medium idle and start transmitting, thereby causing a collision at v_2.

Another technical limitation comes from the lack of a congestion control mechanism. Periodic one-hop *beacons*, messages referred to as Basic Safety Messages (BSMs) in the U.S. or Cooperative Awareness Messages (CAMs) in Europe, are expected to serve as basis for various safety applications and can alone lead to the exhaustion of the wireless channel capacity in dense networks [8, 15]. The available bandwidth might be further reduced if a single wireless transceiver is used. As described earlier, there is one control channel for safety applications and a few service channels for non-safety applications. If vehicles are equipped with only one transceiver, a periodic switching between channels is used to guarantee that safety messages are sent and received with upper delay bounds. This is achieved by defining that in the first 50 ms within every interval of 100 ms, vehicles will be tuned to the control channel. Time synchronization can be achieved, for example, with a GPS time signal. The consequence is the decrease of nearly half of time dedicated for safety applications. Congestion control solutions typically focus on both transmission power control and transmission rate control. At the moment of writing, both aspects are being considered in the ETSI European standardization by means of the Decentralized Congestion Control (DCC) mechanism [12].

1.2 Application requirements for data dissemination

Data dissemination in vehicular environments is sometimes broadly referred to as the process of obtaining, transporting and aggregating data [5]. In this work, however, we refer to data dissemination as the process of *transporting* information to interested vehicles. The data is mostly of interest to a number of vehicles in the region that can be one-hop to many hops away from the location where the information has been generated. Therefore, the *multi-hop broadcast communication* paradigm is used. We consider the dissemination of data generated by applications upon the occurrence of events, i.e., event-driven messages, rather than *beacons* that are limited to provide one-hop neighborhood awareness [16].

Defining what an "interested" vehicle is clearly depends on the requirements of each application. Applications are commonly classified as either safety or non-safety applications. Because of the critical aspect of safety applications, this separation is reflected in every standardization effort as mentioned in the previous section, where separate channels are allocated exclusively for safety messages. Each category has the following requirements [5, 7, 17, 18]:

- ***Safety***: applications in this class are mostly related to hazardous situations. The information is typically expected to fit into **one or few messages** and disseminated with strict requirements for **low latency**. The spatial scope is usually limited to a **few meters** (critically affected vehicles) to a **few kilometers** (vehicles in the surroundings). Given its high priority, **all vehicles** in the region must be warned about any safety-related incident.

  Examples of applications are warning of accidents, poor road condition, collisions in intersection, emergency vehicles (EVs) approaching, and so forth.

- ***Non-safety***: comprises any application that is not safety-related. Applications of this class are expected to generate much **larger data** volumes, however, with **higher delay tolerance**. The spatial scope is more flexible and highly dependent on the application. However, the information generated is typically of interest to vehicles located up to a **few kilometers** from the event location. Also, the information is interesting only to **selected vehicles**. For example, information about available parking is of higher interest to vehicles actually going to park near the location related to the information. Because of the broad classification, non-safety applications are normally further divided into *traffic efficiency* and *infotainment*. Examples of applications of each subclass are:

- *Traffic efficiency*: up-to-date traffic information, route advisory, speed limit notification, traffic light optimal speed advisory, etc.

- *Infotainment*: convenience information such as parking availability, points of interest, road map, local commerce information. It also includes general applications such as media and file downloading, Internet access, etc.

	Safety	Non-safety
Spatial scope	Critical: 250 meters Non-critical: few kilometers	Few to many kilometers
Temporal scope	Critical: < 100 milliseconds Non-critical: few seconds	Few seconds to weeks
Interest scope	All vehicles	Selected vehicles
Data amount	Few messages	Many messages

Figure 1.4: Overview of requirements, based on [7, 18]

An overview of general application requirements is shown in Figure 1.4. Along with specific requirements of each class of application, **scalability** is a major concern due to frequent network density variations and has a direct influence on meeting, in particular, **latency** requirements. In dense networks, disseminating data based on a pure flooding scheme results in excessive redundancy, contention, and collision rates [19], which is referred to as the *broadcast storm problem*. Conversely, in sparse networks vehicles may face network disconnections and intermittent connectivity when the transmission range employed cannot reach other vehicles farther in the region of interest. Especially for non-safety applications generating many data messages, such limited connectivity raises challenges with respect to which information to broadcast and at which moment in time.

Finally, mechanisms ensuring **security and privacy** are required to prevent attackers from inserting malicious information in the network and at the same time to protect the identity of the driver [20].

1.3 Research objectives

The main focus of this thesis is to study data dissemination solutions for vehicular environments that fulfill the requirements of both safety and non-safety applications. Although security and privacy are important requirements, they are out of the scope of this thesis. Instead, we concentrate our efforts on *scalable* data dissemination solutions that work seamlessly in both sparse and dense vehicular networks. We further limit our scope to the case of vehicle-to-vehicle communication relying, thereby assuming the presence of infrastructure-less vehicular networks and local knowledge only. This is reasoned by the fact that especially in highways and during early stage deployment in urban scenarios, it is desirable that data dissemination solutions work in the absence of any infrastructure support. We also restrict the broad set of non-safety applications to the case of dissemination of data acquired by on-board sensors where vehicles collaboratively build and share information about traffic efficiency and convenience applications. Therefore, we do not address multimedia streaming or Internet access applications, which normally deal with more stringent requirements of real-time communication and are usually assumed to rely on infrastructure [18].

Considering the scope above, the main research question of this thesis is:

> *How to achieve scalable data dissemination in infrastructure-less vehicular environments while fulfilling specific requirements of both safety and non-safety applications?*

In view of the distinct requirements between safety and non-safety applications, we approach our main research question by answering the following two sub research questions:

> ***(RQ.1) Safety***: *how to disseminate data in a timely manner to all vehicles in the affected region while minimizing the number of transmissions?*

> ***(RQ.2) Non-safety***: *how to select and disseminate the most relevant data to interested vehicles while controlling the network load?*

1.3.1 Hypotheses

In order to answer research question ***(RQ.1)***, we start from the hypothesis that in sparse networks we can cope with intermittent connectivity by exploiting the mobility of vehicles to *store, carry, and forward* messages to further vehicles on the road. In addition, the presence of *beacons* can be *exploited* to achieve efficient selection of neighboring vehicles to forward messages, especially in dense networks.

We address question ***(RQ.2)*** with the hypothesis that when considering vehicles with conflicting data interests, a data dissemination solution should rely on concepts of *fairness*. We argue that in this way, we can maximize individual interest gains and prevent situations where only a subset of vehicles receive relevant information. In addition, to cope with both sparse and dense networks, a mechanism to control the *network load* should *adaptively* adjust its parameters according to the current network conditions.

1.3.2 Approach

We approach the research questions of this thesis by exploring data dissemination protocols placed on top of the WAVE protocol stack. Therefore, no modification is required in the IEEE 802.11p standard for vehicular communication.

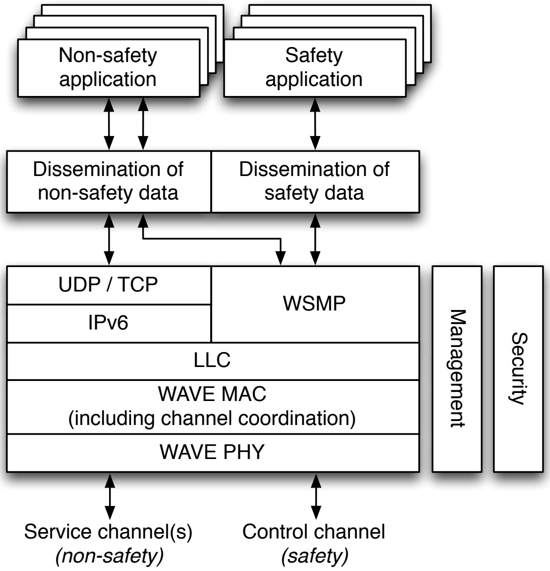


Figure 1.5: The WAVE protocol stack + data dissemination modules

In view of the separation of safety and non-safety radio channels and underlying network protocols in the standard, we define one separate module for each type of application to take the responsibility of coordinating the transmission of broadcast messages, as shown in Figure 1.5. Safety messages are sent with the WAVE Short Message Protocol (WSMP) whereas non-safety messages are sent with either WSMP or typical Internet protocols. In the PHY layer, safety messages are sent in the control channel whereas non-safety messages are sent in one or multiple service channels.

1.4 Contributions

The contributions with respect to data dissemination for **safety** applications can be summarized as follows:

(Contribution 1) A directional data dissemination protocol for highway scenarios: we present a data dissemination protocol that deals with data dissemination in both dense and sparse vehicular networks. Our main focus is on coping with disconnected highway scenarios while preventing the broadcast storm problem in dense networks. To achieve this goal, we propose a straightforward store-carry-forward communication model for sparse networks and an optimized delay-based suppression technique for dense networks. This work appeared in [21, 22]:

- R.S. Schwartz, R.R.R. Barbosa, N. Meratnia, G. Heijenk, and H. Scholten. *A Simple and Robust Dissemination Protocol for VANETs*. In: 16th European Wireless Conference, 12-15 April 2010, Lucca, Italy. pp. 214-222.
- R.S. Schwartz, R.R.R. Barbosa, N. Meratnia, G. Heijenk, and H. Scholten. *A directional data dissemination protocol for vehicular environments*. Elsevier Computer communications, 34 (17), 2011. pp. 2057-2071.

(Contribution 2) A scalable directional data dissemination protocol for dense highway scenarios: we further elaborate on the broadcast storm problem in dense networks by designing a suppression technique that tackles scalability issues in terms of number of transmissions when increasing network densities are taken into account. To this end, we exploit the information contained in *beacons* to select the best available vehicles to forward messages. This work appeared in [23]:

- R.S. Schwartz, K. Das, H. Scholten, and P. Havinga. *Exploiting beacons for scalable broadcast data dissemination in VANETs*. In: Proceedings of the 9th ACM international workshop on Vehicular inter-networking, systems, and applications (VANET), 25 June 2012, Low Wood Bay, Lake District, United Kingdom. pp. 53-62.

(Contribution 3) A scalable data dissemination protocol for both highway and urban scenarios: we adapt and extend concepts used in the two previous contributions for the case of multi-directional dissemination, thereby tackling scalability issues in both highway and urban scenarios. We present an infrastructure-less protocol that combines a generalized delay-based suppression technique based on directional sectors and a store-carry-forward algorithm to support multi-directional data dissemination. This work has been submitted to:

- R.S. Schwartz, H. Scholten, and P. Havinga. *A Scalable Data Dissemination Protocol for Both Highway and Urban Vehicular Environments*. In: Springer EURASIP Journal on Wireless Communications and Networking, accepted for publication, submitted in February 2013.

The contributions with respect to data dissemination for **non-safety** applications can be summarized as follows:

(Contribution 4) A comparative study between fairness and efficiency as goals for data selection: we study the trade-offs between fairness and efficiency to tackle the problem of selecting data when the connectivity time or available bandwidth is not large enough for all data to be broadcast. Such data selection aims to maximize the utility (importance) gain of all vehicles. For this study, we propose a basic protocol to exchange messages between pair of vehicles. This work appeared in [24]:

- R.S. Schwartz, A.E. Ohazulike, H.W. van Dijk, and H. Scholten. *Analysis of Utility-Based Data Dissemination Approaches in VANETs*. In: 4th International Symposium on Wireless Vehicular Communications (WIVEC) - VTC Fall, 5-6 September 2011, San Francisco, CA, USA. pp. 1-5.

(Contribution 5) A fair data dissemination protocol via synchronous broadcasting: we design a data dissemination protocol that distributes data utility fairly among vehicles in the neighborhood. To achieve this goal, we propose a synchronous periodic dissemination protocol that is used to prioritize broadcast messages according to a fairness criteria. This mechanism is also able to suppress the least relevant data, given a defined maximum network load allowed. This work appeared in [25]:

- R.S. Schwartz, A.E. Ohazulike, and H. Scholten. *Achieving Data Utility Fairness in Periodic Dissemination for VANETs*. In: IEEE 75th Vehicular Technology Conference (VTC Spring), 6-9 May 2012, Yokohama, Japan. pp. 1-5.

(Contribution 6) A fair and adaptive data dissemination protocol: we take one step further and design a data dissemination protocol that distributes data utility fairly over vehicles while adaptively controlling the network load. The protocol dynamically adjusts the intervals between consecutive broadcasts based on both data priority and network load. In addition, we show the applicability of the protocol by giving example of utility functions for two Traffic Information Systems (TIS) applications: parking-related and traffic information applications. The protocol is validated with both real-world experiments and simulations of realistic large-scale networks. This work has partially appeared in [26] and partially submitted to Elsevier Ad Hoc Networks journal:

- R.S. Schwartz, A.E. Ohazulike, C. Sommer, H. Scholten, F. Dressler, and P. Havinga. *Fair and adaptive data dissemination for traffic information systems*. In: 4th IEEE Vehicular Networking Conference (VNC), 14-16 Nov 2012, Seoul, South Korea. pp. 1-8.
- R.S. Schwartz, A.E. Ohazulike, C. Sommer, H. Scholten, F. Dressler, and P. Havinga. *On the applicability of fair and adaptive data dissemination in traffic information systems*. In: Elsevier Ad Hoc Networks, accepted for publication, submitted in April 2013.

1.5 Organization of the thesis

The remainder of this thesis is organized as shown in Figure 1.6. Chapter 2 gives an overview of the state-of-the-art data dissemination solutions by describing their characteristics and open issues for both safety and non-safety applications. In Chapter 3, we describe in detail our contributions 1, 2, and 3 for data dissemination for safety applications in order to answer research question ***(RQ.1)***. Chapter 4 describes our contributions 4, 5, and 6 for data dissemination for non-safety applications in order to answer research question ***(RQ.2)***. Finally, Chapter 5 concludes this thesis with a summary and directions for future work.

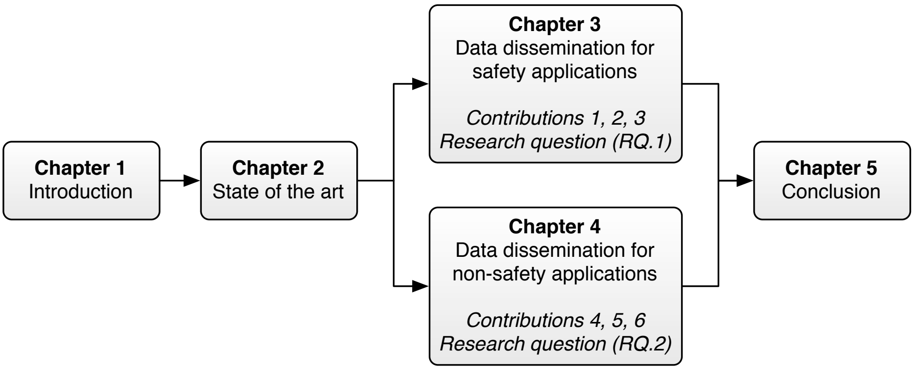


Figure 1.6: Organization of the thesis

CHAPTER 2

State of the art

In this chapter, we review state-of-the-art solutions related to data dissemination in vehicular networks and outline issues not yet addressed in the literature. In Section 2.1, we discuss solutions designed for safety applications. Section 2.2 reviews solutions for non-safety applications. Finally, Section 2.3 closes this chapter with concluding remarks.

2.1 Data dissemination for safety applications

Various solutions for safety applications in VANETs have been proposed to cope with message dissemination under different traffic conditions. In dense networks, broadcast suppression techniques have been proposed to prevent the so-called *broadcast storm problem*. The ultimate goal is to select only the set with the minimum number of vehicles to rebroadcast and disseminate a message within the region of interest.

In the context of Mobile Ad-hoc Networks (MANETs), several solutions to address this problem were proposed and outlined in [19, 27]. In [27], authors present a comprehensive comparison study of various broadcasting techniques in MANETs organized into four categories: (i) *simple flooding methods*, without any form of suppression; (ii) *probability based methods*, that rely on network topology information to assign a probability for each rebroadcast; (iii) *area based methods*, which use distance information to decide which nodes should rebroadcast; and (iv) *neighbor knowledge methods*, which maintain state on the neighborhood via periodic *hello* messages to decide on the next forwarding node. However, these solutions are mostly concerned with providing means for route discovery with minimum extra network load and, therefore, do not take into account the highly dynamic environment present on roads, neither exploit specific characteristics of vehicular networks such as the predictable mobility pattern of vehicles' movements.

In VANETs, it is generally assumed that each broadcast data message relates to a certain event of a specific geographical region and, thus, it is targeted mostly to vehicles traveling through that region. With this goal, protocols that rely on positioning information falling into categories (iii) and (iv) are most suitable. In category (iii), nodes in the Location-Based scheme [19] rebroadcast whenever the additional coverage is higher than a pre-defined threshold. In category (iv), most protocols require nodes to share 1-hop or 2-hop neighborhood information with other nodes [28, 29, 30]. This is particularly not suitable in vehicular environments, since such information can quickly become outdated due to the high speed of vehicles. In addition, adding neighborhood information to periodic messages results in high network overhead. As pointed out in [16], decreasing message overhead is crucial for leaving sufficient bandwidth for even-critical messages. In view of these drawbacks, several protocols have been proposed specifically for VANET applications. Such protocols present lightweight solutions in terms of overhead and elaborate on previous solutions in category (iii) such as in [19] in order to control, based on distance, the thresholds determining when vehicles should rebroadcast. In the following, we select and describe a few of these efforts. For a complete survey of solutions, we refer the reader to [31].

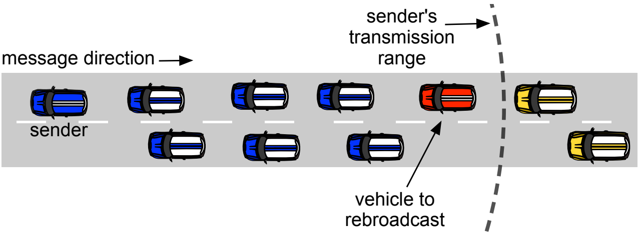
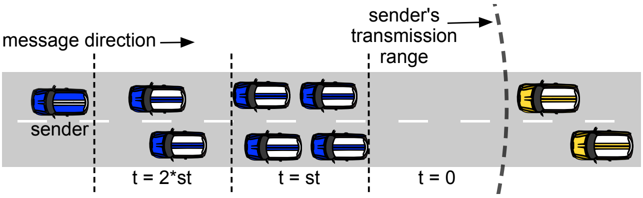


Figure 2.1: The common goal of suppression techniques in vehicular networks: select only the farthest vehicle in each target direction to rebroadcast

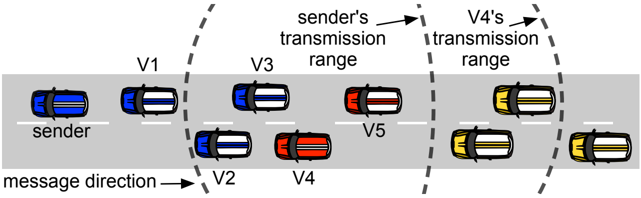
The common approach to reduce broadcast redundancy and end-to-end delay in dense vehicular networks is to give highest priority to the most distant vehicles towards the message direction, as shown in Figure 2.1. In [32], three ways of assigning this priority are presented: Weighted p-Persistence, Slotted 1-Persistence and Slotted p-Persistence. In the first scheme, the farthest vehicles rebroadcast with highest *probability*. In the second approach, vehicles are assigned to different time slots depending on their distance to the sender, where vehicles with highest priority are given the shortest *delay* before rebroadcasting. Finally, the third approach mixes probability and delay by giving vehicles with highest priority the shortest delay and highest probability to rebroadcast. In delay-based schemes, vehicles assigned to later time slots have time to cancel their transmissions upon the receipt of an echo. This would be an indication that the information has already been disseminated and redundant rebroadcasts can be *suppressed*. Notably, to achieve the lowest possible end-to-end delay, deterministic approaches such as Slotted 1-Persistence should be preferred over probabilistic methods such as Weighted p-Persistence and Slotted p-Persistence. The reason lies in always guaranteeing that the farthest vehicle is chosen, which is not the case with probabilistic-based methods.

Delay-based schemes have been used in several other works with the goal of reducing rebroadcast redundancy, e.g., [33, 34, 35]. In [33], the Contention-Based Forwarding scheme (CBF) is presented. Authors focus on a distributed delay-based scheme for mobile ad hoc networks that requires no periodic messages. In [34], the Urban Multi-hop Broadcast (UMB) protocol is designed to cope with broadcast storm, hidden node, and reliability problems of multi-hop broadcast in urban areas. UMB has a special operation mode for scenarios with intersections. Nevertheless, it relies on the same time slot principle for directional data dissemination.

Although efficient in tackling the broadcast storm problem, delay-based schemes still present scalability issues when not employed with optimal parameters. One clear limitation in most schemes is the inability to dynamically choose the optimal value for the number and boundaries of the time slots used. As shown in Figure 2.2(a), time slots are usually matched to geographical regions within the transmission range of the sender. The farther the vehicle is, the lower is the time t waited before rebroadcasting the message from the sender, where st represents the pre-defined slot time. However, this can clearly lead to an uneven distribution of vehicles in each time slot. Since transmissions in a single time slot occur nearly simultaneously (see [36]) and cannot be canceled, the level of rebroadcast redundancy and collision is unnecessarily increased. To cope with collisions, authors in [37] introduced the concept of *micro* slots to separate in time transmissions assigned to a single time slot. Another consequence of relying on fixed time slot parameters is that there might simply be no vehicle in one of the time slots, thereby increasing end-to-end delay of a message. In this line, the work in [38] introduces a means to control the number of time slots according to the network density. However, authors do not cope with the problem of nearly simultaneous transmissions in a single time slot.



(a) Uneven distribution of vehicles among time slots



(b) Sub-optimal vehicle selection in a centralized approach

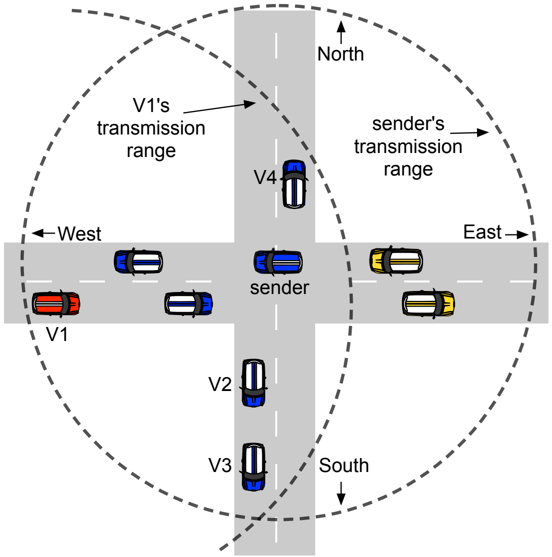
Figure 2.2: Overview of problems with typical delay-based suppression schemes

One way to tackle the problem of uneven distribution of vehicles among time slots is to adopt a centralized approach for selecting the next relay vehicle. This is generally achieved with typical periodic *hello* messages containing the vehicle's location and protocol-specific information. Alternatively, protocols can make use of *beacons*, referred to as Basic Safety Messages (BSMs) in the U.S. or Cooperative Awareness Messages (CAMs) in Europe, that are expected to coexist with other systems in the vehicle and serve with the same purpose of providing neighbors' awareness. In [39], the protocol proposed aims to classify vehicles into groups and select the relay vehicle with the best line-of-sight of each group. In [16], the Emergency Message Dissemination for Vehicular environments (EMDV) protocol combines both centralized and distributed approaches. In EMDV, the sender determines the next relay vehicle based on neighborhood information received from *beacons*. The remaining vehicles still follow a delay-based scheme to rebroadcast in case the transmission from the selected vehicle fails. However, one problem arises in centralized approaches when vehicles transmit messages with different power levels, as shown in Figure 2.2(b). In this scenario, v_5 is the farthest vehicle able to rebroadcast the message received from the sender. However, since v_5 employed a lower power level to send its periodic *beacons*, the sender could not be aware of v_5's presence and mistakenly chooses v_4 as the next relay vehicle. The direct consequence of such a mistake is a sub-optimal vehicle selection, leading to higher end-to-end delays. Finally, authors in [40] aim to solve these limitations by letting only the farthest (last) vehicle rebroadcast with The Last One method (TLO). In case the last vehicle fails, after a time threshold the protocol repeatedly defines the next farthest vehicle until the message is successfully broadcasted. Although a distributed approach is used in TLO, authors do not discuss how the threshold value is chosen. In addition, they do not present alternatives for improving end-to-end delay, e.g., by letting more than one vehicle rebroadcast in a single time slot in case of failed transmission or inaccurate positioning information.

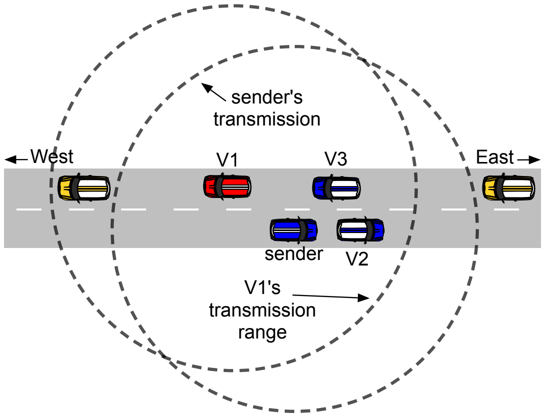
To the best of our knowledge, the DOT scheme [23] that we present in Section 3.2 pioneered in proposing a precise control of the time slots' density by exploiting the presence of periodic *beacons*. As mentioned earlier, *beacons* are expected to be inevitably periodically transmitted in order to increase cooperative awareness in safety applications [41]. Authors in [42] had later a similar insight of time slots' density control with the DAZL protocol.

Another problem when relying on time slots schemes arises when the message must be disseminated to multi-directions, as shown in Figure 2.3. In Figure 2.3(a), vehicles follow a typical time slot scheme based on distance. Therefore, the most distance vehicle from the sender, i.e., vehicle v_1, has the highest priority to rebroadcast in the neighborhood. However, such a naive solution clear prevents the dissemination of the message to both north and south directions, as vehicles v_2, v_3, and v_4 would cancel their rebroadcasts upon hearing the early transmission from v_1. The same problem occurs in a highway scenario as shown in Figure 2.3(b), where the rebroadcast performed by v_1 prevents the dissemination of the message to the other direction where vehicles v_2 and v_3 are located. This problem is addressed in [43], however, with no support for disconnected networks.

All suppression schemes still depend on additional measures to cope with sparse disconnected networks when the transmission range does not reach farther vehicles in each possible road direction. The typical approach to cope with disconnected networks is to assign selected vehicles the task of storing, carrying, and forwarding messages when new opportunities emerge. The *store-carry-forward* paradigm is mostly present in works falling in the area of Delay Tolerant Networks (DTN) and opportunistic networks. In its simplest form,



(a) Incorrect suppression in urban scenarios



(b) Incorrect suppression in highway scenarios

Figure 2.3: Limitations of typical time slot schemes for multi-directional dissemination

an Epidemic Routing is used [44], where flooding is used to disseminate messages throughout the network. In this approach nodes exchange data as soon as new neighbors are discovered. The Spray Routing [45] generates only a small number of message copies in order to ensure that the number of transmissions are small and controlled. In the context of Pocket Switched Networks (PSNs), where the nodes are devices carried by people, the BUBBLE algorithm is proposed [46]. It takes into account people's social relationships to select the nodes that can best relay messages. However, these approaches were designed assuming a different mobility model from the one present in VANETs, as they usually consider a combination of the mobility of pedestrians, bicycles, and cars. In VANETs, the mobility of vehicles is constrained to single or multiple roads and by well-defined rules. Therefore, in order to achieve optimal results, more tailored solutions are needed.

A few works apply the store-carry-forward mechanism specifically for vehicular networks [47, 48, 49, 50, 51, 22]. In [47], the Distributed Vehicular Broadcast (DV-CAST) protocol is presented with a combination of a suppression technique and a store-carry-forward approach to cope with both sparse and dense networks in highways. The Acknowledged Broadcast from Static to highly Mobile (ABSM) protocol [48, 49] relies on the use of Connected Dominating Sets (CDS) to perform the broadcast of messages. In [50], authors present the enhanced Message Dissemination based on Roadmaps (eMDR), a scheme that mitigates the broadcast storm disconnected networks in real urban scenarios. The UV-CAST is a protocol that specifically addresses urban scenarios with zero infrastructure support [51]. In Section 3.1, we present the SRD protocol [22]. Just as with DV-CAST, SRD combines both a store-carry-forward approach and suppression technique to tackle disconnected and dense networks, respectively. Its suppression technique, Optimized Slotted 1-Persistence, relies on an optimized version of the Slotted 1-Persistence suppression method to prevent nearly simultaneous rebroadcasts in a single time slot in dense networks.

Most related works mentioned above address either highway or urban scenarios, or sometimes only the broadcast storm problem in dense networks. Moreover, protocols designed specifically for urban scenarios usually rely on infrastructure to support the data dissemination. In Section 3.3, we present the infrastructure-less AMD protocol that scales properly from sparse to dense networks and that works seamlessly in both highway and urban scenarios. AMD combines a generalized delay-based suppression technique based on directional sectors and a store-carry-forward algorithm to support multi-directional data dissemination.

2.2 Data dissemination for non-safety applications

In contrast to safety applications, there have been fewer works related to non-safety applications in vehicular networks. Since non-safety applications comprise everything that is not safety-related, solutions are often tailored to specific applications. However, these solutions still share the challenge of having to deal with large amounts of data in an environment where there is not enough available bandwidth for all data to be broadcast. While this is not a problem for disseminating the few messages generated by not so frequent safety events, vehicles running non-safety applications are expected to collaboratively build unbounded amounts of data related to, for example, road traffic, parking, interest points, video, and so forth.

In sparse networks, the connectivity time is particularly limited due to the high speed of vehicles and can be as low as 3 seconds [52]. As we show later in Section 4.3.5, two vehicles moving in opposite direction at approximately 120 km/h with a typical 250 meters of transmission range leads to a link connectivity time of only 7.62 seconds. In practice, due to the inherent unreliability of broadcast communication, this results in a throughput of only 743.8 kbit/s when radio devices are configured with a data rate of 6 Mbit/s. Also, the average link duration time between any pair of vehicles in urban scenarios has shown to be bounded to only 20 seconds regardless of the network density [53].

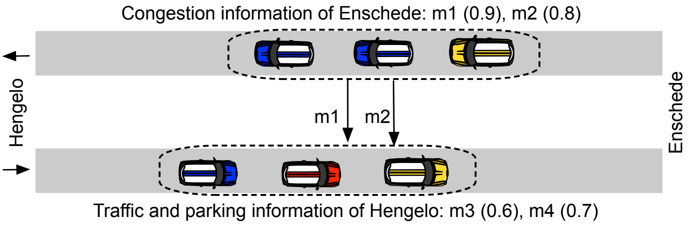
On the other hand, increasing network densities result in more vehicles sharing the bandwidth, which can further limit the amount of time for each vehicle to broadcast data. In [54], it has been observed that high network densities lead to undesirable effects such as increase in service time, decrease in reception probability, and decrease in throughput after the saturation point of the channel is reached. In addition, authors in [55] show that when a single radio is used for both safety and non-safety applications, the bandwidth is further limited due to the use of channel hopping between control and service channels.

Few works have been devoted to delay-constrained and loss-sensitive non-safety applications such as multimedia streaming [56, 57, 58, 59]. These solutions generally propose mechanisms such as network coding to increase robustness when disseminating data to a group of vehicles on the road. However, in this thesis, our focus is rather on the selection (prioritization) of data based on the data's utility to neighboring vehicles. In this line, one of the earliest works proposing the use of application utility for data selection is [60]. Authors focus on solving scalability issues when disseminating data in VANETs by selecting messages that maximize the total utility gained by all vehicles in the neighborhood. Differently, authors in [61] introduce a protocol that allows content to remain available in areas where vehicles are most interested in it. A detailed study of using utility to reduce the uncertainty of sensor data gathered by vehicles is presented in [62]. Similar to this work is [63], where authors consider the average system information age to maintain up-to-date state information among all nearby vehicles. In [64], a Peer-to-Peer (P2P) approach is introduced to address the problem of popular content distribution (PCD) in VANETs when a file is broadcast by roadside units (RSUs) to vehicles. Vehicles cooperate by exchanging data and complementing their missing packets. In [65], PrefCast is proposed. The protocol focuses on a preference-aware content dissemination that targets on maximizing the user's satisfaction in terms of content objects received. When a node meets neighboring users for a limited contact duration, it disseminates the set of objects that can bring possible future contacts a high utility. Although not explicitly defined in a general utility function, the Road Information Sharing Architecture (RISA) is presented in [66]. The architecture comprises a distributed approach to road condition detection and dissemination for vehicular networks. A Time-Decay Sequential Hypothesis Testing (TD-SHT) approach is used to combine event information from multiple sources to increase the belief of such events. Finally, [67] presents an information dissemination function to maximize the total utility across all applications while respecting communication constraints.

One key aspect missing in these works is the consideration of utility fairness when vehicles have conflicting interests. We argue that data selection mechanisms must aim at a *fair* distribution of data utility, given the possible conflicting data interests among vehicles. As exemplified in Figure 2.4, vehicles moving in opposite directions are potentially interested in each other's data, since a group of vehicles in one direction holds data related to the destination of vehicles in the opposite direction. If we consider a hypothetical situation where there is only time/available bandwidth for the exchange of two messages, a fair approach would choose messages m_1 and m_4, thereby providing a gain of 0.9 of utility to vehicles moving to Enschede and a gain of 0.7 to vehicles moving to Hengelo. In contrast, an altruistic-based approach [60] that maximizes the total utility gained by all vehicles in the neighborhood would choose m_1 and m_2, thereby leaving vehicles in one direction with no information about their destination. In Section 4.1, we elaborate on the comparison between fairness and efficiency as goal for data selection, as described in [24].

Although in [68] authors introduce the concept of application-utility-based fairness, their focus is on controlling flow rates in time-constraint data traffic. One work that takes the conflict of interests into account is [69]. However,

(a) Altruistic Approach



(b) Fair approach

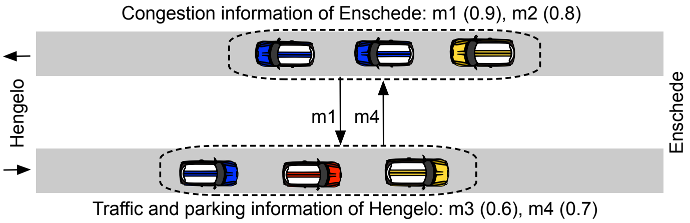


Figure 2.4: Motivation for a *fair* data selection. In (a), only vehicles heading to the city of Enschede receive information, namely, congestion information about Enschede. A fair approach in (b) leads to a more even distribution of utility, providing traffic awareness to vehicles in both road directions

the data selection considered is restricted to only pairs of vehicles. In Section 4.2, we go one step further and present a generalized and fully distributed approach for utility data selection, i.e., FairDD [25], that is suitable for broadcasting communication. Later in [70], authors present a generic framework for describing the characteristics of content exchange among nodes in Delay Tolerant Networks (DTNs). A distributed information popularity measurement is included and the pairwise interaction of nodes is modeled as a bargaining problem.

With respect to controlling the load in the radio channel, numerous works have focused on either adjusting the power level or transmission rate of messages [71, 72, 41]. However, such works focus mainly on disseminating safety *beacons* that are valid for a very short period of time to provide cooperative awareness. In this work, we are rather interested in approaches that control the network load when messages carrying application data have to be disseminated throughout the network, for longer distances and timespans. In this line, the protocol presented in [73] determines the data rate of each vehicle based on the application utility of each message in the transmission queue. Similarly, [74] proposes a method for controlling the network congestion by considering different aspects such as the message priority and vehicles' speeds. Different forms of data aggregation have also been used to improve the quality of information exchanged and reduce the network load inserted into the network. Among works following this approach is the Self-Organizing Traffic Information System (SOTIS) [75]. It stores information in the form of annotated maps of different resolutions and performs information exchange through a specialized MAC protocol. Instead of relying on an ad-hoc network, the PeerTIS [76] builds a peer-to-peer overlay over the Internet by means of cellular network to provide data about the current road traffic conditions.

One major drawback of these solutions is that they either focus on message utility or network load control in order to address scalability issues of data dissemination in VANETs. To the best of our knowledge, the Adaptive Traffic Beaconing (ATB) [17, 77] pioneered an approach that combines both aspects into one adaptive transmission rate control. However, just as with other approaches that define the message utility, it lacks the consideration of utility fairness when vehicles have conflicting interests. In Section 4.3, we extend and improve ATB to achieve data utility fairness in the neighborhood. Although aggregation mechanisms certainly help in reducing the network load [78], they also involve making trade-offs between data amount and information quality (completeness). In this thesis, we argue that even with aggregation mechanisms, vehicles will still need to make decisions with regard to selecting the data to broadcast depending on the vehicles' interests. This is precisely what we explore in Chapter 4.

2.3 Concluding remarks

In this chapter, we have reviewed state-of-the-art solutions designed for disseminating data of safety and non-safety applications. While there is considerable amount of work done in the field of safety applications, such works lack in proposing a solution that cope with both highway and urban scenarios. Throughout Chapter 3, we elaborate on store-carry-forward and suppression techniques solutions to fill this gap. In the other side of the spectrum, very few works related to maximizing data utility gain in the neighborhood have taken into account the potential conflict of data interests that vehicles may have depending on their context. Furthermore, current solutions also lack in simultaneously considering both network load control and data utility gains. We address both issues with a single solution developed throughout Chapter 4.

CHAPTER 3

Data dissemination for safety applications

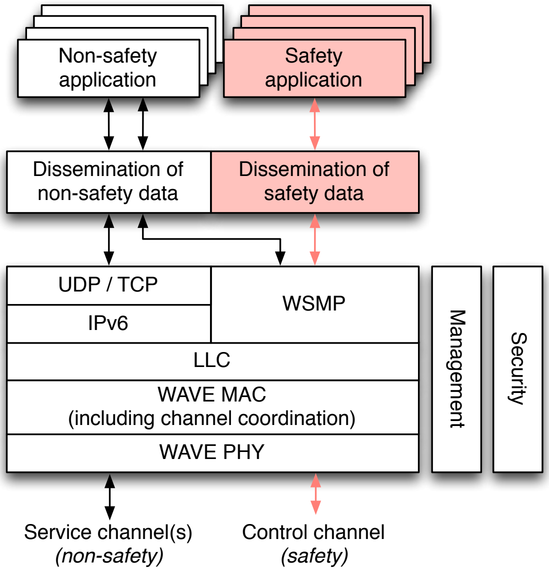


Figure 3.1: The WAVE stack with highlighted module for safety applications

In this chapter[1], we present solutions for the dissemination of data related to safety applications. Our goal is to rely on the minimum number of vehicles to deliver event-driven messages as quickly as possible to all vehicles within

---

[1] This chapter is based on the following publications: (i) *A Simple and Robust Dissemination Protocol for VANETs*, 16th European Wireless Conference 2010 [21]; (ii) *A directional data dissemination protocol for vehicular environments*, Elsevier Computer communications 34 (17) 2011 [22]; (iii) *Exploiting beacons for scalable broadcast data dissemination in VANETs*, Proceedings of the 9th ACM international workshop on Vehicular inter-networking, systems, and applications (VANET) 2012 [23]; and (iv) *A Scalable Data Dissemination Protocol for Both Highway and Urban Vehicular Environments*, Springer EURASIP Journal on Wireless Communications and Networking (accepted for publication).

the region affected by the event. Figure 3.1 highlights the components used for safety-related data dissemination as defined by our approach described in Chapter 1. A module placed between safety applications and the WAVE protocol stack takes care of coordinating the messages among neighboring vehicles. Throughout the chapter, we consider one of the possible radio set-ups where one transceiver is dedicated to the control channel and another is used to handle one or multiple service channels. This allows us to study the multi-hop dissemination of safety messages under optimal conditions, i.e., without loss in performance due channel hopping when a single transceiver is employed. In addition, we assume that every vehicle is able to determine its current geographical position on the road using, for example, the Global Positioning System (GPS). Finally, evaluation parameters such as transmission range and scenario size are adjusted throughout the sections according to their suitability to meet scalability requirements in terms of simulation execution time.

The remainder of the chapter is organized as follows. In Section 3.1, we present a data dissemination protocol that copes with disconnected networks in highway scenarios while also preventing the broadcast storm problem in dense networks. Section 3.2 elaborates on the broadcast storm problem in dense highways by presenting a suppression technique that tackles scalability issues in terms of number of transmissions when increasing network densities are considered. In Section 3.3, we adapt and extend concepts introduced in the previous sections for the case of multi-directional dissemination, thereby tackling scalability issues in both highway and urban scenarios. Finally, Section 3.4 finalizes this chapter with concluding remarks.

3.1 Dealing with disconnected networks in highways

3.1.1 Introduction

For many safety applications, the data acquired by sensors, e.g., crash detection data, must be broadcast (disseminated) to all vehicles nearby. Because these events might not directly affect all vehicles within the event perimeter, broadcast messages can be propagated towards a specific direction such as to vehicles that are in fact approaching the dangerous area. In this section, we consider the problem of coordinating these broadcast messages to a specific direction in a reliably, timely, and efficiently manner using vehicle-to-vehicle communication. We present a dissemination protocol which assumes no information available about the road topology. Therefore, we focus here on highway scenarios, where simple long bidirectional roads are present.

The contribution described in this section lies in combining an optimized broadcast suppression technique with a store-carry-forward model into a single dissemination protocol called the Simple and Robust Dissemination (SRD) protocol. We argue that SRD is simple because there are only two protocol states that a vehicle can operate: either as the cluster tail or as a non-tail vehicle. This comes from the fact that protocols such as DV-CAST [47] have increased complexity due to additional required rules, e.g., whether the vehicle is the intended recipient of the message or whether the vehicle is in the opposite direction of the road. Furthermore, we argue that SRD is robust, since it can cope with a highly dynamic environment where vehicles may suddenly leave the road. We show throughout this section that SRD operates seamlessly in both dense and sparse networks and outperforms other state-of-the-art protocols that share a similar goal.

3.1.2 Simple and robust dissemination

The Simple and Robust Dissemination (SRD) protocol aims to efficiently disseminate data in both dense and sparse vehicular networks. More specifically, it aims to achieve a high delivery ratio with a low propagation delay and yet without introducing an excessive load in the network. For this purpose, we take the following approach:

- In sparse networks, disconnections are predominant and, therefore, measures must be taken to guarantee that a message can still travel to its direction after the connectivity is reestablished. With this goal, the *store-carry-forward* communication model is employed.

- In contrast, in dense networks, the number of vehicles might be excessively high and, therefore, we must deal with the *broadcast storm problem*. To this end, we rely on an *optimized broadcast suppression technique* to always relay messages using a low number of vehicles.

3.1.2.1 Concept definitions

To better understand the protocol, we define the following concepts which are used throughout the remaining sections:

Definition 1 (Message Direction). Given a multiple-lane road, where vehicles move in both road directions, the message direction d is defined by the application responsible for generating the message and it is one of the two road directions. We assume that this application can be running in either a vehicle or in a road-side unit (RSU), e.g., to broadcast a vehicle crash or an upcoming danger area. For the sake of simplicity, we refer to each direction as the easterly and westerly directions.

Definition 2 (Vehicle Cluster). Given a multiple-lane road, where vehicles move in both easterly and westerly directions, a vehicle cluster vc is defined as a group of vehicles moving in any direction with multi-hop radio connectivity at a time instant.

Definition 3 (Cluster Front). Given a vehicle cluster vc and a message direction d, the cluster front *cf* is defined as the vehicle within cluster vc with no radio connectivity with other vehicles positioned farther in the direction opposite to d. For instance, for an easterly message direction the front vehicle would be the farthest vehicle in the westerly direction within vc.

Definition 4 (Cluster Tail). Given a vehicle cluster vc and a message direction d, the cluster tail ct is defined as the vehicle within cluster vc with no radio connectivity with other vehicles positioned farther in message direction d, i.e., the final vehicle belonging to vc in the message direction.

Definition 5 (Radio Gap). Given two vehicle clusters vc_1 and vc_2, message direction d, and cluster tail ct_1 of cluster vc_1, the gap is defined as:

$$Gap = D(vc_1, vc_2) - CR(ct_1), \tag{3.1}$$

where $D(vc_1, vc_2)$ is the distance between clusters vc_1 and vc_2, i.e., the distance between ct_1 and the cluster front of vc_2 in the message direction d. $CR(ct_1)$ is denoted as the communication range of vehicle ct_1.

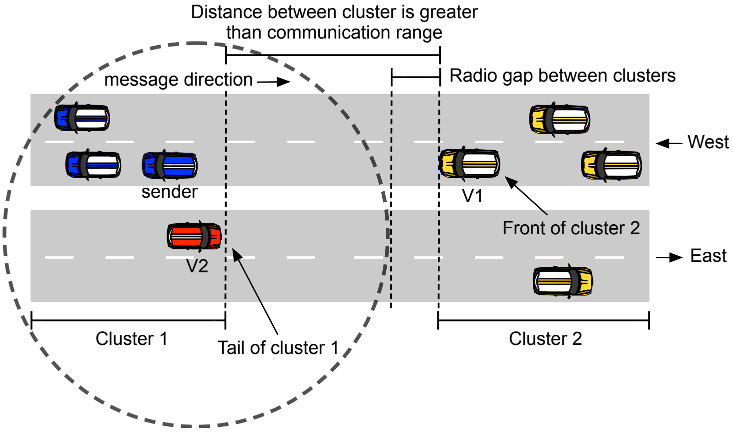


Figure 3.2: Protocol concepts applied in a simple example

Figure 3.2 shows an example of how these concepts are applied. A road with multiple lanes contains vehicles moving in both westerly and easterly directions. Two groups of vehicles are separated by a radio gap and are classified as vehicle clusters vc_1 and vc_2. Let us suppose that a message is generated by the sender towards the easterly direction. With respect to this direction, vehicle v_2 is defined as the tail of vc_1 and v_1 as the front vehicle of vc_2.

3.1.2.2 Optimized Slotted 1-Persistence

The suppression technique that we propose is based on the Slotted 1-Persistence presented in [32]; however, with a slightly altered formula to guarantee an equal distribution of vehicles among the time slots assigned[2]. Therefore, vehicles are assigned to different time slots depending on their distance to the sender, where vehicles with highest priority are given the shortest delay before

---

[2] A typographical error with regard to the ceiling math function position has been identified in the formula for the Slotted 1-Persistence technique proposed in [32], which leads to inaccurate distribution of vehicles among different time slots.

rebroadcasting. The differences lie in two optimizations designed to decrease the number of collisions during a rebroadcast and improve the overall delivery ratio: (i) assignment of different time slot priorities for each road direction; and (ii) the introduction of an additional delay within each time slot to cope with the synchronization problem found in [37, 36]. We refer to our suppression mechanism as *Optimized Slotted 1-Persistence*.

The time slot assignment is defined as follows. If a vehicle j farther in the message direction receives a new message from vehicle i, it schedules a rebroadcast for that message; otherwise it ignores it and if the sender is farther in message direction it suppresses any previously scheduled rebroadcast for that message.

When scheduling a message, vehicle j first calculates the percentage distance PD_{ij} between the two vehicles with respect to the estimated transmission range R.

$$PD_{ij} = \left\lceil \frac{min(D_{ij}, R)}{R} \right\rceil, \tag{3.2}$$

where D_{ij} is the distance between vehicles i and j. As a result, the PD_{ij} value will vary within the interval (0,1] with large distances being closer to 1. The *minimum* function is necessary, since the transmission range R is an estimate based on the power level employed and vehicles in reality could be positioned at farther positions.

The time slot number S_{ij} assigned to vehicle j is then defined by the following equation:

$$S_{ij} = \lfloor NS \times (1 - PD_{ij}) \rfloor, \tag{3.3}$$

where NS is the total number of time slots utilized. If vehicles are uniformly distributed within the transmission range of vehicle i, they will be equally distributed among the NS time slots reserved. S_{ij} will vary within the interval [0, $NS - 1$].

In most vehicular applications, a message should be forwarded only in one direction and the intended receivers are vehicles moving in one particular road direction. Therefore, the first (i) optimization we propose is to give a higher priority to one road direction and assign later time slots to vehicles in the opposite direction. The objective of this modification is to have fewer vehicles assigned to each time slot thereby reducing the number of rebroadcasts and collisions.

We update the assignment of time slots S_{ij} to include this modification as follows:

$$S_{ij} = \begin{cases} \lfloor NS \times (1 - PD_{ij}) \rfloor & \textbf{if } v_{dir} = hp_{dir}; \\ \lfloor NS \times (2 - PD_{ij}) \rfloor & \textbf{if } v_{dir} \neq hp_{dir} . \end{cases} \tag{3.4}$$

The v_{dir} and hp_{dir} values are the vehicle direction and high priority direction, respectively. In this way, the time slot range is equally divided in [0, $NS - 1$] for the high priority direction and [NS, $(2 \times NS) - 1$] for the opposite (low priority) direction.

Finally, the time that vehicles have to wait before rebroadcasting at time slot S_{ij} is calculated by equation 3.5:

$$T_{Sij} = S_{ij} \times st, \tag{3.5}$$

where the slot time *st* is a value larger than the one-hop delay that includes the medium access delay, transmission delay and propagation delay.

Assigning vehicles to different time slots clearly breaks the synchronization present in the simple flooding approach, where all nodes would rebroadcast simultaneously upon the receipt of a message. The slot time *st* is defined in such a manner that it gives vehicles assigned to later time slots the opportunity to cancel their transmissions, since the message has already been rebroadcast. Therefore, ideally only vehicles assigned to the earliest time slot would rebroadcast. However, a similar synchronization on a smaller scale can still occur when multiple vehicles are assigned to a single time slot and start their transmission simultaneously. Such a synchronization problem has been identified in [37]. To cope with this problem, in that work a variation of the slotted 1-Persistence technique called microSlotted 1-Persistence Flooding has been proposed. The proposed scheme functions in the same way as the Slotted 1-Persistence Broadcasting scheme but with a small additional delay, i.e., the *micro slots*, within each time slot to break the defined synchronization. The same problem has been identified and referred to as the *Timeslot Boundary Synchronization Problem* in [36]. Differently, such a work describes design guidelines for extra measures to be taken not only in the network layer but also in the link layer by inserting a pseudo-random delay to SIFS in the IEEE 802.11p MAC layer. Especially in congested networks, an additional delay introduced uniquely in the network layer does not suffice when nodes experience high contention in the link layer, as their timeslots could be again aligned.

As in [36], we support the position that the synchronization must be broken in both the network and link layers to be completely effective. However, in order to comply with the current MAC and PHY layers of the 802.11p standard, we propose the use of a small extra delay but only in networks layer and maintain the MAC layer unaltered. We follow the guidelines proposed in [36] that suggest that this extra delay must be chosen from a near continuous interval in order to completely break the alignment of time slot boundaries. The additional delay AD_{ij} is our second (ii) optimization and is defined as follows:

$$AD_{ij} = \begin{cases} D_{max} \times (1 - PD_{ij}) & \textbf{if } v_{dir} = hp_{dir}; \\ D_{max} \times (2 - PD_{ij}) & \textbf{if } v_{dir} \neq hp_{dir}. \end{cases} \tag{3.6}$$

In the equation above, D_{max} is the maximum delay allowed. Following the idea adopted for the assignment of time slots, vehicles driving in the high priority direction receive smaller delay values than vehicles driving in the opposite (low priority) direction.

The time that vehicles have to wait before rebroadcasting is updated to include the additional delay described and is expressed as:

$$T_{Sij} = (S_{ij} \times st) + AD_{ij}. \tag{3.7}$$

The result of equation 3.7 is that for each road direction each time slot is stretched with an equal fraction of D_{max}. Moreover, the beginning of each time slot is shifted by the accumulated additional time of earlier time slots, thereby preserving the pre-defined st value and preventing overlapping between different time slots.

The complete suppression mechanism is shown in Figure 3.3. In addition to the time slot assignment, the rule for canceling a rebroadcast also differs from the original Slotted 1-Persistence. In [32], a suppression of a rebroadcast occurs whenever a duplicate of a scheduled message is received before its transmission time. Because the most distant vehicles in the message direction will rebroadcast first, there is no harm in unconditionally canceling rebroadcast when receiving an echo. However, due to our separation of priorities for each direction, the most distant vehicles can be positioned in the low priority direction and thus assigned to late time slots. As a consequence, these vehicles might cancel their rebroadcasts erroneously upon the receipt of an echo coming from vehicles closer to the sender (in the high priority direction) and impede a further propagation of the message. To prevent this behavior, we define that

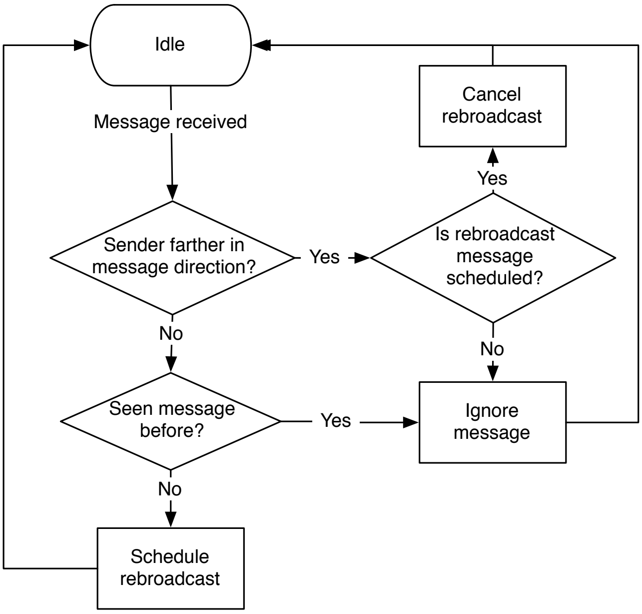


Figure 3.3: The Optimized Slotted 1-Persistence technique

vehicles can only cancel their rebroadcasts when receiving an echo from other vehicles farther in the message direction. To guarantee that we only consider new messages, we keep track of the most recent message IDs in a list, namely, the last m message IDs received. New messages are managed by the *Schedule Rebroadcast* function which calculates the proper waiting time T_{Sij} and places the message in a sending queue. Accordingly, the *Cancel Rebroadcast* function removes the message corresponding to the echo from the queue.

3.1.2.3 The protocol

The SRD protocol decision tree diagram is shown in Figure 3.4. In the tail state, a vehicle *stores* all broadcasts received and rebroadcasts them with the flag *FromTail* set to true. The tail is responsible for *carrying* these messages until the connectivity in the message direction is established. The tail then *forwards* its stored messages, in this way concluding the *store-carry-forward* mechanism.

Vehicles in the non-tail state have two responsibilities: (i) store all messages sent by the tail (with the *FromTail* flag set to true). This is especially important for improving the protocol robustness as we show later on; (ii) rebroadcast received messages using the Optimized Slotted 1-Persistence technique to reduce redundant retransmissions.

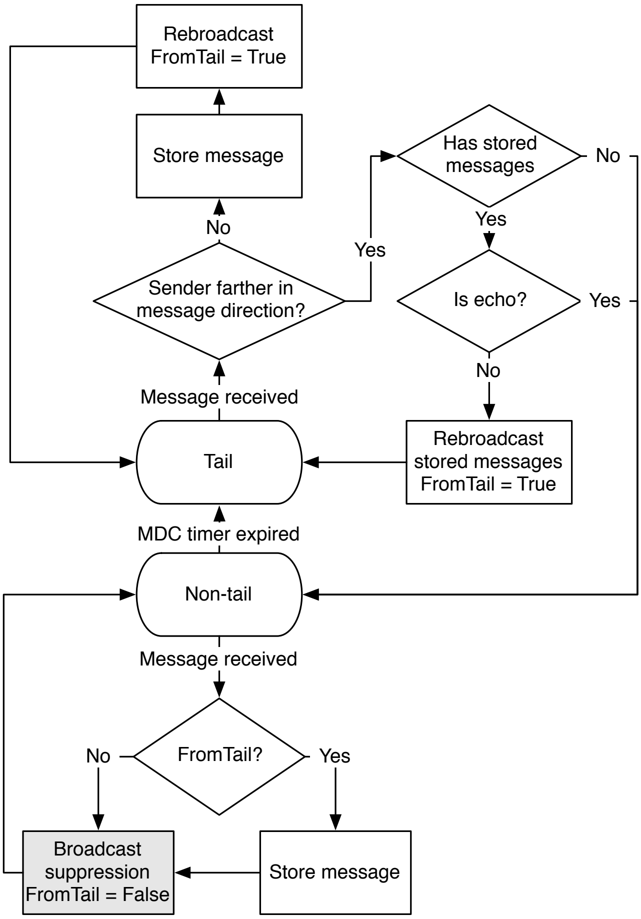


Figure 3.4: SRD protocol decision tree

Transitions between the states occur as follows. A vehicle goes from the non-tail to the tail state when it goes for longer than the Message Direction Connectivity (MDC) timer value without receiving a message retransmission from a vehicle farther in the message direction. The MDC timer duration time is defined in such a way that it considers the maximum possible time for a rebroadcast to be performed by other vehicles farther in the message direction, i.e., it must take into account message collisions, the *exponential backoff* mechanism, and the broadcast suppression technique used. The transition from tail to non-tail is triggered by the reception of a message from a vehicle farther in the message direction, as this is an indication that the cluster tail established connection with another cluster. Normally the tail has some messages stored that it needs to forward to this new cluster, so it rebroadcasts them. The vehicles from the new cluster will follow the protocol and rebroadcast these messages. Upon the receipt of an echo (i.e., a rebroadcast from a vehicle farther in the message direction) from at least one of the messages, the tail makes the transition. While the tail does not receive an echo it assumes that no vehicle in the new cluster received the messages and, therefore, the transition is not made. Once a new message from a sender farther in the message direction is received, the tail will retry this procedure. This is done to increase the protocol robustness. Note that if the tail does not have stored messages, it simply makes the transition to non-tail state as soon as a message from a vehicle farther in the message direction is received.

In order to increase robustness, every time the tail receives a message it not only stores the message but also retransmits the message with the *FromTail* set to true. By doing so, all vehicles in the range of the tail will also have a copy of that message. If the tail fails or turns off the road, eventually another vehicle will become the new tail. Since such a vehicle would already have a copy of all messages received from the old tail, it can rebroadcast them whenever the MDC is reestablished. Message delivery thus is not dependent on a single vehicle. In the example shown in Figure 3.5, the tail v_3 turns off the road, causing v_1 or v_2 to make the transition to the tail state. As they both have copies of v_3's messages, whichever one makes the transition will be able to retransmit the messages once the MDC is re-established. Robustness is then referred to as the ability of the protocol to cope with highly dynamic mobility environments.

One important remark regarding broadcasting efficiency is that since the rebroadcast from the tail is always required, the tail has a higher priority in the broadcast suppression technique, in order to avoid redundant retransmissions from non-tail vehicles within that region. This priority is implemented by as-

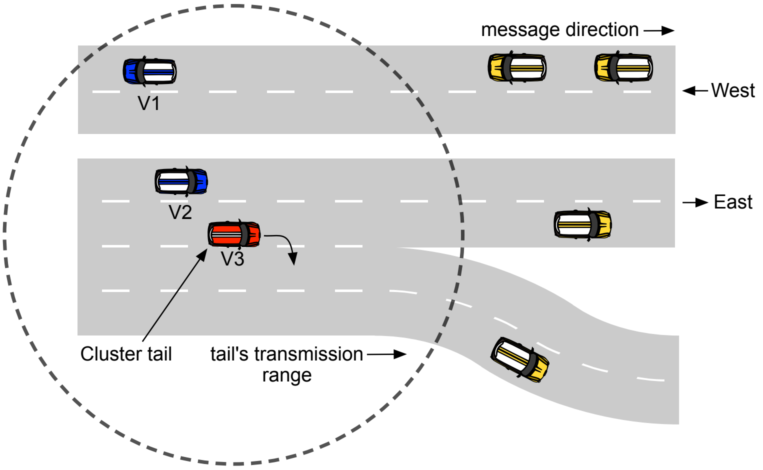


Figure 3.5: Robustness motivation

signing cluster tails to the first time slot in the Optimized Slotted 1-Persistence technique and with a smaller additional delay when compared with other non-tail vehicles also assigned to first time slot.

3.1.2.4 Defining cluster front and tail vehicles

Existing protocols such as DV-CAST require the complete knowledge of the local (1-hop neighborhood) network topology. This knowledge is usually acquired by means of *hello* messages sent by *every* vehicle periodically. These messages coexist with *event-driven* messages which are triggered upon an event such as the detection of a hazard, e.g., hard braking of cars in front. Notably, *hello* messages introduce undesirable side-effects such as increase of the network load, contention and collisions when not properly coordinated. All these effects together may harm the correct delivery of event-driven messages which are of primary importance in vehicular environments.

Unlike these protocols, SRD requires only the knowledge of whether the vehicle is currently the front, tail, or simply a relay vehicle in the cluster. Although the conventional *hello* message mechanism described above suffices to gather such information, we propose the use of a *suppressed* periodic *hello* message mechanism. In essence, such a mechanism relies on the optimized suppression technique described in Section 3.1.2.2 to broadcast *hello* messages by executing the following rules:

- On a highway, assuming a pre-defined fixed message direction d, *hello* messages are generated by the cluster front and broadcast with periodicity λ to vehicles farther in the message direction. For the sake of explanation, let us assume d to be the easterly direction.
- Upon the receipt of a new *hello* message, the SRD protocol is run (as described in Figure 3.4) and the message is rebroadcast with the optimized suppression technique accordingly. In order to further decrease the number of *hello* messages transmitted, the following modifications are introduced into the SRD protocol when dealing with *hello* messages:
  - *hello* messages are never stored as they are simply meant to gather topology information.
  - If an event-driven message is scheduled to be rebroadcast in the suppression mechanism, any previously scheduled *hello* message is canceled.
- A vehicle is said to be the cluster *front* regarding the easterly direction if it does not receive a *hello* or event-driven message for a period longer than λ originated by a vehicle farther in the opposite (westerly) direction.
- A vehicle is said to be the cluster *tail* regarding the easterly direction if it does not receive a *hello* or event-driven message for a period longer than μ (set by the MDC timer) originated by a vehicle farther in the same (easterly) direction.
- The cluster front and tail vehicles of the opposite (westerly) direction are simply the cluster tail and front vehicle defined for the easterly direction, respectively.

This approach brings several advantages over typical *hello* message mechanisms: (i) the number of messages introduced in the network is reduced; (ii) all messages within the cluster are rebroadcast in a synchronized manner by following the optimized suppression mechanism, thereby reducing the chance of collisions; and finally, (iii) *suppressed hello* messages coexist more efficiently with event-driven messages, since the former are canceled upon receiving event-driven messages.

We argue that this is one possible mechanism to gather the required topology information; any other method to identify the front and tail vehicles in a cluster can be used. For instance, in [79] a protocol designed to collaboratively build an overview of the upcoming traffic in highways is presented. The front vehicle starts building a traffic map and vehicles behind it aggregate data whenever requested up to the cluster tail. Such type of mechanisms can replace our suppression *hello* message mechanism and still provide the required information.

3.1.2.5 Message structure

Both event-driven and *hello* messages have vehicle and message IDs to enable vehicles to distinguish different broadcasts and to identify rebroadcasts. An example of vehicle ID is the MAC address, while for the message ID can either be a sequence number or a timestamp of the message generation time. The timestamp is in either case necessary in order to set an expiration time for each message and prevent old messages from being disseminated. Moreover, depending on the application-dependent size limit n for the list of stored messages, the timestamp can be used to remove the oldest messages when the limit is reached. In addition to time, the expiration mechanism can also be based on distance to prevent receiving irrelevant messages originated hundreds of kilometers away. A message could be considered expired when, for example, it reaches the end of a highway or simply after it reaches vehicles more than 10 km away from the event. For the optimized suppression technique, both message direction and geographical position of the sender are required. The former indicates to which direction the message must be propagated while the latter allows vehicles to calculate their distance with respect to the sender and choose a time slot accordingly. Finally, the *FromTail* flag utilized by SRD is included to allow vehicles surrounding the tail to store event-driven messages.

The message header structure is therefore defined by the following values: *[Vehicle ID, Message ID, timeStamp, Distance Propagated, Message Direction, Sender Coordinates, FromTail Flag].*

3.1.2.6 Basic operation example

The basic operation of the SRD protocol is shown in Figure 3.6. In Figure 3.6(a), a message is broadcast towards the easterly direction by the sender. All receiving vehicles except for the tail simply rebroadcast the message generated using the proposed broadcast suppression technique. When a non-tail vehicle receives a message from another non-tail vehicle that is farther in the message direction, it simply drops the message and cancels (suppresses) any previously scheduled transmission in case the message received is an echo. All vehicles help disseminating the message, regardless of the direction they are moving. Whenever the broadcast message reaches the tail (v_1 in Figure 3.6(a)), the cluster tail stores the message and rebroadcasts it with the *FromTail* flag set to true. In this way, all non-tail vehicles that hear the rebroadcast from the cluster tail also store the message.

A change in the cluster tail is shown in Figure 3.6(b), in which, after some period of time, v_2 listens to a rebroadcast from the tail v_1. Even though v_2 realizes that the sender was not farther in the message direction, the message is stored as it comes with the *FromTail* flag set to *True*. Following the protocol, v_2 rebroadcasts it using the broadcast suppression technique. This rebroadcast is needed since v_2 does not know yet that it is the new tail. v_1 then receives this retransmission and verifies that v_2 is farther in the message direction. Consequently, v_1 retransmits all stored messages and starts the transition procedure to the non-tail state. This retransmission is done to cover two possibilities. First, there could be a gap after v_2 farther in the message direction and v_2 would become the new cluster tail (as shown in Figure 3.6(b)). In this case, the rebroadcast is done to guarantee that the new tail has a copy of all messages from the old tail (v_1). In the second case (not shown in the figure), the gap does not exist, i.e. there is a vehicle in the range of the v_2 that is not in the range of v_1. The retransmission in this case will cause v_2 to relay all messages to this farther vehicle and consequently to all others that it might be connected to. In either case, upon receiving the rebroadcasts from v_2, v_1 concludes the transition to non-tail state.

As v_2 is moving farther in the message direction, at some point it enters in the communication range of v_3, thereby reaching a new cluster, as shown in Figure 3.6(c). When this happens, v_2 eventually receives a message from v_3. As v_3 is farther in the message direction, v_2 starts the transition from tail to non-tail state by rebroadcasting every stored message it carries. At this point, v_3 and all non-tail vehicles within its cluster will rebroadcast the messages received in order to spread them to other vehicles farther in the message direction. When v_2 receives one of these rebroadcasts, it concludes the transition to non-tail state.

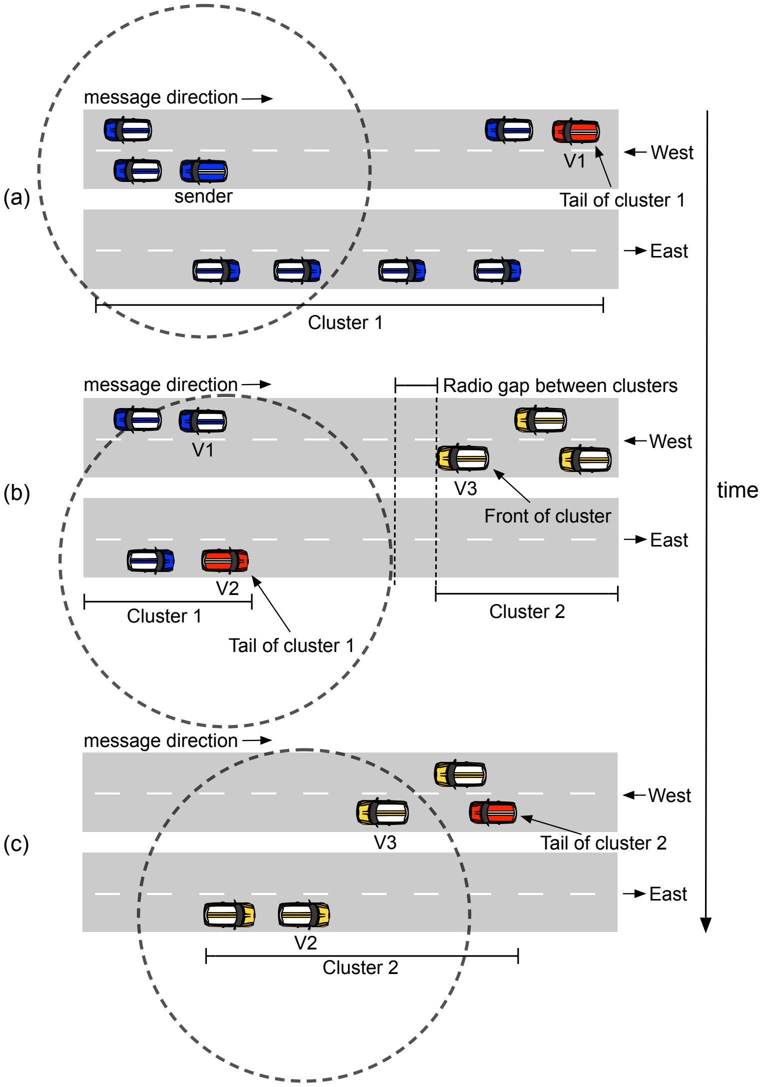


Figure 3.6: Protocol description

Table 3.1: Simulation parameters

Physical layer	Frequency band	5.88 GHz
	Bandwidth	10 MHz
	Transmission range	∼176 m
	FSPL exponent α	3.5
	Receiver sensitivity	-119.5 dBm
	Thermal noise	-110 dBm
Link layer	Bit rate	6 Mbit/s
	CW	[15,1023]
	Slot time	13 μs
	SIFS	32 μs
	DIFS	58 μs
Suppression mechanisms	st	5 ms
	NS_{std}	3
	NS_{opt}	6
	D_{max}	1 ms
	MDC	0.1 s
	Hello message size	24 Bytes
	Hello frequency	1 Hz
Scenarios	Data message size	2312 bytes
	Message frequency	0.5 Hz
	# Runs	50

3.1.3 Performance evaluation

In the following, we present the performance evaluation of the SRD protocol carried out by means of simulations with OMNeT++ 4.1 [80]. We consider four protocol versions: SRD and DV-CAST; and their respective suppression techniques used for dense scenarios, namely, Optimized Slotted 1-Persistence and Slotted 1-Persistence. Our goal is to evaluate SRD under various vehicle scenarios and compare it directly with DV-CAST, which is the existing protocol that also focus on directional broadcasting in both dense and sparse highways. The separate evaluation of the suppression techniques serves to assess the actual gain of the store-carry-forward models employed by both SRD and DV-CAST in sparse networks.

In our simulations, we utilize the MiXiM Framework [81] and adjust the available implementation of the IEEE 802.11b protocol to comply with basic specifications of the 802.11p version. In the MAC layer, we set the bit rate to 6 Mbit/s, the Contention Window (CW) to values between 15 and 1023, the slot time to 13 μs, the SIFS to 32 μs, and the DIFS to 58 μs. In the physical layer, we operate on the 5.88 GHz frequency band, with 10 MHz of bandwidth.

With regard to the transmission power employed, different values may be used according to the application's priority. Efforts put on selecting a proper transmission power value include the Decentralized Congestion Control (DCC) mechanism as defined by the ETSI European standardization [12] that controls the network load by adjusting the transmission power level and transmission rate. However, our goal here is limited to achieving a proper balance between choosing realistic values (i.e., up to 500 meters of range) and achieving scalability in the simulations in terms of the overall processing time. Here, we are interested in guaranteeing that multi-hop communication is used in our simulation scenarios. Despite leading to higher delay, lower transmission ranges are clearly more suitable to meet this goal. They are also preferred in terms of scalability, since fewer vehicles would be sharing the medium in the simulation simultaneously. This explains the different power values used throughout this chapter. Later in Section 3.2.3.2, we evaluate the effects of employing different power levels for different suppression techniques.

In this section, we set the transmission power to 50 mW to achieve approximately 350 meters of interference range and 176 of transmission range, assuming the Friis Free Space Path Loss (FSPL) propagation model with exponent α equal to 3.5.

For the suppression mechanisms, we set st to 5 ms. We define the total number of time slots for the Slotted 1-Persistence protocol (NS_{std}) to 3 and for the Optimized Slotted 1-Persistence protocol (NS_{opt}) to 6 (3 slots for each road direction), except in Section 3.1.3.4 where the protocols are evaluated for different values of NS. The value set for the Slotted 1-Persistence protocol is based on [47] while the value set for the Optimized Slotted 1-Persistence protocol is based on preliminary simulation studies. For the maximum additional delay D_{max}, we use 1 ms. The MDC timer defined in the SRD protocol is set to expire after 0.1 s. For the SRD and DV-CAST protocols we set the size of *hello* messages to 24 bytes and they are generated with 1 Hz frequency. Also for both protocols, for the sake of simplification we do not set size limits for the lists which keep track of the most recent messages IDs and which store the messages in the store-carry-forward mechanisms.

For all simulation scenarios the message size is the same, 2312 bytes, the maximum allowed by the 802.11p standard. New messages are generated every 2 seconds, i.e., the message frequency is 0.5 Hz. Each message is generated by the farthest vehicle in one end of the road. For each simulation scenario 50 runs are executed. Table 3.1 contains a summary of the simulation parameters common to all simulation scenarios.

Our evaluation considers the following metrics:

- ***Delivery ratio***: the percentage of messages generated by the farthest vehicle in one end of the road which fully propagate and are received by a vehicle in the extreme opposite end of the road. Ideally, dissemination protocols must achieve a delivery ratio percentage close to 100%.
- ***Delay***: the total time taken for a message to propagate from one end to the other of the road length. This is particularly important for critical information that must be disseminated as rapidly as possible.
- ***Total number of transmissions***: the total number of transmissions performed on average by an arbitrary vehicle, including both *hello* and data messages. This value is normalized by the total number of vehicles in each scenario. In order to be scalable, protocols must keep a low number of transmissions during a message's dissemination.

3.1.3.1 Controlled scenarios

We start our simulation campaign by studying the performance of the protocols for various controlled scenarios. By *controlled* we mean that we build the vehicle distribution along the road in such a way that it allows us to understand with precision what to expect from the protocols. This is important since we want to test the protocols in specific road conditions which are hard to reproduce in most traffic simulators. In particular, we guarantee for these scenarios that the maximum theoretical delivery ratio is 100%. This does not occur for traffic simulator scenarios as we report in Section 3.1.3.2. First, static scenarios with equally spaced vehicles are used to evaluate the scalability of the protocols in highly congested roads with increasing densities. In this way, we simulate well connected networks in intense traffic jams where vehicles would practically not move. Following, we concentrate on evaluating the protocols in basic mobility scenarios where protocols must deal with both well connected and sparse networks. Here, we focus in particular on cases where store-carry-forward mechanisms should overcome gaps between vehicle clusters.

In this first set of scenarios, we study the performance of the protocols for various traffic densities. To allow that, we simulate a two kilometer road with

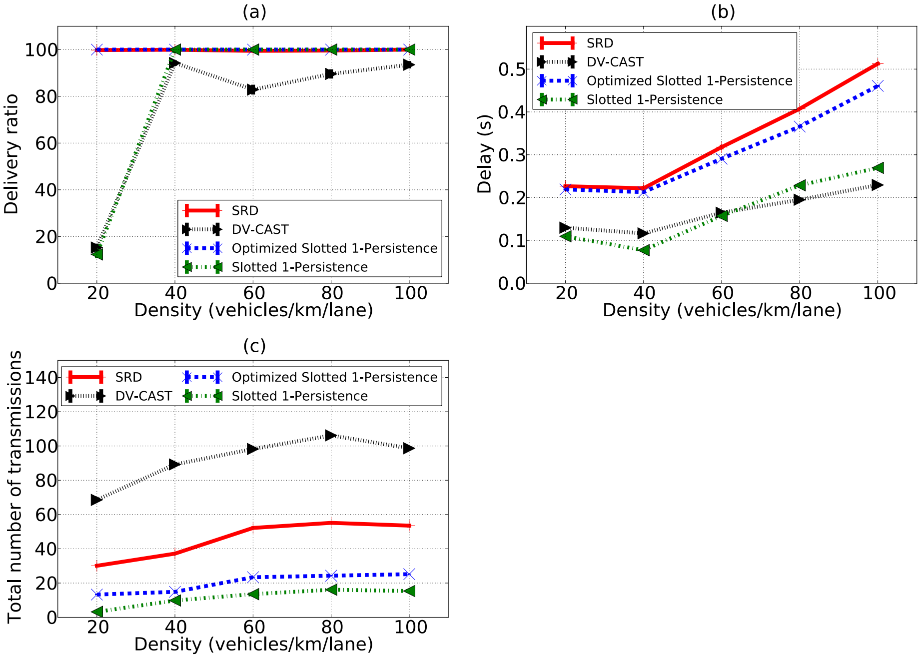


Figure 3.7: Results with 95% confidence intervals for increasing network densities

static vehicles placed in both easterly and westerly directions, with each direction comprising two lanes. We consider congested scenarios where vehicles are equally spaced in such a way that there is no radio gap between them. We vary the number of vehicles from 20 to 100 vehicles/km/lane. Each simulation run has a duration time of 60 seconds.

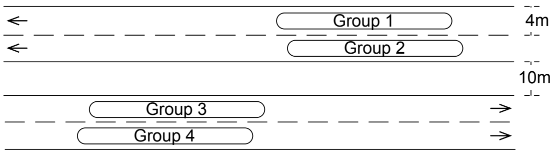
In terms of network load, with an increase in density the number of transmissions performed on average is generally increased, as shown in Figure 3.7(c). This is expected as more vehicles are assigned to individual time slots and thus rebroadcasting. The results show that almost the double number of transmissions is performed by DV-CAST when comparing with SRD. This is explained by the higher number of periodic *hello* messages transmitted. On the other hand, when comparing their respective strategies employed to cope with dense scenarios, Optimized Slotted 1-Persistence presents a slightly higher number of transmissions when compared with Slotted 1-Persistence. Although Optimized Slotted 1-Persistence relies on the double number of time slots to decrease the number of vehicles rebroadcasting, the policy adopted to cancel rebroadcasts is more strict when compared with Slotted 1-Persistence as it only allows a transmission to be suppressed when an echo is received by other vehicles farther in the message direction. For instance, depending on the vehicle distribution, some vehicles positioned in the low priority direction will not cancel their rebroadcasts if they hear earlier from other vehicles in the high priority direction which are not farther in the message direction.

Although leading to a higher number of transmissions, our policy to suppress transmissions achieves higher delivery ratio values, as shown in Figure 3.7(a). The delivery ratio achieved with Optimized Slotted 1-Persistence is near 100% for all densities. Such performance is also valid for SRD. The other protocols, namely, DV-CAST and Slotted 1-Persistence present a low delivery ratio in low densities. Because of the lack of mechanisms to cope with simultaneous broadcasts, collisions become severe for these protocols specially in low densities. However, as the density increases their performance generally improves. This can be explained by the higher probability that at least one rebroadcast is successful when more vehicles are present in each time slot. When higher densities are considered, the delivery ratio of DV-CAST varies from 80% to 95% whereas Slotted 1-Persistence delivers nearly 100% of the messages. This can be reasoned by the higher and asynchronous number of messages transmitted by DV-CAST due to the use of *hello* messages, which leads to a higher number of collisions.

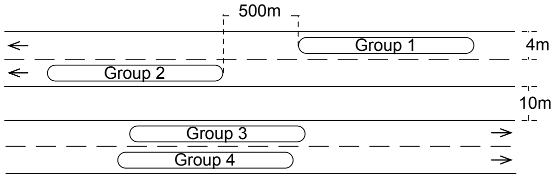
Figure 3.7(b) shows the performance with respect to the end-to-end delay. The delay is higher for all protocols when the density increases. This is due to the higher number of vehicles transmitting near simultaneously in an individual time slot, which leads to a higher number of collisions and contention period in the MAC protocol layer. As expected, the delay is higher with SRD and Optimized Slotted 1-Persistence, since some rebroadcasts are performed by vehicles in the low priority direction in later time slots.

Overall, SRD and Optimized Slotted 1-Persistence outperform DV-CAST and Slotted 1-Persistence with regard to delivery ratio, which is the ultimate goal of dissemination protocols. In order to achieve such performance, we decrease the number of vehicles transmitting in an individual time slot by relying on a higher number of time slots. The trade-off is the higher end-to-end delay in comparison with the other protocols. Finally, the *suppressed* periodic *hello* message mechanism employed by SRD helps to decrease by half the number of transmissions in comparison with DV-CAST.

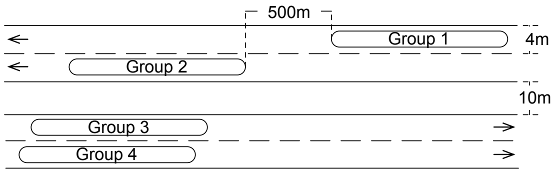
In this following, we consider three basic mobility scenarios which dissemination protocols must address, namely, a well-connected dense network, a



(a) Scenario 1



(b) Scenario 2



(c) Scenario 3

Figure 3.8: Mobility scenarios

network with a radio gap in one road direction, and a network with radio gaps in both road directions, as shown in Figure 3.8. We simulate a two kilometer highway with two lanes per road direction and four vehicle clusters (groups). Lanes are 4 meters wide with a 10 meter space between the directions. In Scenario 1, all four lanes are very busy, with 100 vehicles/km/lane. In each lane there is a group of 250 vehicles separated by 10 meters. The initial state is shown in Figure 3.8(a). Vehicles move at speeds between 2 and 2.5 km/h and there is always connectivity between the groups during the simulation time.

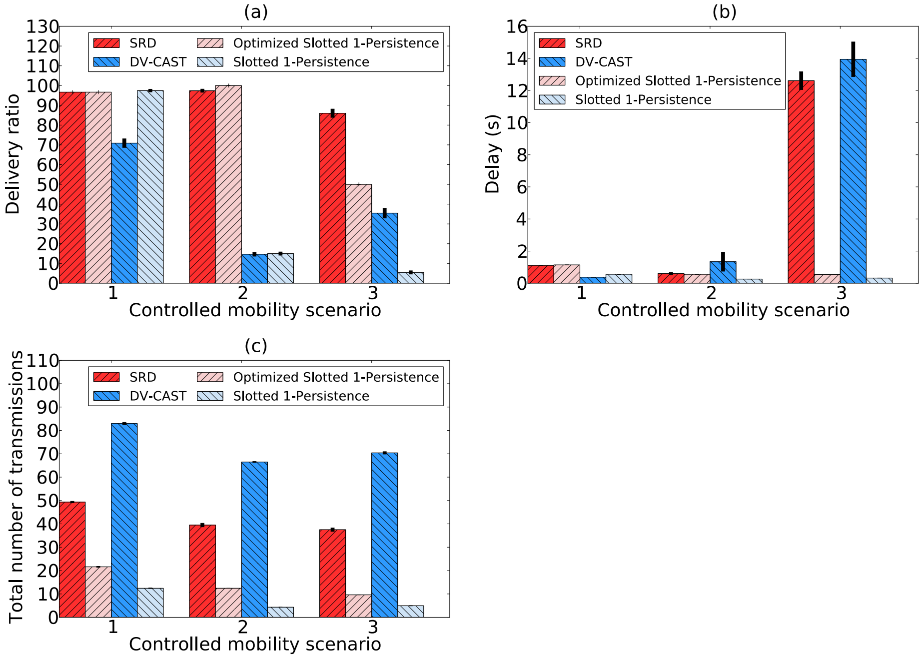


Figure 3.9: Results with 95% confidence intervals for controlled mobility scenarios

Scenarios 2 and 3 simulate situations with radio gaps between vehicles clusters. In scenario 2, shown in Figure 3.8(b), there is a 500 m gap between groups 1 and 2 in such a way that they cannot communicate directly. In this scenario, each group has a density of 20 vehicles/km/lane and vehicles move at speeds between 115 and 120 km/h. In scenario 3 (Figure 3.8(c)) the gap also exists but there are no vehicles moving in the opposite direction in the initial state. To bridge the gap, vehicles moving in the opposite direction use the store-carry-forward mechanism. Vehicle densities and speeds are the same as in scenario 2. Although there is a small vehicle speed variation in these scenarios, no overtaking or lane changing are simulated. Moreover, in each simulation run the duration time is set to 60 seconds and vehicles move at intervals of 0.1 seconds.

Figure 3.9(c) shows the number of transmissions for each scenario. In general, the number of transmissions performed by DV-CAST is notably higher compared with other protocols. Furthermore, Optimized Slotted 1-Persistence presents slight higher values when compared with Slotted 1-Persistence. The network load is lower for scenarios 2 and 3 due to their lower densities. These results are in line with the results obtained for the static scenarios and, therefore, their rationales are analogous.

With regard to delivery ratio in Figure 3.9(a), all protocols with the exception of DV-CAST achieve nearly 100% in scenario 1 where a well-connected network is present. In scenario 2, SRD and Optimized Slotted 1-Persistence present nearly 100% of delivery ratio whereas the percentages with DV-CAST and Slotted 1-Persistence are limited to 15%, which can also be explained by the effect of collisions in low densities. Finally, in scenario 3 the store-carry-forward model of each protocol is evaluated. While SRD achieves nearly 90% of delivery ratio against 50% with Optimized Slotted 1-Persistence, the percentage with DV-CAST is limited to nearly 40% against 5% achieved by Slotted 1-Persistence.

The performance results for end-to-end delay are shown in Figure 3.9(b). As expected, the delay values for scenario 1 and 2 are considerably lower than for scenario 3. In scenario 3, vehicles have to store, carry, and forward all messages from group 2 to group 1 in order to overcome the radio gap. Therefore, the delay is dependent on the speed of vehicles, which in this case leads to over 10 extra seconds compared to the remaining scenarios. In scenario 1, because of the use of later time slots for the low priority direction, SRD and Optimized Slotted 1-Persistence present higher delays than DV-CAST and Slotted 1-Persistence. This is also valid for scenario 2 with respect to Slotted 1-Persistence. However, DV-CAST presents an even higher delay due to the higher contention in the medium caused by more vehicles assigned to a single time slot.

While the results with respect to the network load correspond to the results presented for static scenarios, the delivery ratio gain when employing the store-carry-forward models in scenarios with separate vehicle cluster is evident. More specifically, the gain with SRD is of 40%, thereby reaching 90% of delivery ratio. In contrast, the gain with DV-CAST is of 35%, thereby reaching a maximum of only 40%.

3.1.3.2 Traffic simulator scenarios

After analyzing the performance of the protocols in controlled vehicle distributions, we now consider realistic vehicle distributions generated by professional traffic microsimulators, namely Quadstone Paramics 5.2 [82] and SUMO 0.11.1 [83]. The reason for relying on two different traffic microsimulators is that SUMO has been preferred in a later stage of this work because of its facilities to export vehicle traces to other software such as network simulators. Both simulators are widely used by researchers and professionals and support features such as overtaking, lane changing, and rely on well-known car-following mobility models such as Krauß and Intelligent Driver Model (IDM). In particular, the Krauß model is used in our simulations.

Three scenarios are considered in this set of simulations: a well-connected network (scenario 1), a sparse network (scenario 2), and a network combining both sparse and dense networks (scenario 3). Scenarios 1 and 2 are created with SUMO. We build a 10 kilometer straight highway with two lanes per road direction. Lanes are 4 meters wide with a 10 meter space between the directions. With this road, we differentiate the two scenarios by the traffic demands assigned to each one, namely, a moderate traffic flow generated for scenario 1 which leads to a density of 7.5 vehicles/km/lane and for scenario 2 a low traffic flow which leads to a density of 2.5 vehicles/km/lane. The speed at which vehicles move varies from 80 km/h to 120 km/h. The vehicle generation rate remains constant for both scenarios in such a way that the average density assigned remains also constant. Moreover, in each simulation run the duration time is set to 300 seconds.

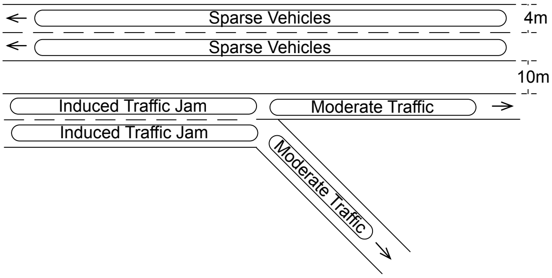


Figure 3.10: Traffic simulator scenario 3

Scenario 3 is shown in Figure 3.10. We simulate a 10 kilometer road with two lanes in each direction. In one direction, vehicles are sparsely distributed with density varying from 2 to 10 vehicles/km/lane while in the other direction a traffic jam is induced near a junction with a rapid increase in density from 20 to 40 vehicles/km/lane. The junction point is located at the center of the road (5 km) and it divides the road into two other roads with one lane each and with moderate traffic (10 to 20 vehicles/km/lane). The distribution of vehicles is generated by the Quadstone Paramics 5.2 traffic simulator executed with the CeeJazz plug-in. In the sparse and moderate traffic lanes, vehicles move on average at 120 km/h whereas in the section with an induced traffic jam, their speed drops rapidly from 120 km/h to 5 km/h. In this scenario, in each simulation run the duration time is set to 300 seconds and vehicles move at intervals of 0.5 second (standard frequency set in the CeeJazz plug-in).

All the three scenarios described have the particularity that vehicles are generated (allocated) in the simulation when they enter the road in one extreme and are deallocated when they reach the other extreme end of the road direction. Because we need to keep track of which messages are generated and successfully propagated along the complete road, we place one static network node in each end of the road: one node to generate (broadcast) messages, e.g., a crashed vehicle, positioned in the foremost position in the westerly direction; and another node, e.g., a road-side unit, to collect all messages and generate statistic results of the simulation. For this reason, the maximum theoretical delivery ratio is not guaranteed to be 100%, since at the moment when a message is generated there might be no vehicles within the transmission range to receive and further propagate the message along the road. In addition, in these traces vehicles perform lane changing, overtaking, and therefore change their order during the simulation. This realistic behavior helps us understand the level of robustness in a wide variety of traffic situations.

Figure 3.11(c) shows the performance of the protocols with regard to the number of transmissions. For all scenarios, the results present a similar pattern found in the controlled mobility scenarios. In scenario 2, because the network is very sparse, there is no multi-hop connectivity between any vehicle. The consequence is that the number of transmissions is equivalent for both SRD and DV-CAST protocols, whereas for the remaining protocols the values are nearly zero due to the lack of store-carry-forward mechanisms.

The results regarding the delivery ratio are shown in Figure 3.11(a). In scenario 1, where a well-connected network is present, protocols should achieve nearly 100%. This is the case for both SRD and Optimized Slotted 1-Persistence. In contrast, DV-CAST and Slotted 1-Persistence perform very poorly in the best case reaching 5% of delivery ratio. Because of the characteristics of scenario 2 and how we set up the simulation, the maximum delivery ratio is limited to a much lower value. In fact, because there is no multi-hop connectivity between vehicles and because *hello* messages which are constantly sent by each vehicle can collide with the messages generated by the static network node, the probability that a message is correctly received by any vehicle to be further disseminated along the road is much lower than in the other scenarios. For this scenario both SRD and DV-CAST achieve near 10% of delivery ratio. The remaining protocols present delivery ratio of zero, as expected in a very sparse

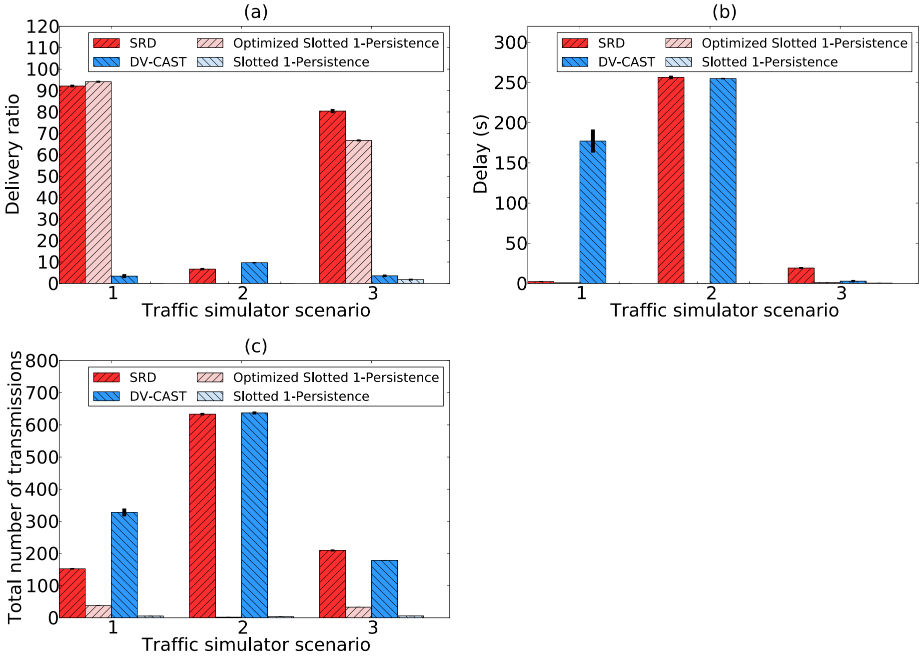


Figure 3.11: Results with 95% confidence intervals for traffic simulation scenarios

scenario.

The results for scenario 3 show a high difference in performance when comparing SRD and Optimized Slotted 1-Persistence with DV-CAST and Slotted 1-Persistence. SRD achieves 80% of delivery ratio against 65% achieved by Optimized Slotted 1-Persistence – 15% of gain with the use of the store-carry-forward model employed by SRD. In contrast, DV-CAST and Slotted 1-Persistence present delivery ratio of only approximately 4% and 2%, respectively. Such difference explains the higher number of transmissions performed by SRD in comparison with DV-CAST, since with a higher delivery ratio more vehicles rebroadcast along the road.

Figure 3.11(b) shows the results with respect to the end-to-end delay. For scenario 1, DV-CAST presents a much higher delay (over 150 seconds) when compared with the remaining protocols. This shows that Slotted 1-Persistence employed by DV-CAST does not disseminate the messages properly, since one would expect a quick message dissemination in a well-connected network. The high delay values indicate that DV-CAST can only deliver a few messages by using its store-carry-forward mechanism. In scenario 2, because of the lack of multi-hop connectivity, the delay is directly related to the speed at which vehicles move. Thus, the average of both SRD and DV-CAST is near 250 seconds. Finally, in scenario 3 the delay for SRD is higher than for other protocols. This is explained by the higher delivery ratio achieved when using its store-carry-forward mechanism, with some messages arriving later in the simulation.

This evaluation shows the significant improvement when using SRD and Optimized Slotted 1-Persistence over DV-CAST and Slotted 1-Persistence with regard to delivery ratio. Notably, the number of transmissions with DV-CAST is in certain scenarios higher even with such poor results with respect to delivery ratio.

3.1.3.3 Robustness test

In the following, we assess the mechanism which SRD employs to improve robustness in highly dynamic environments. As mentioned in Section 3.1.2.3, we define robustness as the ability of each protocol to cope with rapid changes in the network topology. Our goal is to evaluate the effects on the delivery ratio when running the protocols in scenarios including, but not limited to, vehicles leaving the road and vehicle crashes. In particular, we evaluate the ability of each protocol to cope with the situation shown in Figure 3.5 where the cluster tail leaves the road and the messages stored must still be forward to other vehicles ahead on the road despite such change in the network topology. Such situation is present in traffic simulator scenario 3, where a junction is present. However, since there is a constant change in topology in all traffic simulator scenarios, the impact on robustness becomes hard to be assessed. Therefore, for this assessment we reuse the scenario 3 among our controlled mobility scenarios. The reason for choosing this scenario is that it is the only controlled scenario where a radio gap is present and thus the only scenario possible to reproduce the situation described. In addition, only SRD and DV-CAST are evaluated since their suppression mechanism alone cannot handle radio gaps.

Identically to what is done in the controlled scenarios, messages are generated by the foremost vehicle in the westerly direction. We simulate the tail vehicles of the cluster formed by groups 2, 3, and 4 (Figure 3.8(c)) leaving the road by turning off their radios. After the tail is placed out from the road, this process is repeated and the new tail leaves the road after a pre-defined time. We define three different frequencies (intervals) at which vehicles leave the road: at every 5, 10, and 15 seconds. In this way, at the highest frequency, i.e., the current tail leaving the road at every 5 seconds, we simulate a highly dynamic environment.

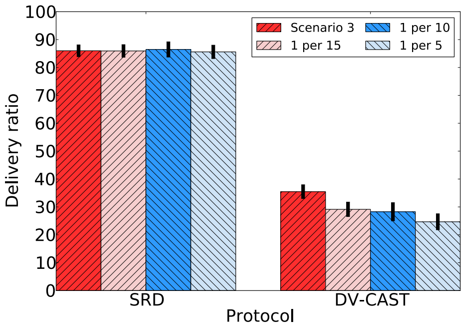


Figure 3.12: Delivery ratio x frequencies at which vehicles leave the road with 95% confidence intervals

The performance regarding the delivery ratio is shown in Figure 3.12. We compare the results obtained in the simulations with vehicles leaving the road at each frequency side-by-side with the previous results obtained in the original controlled mobility scenario 3. The results for this simple scenario validate the mechanism employed by SRD by showing an unaltered performance for all frequencies when compared with the previous results in scenario 3. In contrast, DV-CAST is affected with a decrease in delivery ratio as the frequency increases. When tail vehicles leave the road at a rate of one at every 5 seconds, the delivery ratio decreases from 35% to 25%. This deterioration is explained by the reliance on a single vehicle by DV-CAST to store, carry, and forward messages in sparse networks. As explained in Section 3.1.2.3, with SRD all vehicles which receive a message with the flag *FromTail* equal to true will store the message and act as back-up vehicles in case the tail vehicle leaves the road or fail, thereby increasing robustness in such usual road scenarios.

With regard to the network load, the results follow the same pattern as the ones obtained for the delivery ratio. More specifically, in the results obtained with DV-CAST there is a decrease in transmissions when higher frequencies for the cluster tail leaving the road are considered. The end-to-end delay remain unaltered. With respect to SRD, all results are practically analogous with negligible variations compared to the ones obtained for the controlled mobility scenario 3. For these reasons, such results are not depicted.

3.1.3.4 Effects of the total number of time slots

Another important aspect is to assess the impact of choosing various values for the parameter regarding the total number of time slots (NS). Notably, this parameter influences directly on the performance of the suppression mechanisms employed by SRD and DV-CAST protocols as it defines the number of vehicles assigned to each time slot and thus the number of vehicles rebroadcasting near simultaneously. For this evaluation, we run both SRD and DV-CAST with NS values varying from 1 to 10. However, because SRD always rely on an equal number of time slots for each road direction, only even numbers from this range are evaluated for SRD. We choose traffic simulator scenario 3 for this simulation, due to the presence of a high dynamic road environment which yields to a wide variety of situations. Since the suppression mechanisms employed by both SRD and DV-CAST have already been evaluated separately in previous sections, we omit their assessment here.

We start our discussion with Figure 3.13(a) where the delivery ratio for each protocol is shown for increasing values of NS. While SRD improves its performance with higher values for the number of time slots, DV-CAST performs poorly and reaches delivery ratio of nearly zero percent when NS is equal to 4, going down to zero in the remaining values. As explained previously, DV-CAST presents poor results when there are few vehicles assigned to each time slot, which indicates that collisions have a high impact on its functioning. On the other hand, the optimizations proposed in SRD clearly deals efficiently with different values for NS. More specifically, with fewer vehicles assigned to each time slot fewer rebroadcasts are performed. With the optimizations proposed to tackle collisions and increase robustness by being more strict when canceling rebroadcasts, fewer collisions are also present which in turn allows for a better performance in delivery ratio. As we emphasized before, because of the characteristics of the simulation in traces generated by traffic simulators, 100% of delivery ratio is not possible in this scenario.

Figure 3.13(c) shows the results for the number of transmissions. Generally, the number of transmissions decreases with a higher number of time slots. While with SRD this result is directly explained by the fewer number of vehicles rebroadcasting in each time slot, the lower values with DV-CAST are directly related to the poor delivery ratio which yields obviously in fewer transmissions. After the delivery ratio reaches zero percent for DV-CAST, the values for the number of transmissions remain stable. This might indicate that

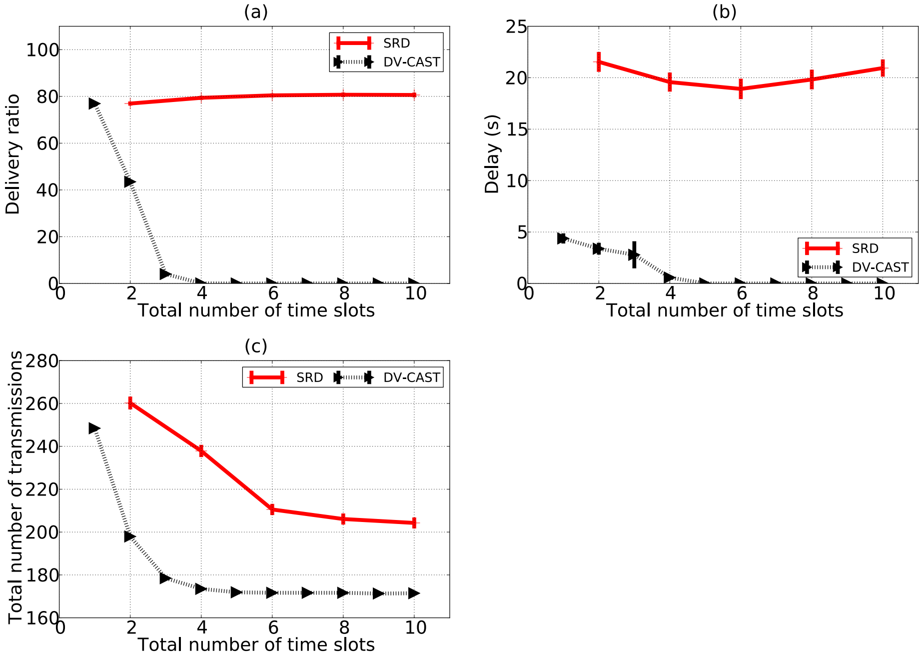


Figure 3.13: Results with 95% confidence intervals for varying the time slot parameter in traffic simulation scenario 3

the message propagation ends (e.g., due to collisions) at similar points in the end-to-end path, which makes the results for the network load very similar with predominantly *hello* messages being sent an received.

Finally, the performance regarding the end-to-end delay is shown in Figure 3.13(b). With SRD, the decrease in delay up to NS equal to 6 is a result of the presence of fewer collisions per time slot. With fewer collisions, the probability that the farthest vehicle from the sender succeeds in rebroadcasting is higher, which decreases the multi-hop end-to-end delay. After NS equal to 6, the delay start to increase. This is an indication that the earliest time slot is not always utilized due to the absence of vehicles assigned to it. Because of the higher delivery ratio and the consequent higher use of the store-carry-forward mechanism, the delay values for SRD are generally higher than for DV-CAST. The delay also decreases with DV-CAST also due to the fewer number of messages that travels the complete road.

From the results above, assigning NS to 6 seems to be the optimal value for this scenario. This can be concluded based on the delay and the results achieved for the network load and delivery ratio.

3.1.3.5 Effects of *hello* messages

Our final evaluation concerns the impact that *hello* messages have on the performance of SRD. As described in Section 3.1.2.4, SRD relies on what we call a *suppressed* periodic *hello* message mechanism. However, a beaconing [8] mechanism is expected to coexist with other systems in a vehicle. Such a mechanism sends out periodic messages called *beacons* which have the same function as *hello* messages and contain information such as geographical location, speed, and acceleration. Therefore, it is also important to evaluate how SRD behaves when employed with a regular *un-suppressed hello* message mechanism. To accomplish this evaluation, we remove the mechanism used by SRD to gather the minimum topology information required, namely, the MDC timer and the use of *suppressed hello* messages. Instead, we insert an equivalent *hello* message mechanism as employed by DV-CAST. Thus, *hello* messages contain only the vehicle ID, message ID, timeStamp, and the sender coordinates in order to derive which vehicles are the cluster front and tail. We compare this SRD version with SRD using *suppressed hello* messages and also with DV-CAST. We choose traffic simulator scenario 3 for this evaluation.

Figure 3.14(c) shows the results for the number of transmissions. With an un-suppressed scheme, the higher numbers of transmissions when compared with SRD running the *suppressed* mechanism is evident. Such numbers are also higher compared to DV-CAST, which is explained by the higher delivery ratio as shown in Figure 3.14(a). In fact, the use of *un-suppressed hello* messages reduces the delivery ratio in approximately 5%. However, compared with DV-CAST such loss is negligible. Finally, Figure 3.14(b) shows that with an *un-suppressed hello* message scheme, lower delay values are present. Since the delivery ratio achieved by the protocols diverge, the assessment of the end-to-end delay becomes difficult. For instance, if more messages succeed in propagating the complete road path, some messages could arrive later due to radio gaps and thus increase the end-to-end delay average achieved by a certain protocol. This can be one reason for such lower values. Another reason can be that with *un-suppressed hello* messages, vehicles are able in this scenario to estimate their current role in the cluster (front or tail) more quickly and accurately leading in this way to a quicker propagation along the road. Further study is necessary to validate this assumption.

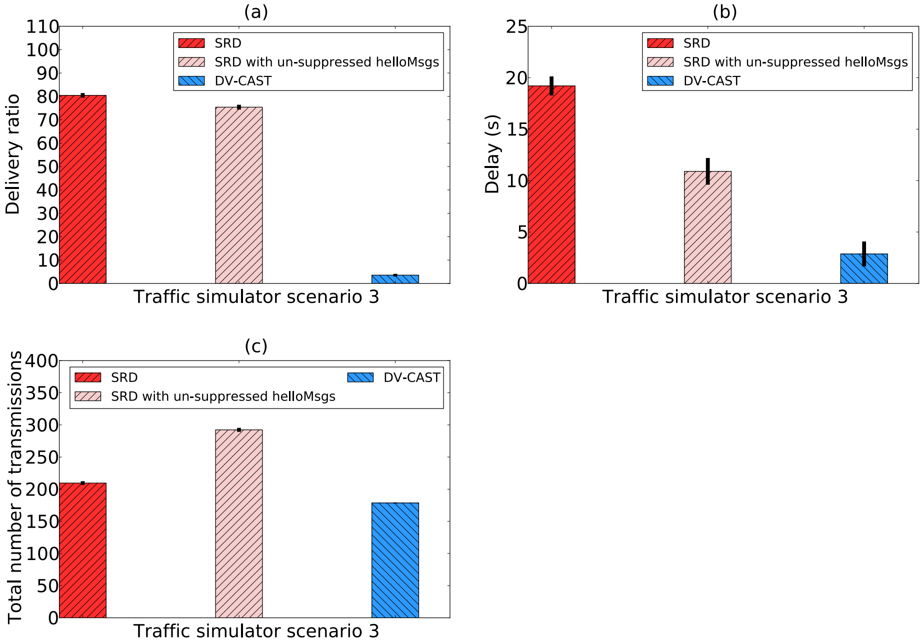


Figure 3.14: Results with 95% confidence intervals for SRD running with *un-suppressed hello* messages in traffic simulation scenario 3

Overall, SRD with an *un-suppressed hello* messages presents a similar performance in terms of delivery ratio when compared with SRD running its *suppressed hello* message mechanism. The main difference lies in the higher number of transmissions when using the *un-suppressed* mechanism.

3.1.4 Conclusion

In this section, we have presented a dissemination protocol suitable for both sparse and dense vehicular networks. The use of suppression techniques has been motivated and employed in dense networks while the store-carry-forward communication model has been used in sparse networks. The designed protocol is both *simple* and *robust*. We have proposed an optimized suppression technique which is based on the Slotted 1-Persistence [32]. Furthermore, because SRD requires only limited local topology information, we have presented an efficient periodic *hello* message mechanism in which only a subset of vehicles is required to participate. Our simulation results show that SRD outperforms DV-CAST in terms of delivery ratio for the diverse set of scenarios considered and introduces a lower load into the network. In addition, SRD presents better performance with regard to robustness in highly dynamic scenarios where vehicles move to different roads frequently.

3.2 Achieving scalability in dense highways

3.2.1 Introduction

Just as the approach described in Section 3.1, most suppression techniques designed to cope with the broadcast storm problem in dense networks aim to assign vehicles to different delay values that are inversely proportional to their distance to the sender. The goal is to let the farthest vehicles rebroadcast first and suppress the transmission of other vehicles, thereby allowing for quick data dissemination [32]. This separation in time is accomplished by means of time slots, where each time slot is equivalent to a message's transmission time. However, as highlighted previously in Figure 2.2, the number of vehicles rebroadcasting nearly simultaneously in a single time slot can increase considerably in dense networks, thereby still leading to undesirable levels of contention and collision [31]. Since time slots match regions within the transmission range of the sender, another problem occurs when there is simply no vehicle assigned to the earliest time slot, what increases the end-to-end delay.

In this section, we present a suppression technique for dense networks, namely, the **D**istributed **O**ptimized **T**ime (DOT) slot scheme. We focus on solving *scalability* issues of current approaches by controlling with high precision the density of vehicles within each time slot. To accomplish this goal, we exploit the presence of *beacons*, which are messages periodically sent by each vehicle containing information such as the vehicle's position and speed. As briefly mentioned in Section 3.1.3.5, *beacons* have the same function as *hello* messages with regard to providing awareness of neighbors' presence in the network vicinity. While the use of periodic *beacons* or *hello* messages has been sometimes avoided due to an increase in the network load [32], *beacons* have been an important topic of research and are expected to be massively present to increase cooperative awareness in safety applications [41]. Since such beaconing mechanism is expected to coexist with other systems in the vehicle, we advocate in the remainder of this chapter relying only on the presence of *beacons* to design dissemination solutions for safety applications.

3.2.2 Optimized time slot scheme

DOT aims at always selecting the farthest vehicles, i.e., optimal relay vehicles, while controlling transmission redundancy used to increase robustness. To achieve this goal, DOT has the following characteristics:

- ***Time slot density control***: it exploits positioning information of 1-hop neighbors to control with precision the time slots′ boundaries and, therefore, the number of vehicles assigned to each time slot. This prevents the uneven distribution of vehicles among time slots (Figure 2.2(a)) when a simple matching of time slots into fixed regions within the transmission range of the sender is used. As a result, transmission redundancy is controlled and end-to-end delay is kept at a minimum, as there is always a vehicle assigned to the earliest time slot.

- ***Distributed***: each vehicle takes the decision regarding when to retransmit a message in a distributed fashion. This prevents sub-optimal selections of a relay vehicle as it can occur with a centralized decision (Figure 2.2(b)).

3.2.2.1 Requirements and assumptions

The scheme relies on the existence of periodic *beacons* transmitted by each vehicle at a certain rate. These *beacons* are defined to be transmitted in the form of WAVE Short Messages (WSMs), according to the IEEE 1609 Family of Standards for Wireless Access in Vehicular Environments (WAVE) [84, 85]. The IEEE WAVE standard determines that these messages carry information such as the data rate, channel number and the transmission power level employed. In addition, contextual information about the vicinity is expected to be included, namely, the vehicle's geographical position, speed and acceleration [8]. We assume that each vehicle is equipped with a device capable of obtaining the current vehicle's geographical position, such as a GPS receiver. Therefore, we consider the following message header structure: <*Vehicle ID, Message ID, Time Stamp, Vehicle's Geographical Coordinates, Power Level*>.

3.2.2.2 The protocol

By gathering the information contained in *beacons*, each vehicle keeps a table of one-hop neighbors T_n containing the latest information about the vicinity. Each entry in T_n contains the following information: <*Vehicle ID, Expiration Time, Vehicle's Geographical Coordinates*>. The Expiration Time field is used to remove vehicles from the table that are no longer in the vicinity. Since there may be failures (e.g., collisions) when sending these *beacons*, we introduce a time tolerance before removing an entry defined as $t_t = 2.5(\frac{1}{b_f})$, where b_f is the beaconing rate, e.g., 10 Hz. This accounts for failure in one beaconing period plus possible extra delay.

The DOT protocol works as follows. Let i be the vehicle sender of message m, and R be the set of vehicles that received m. Every vehicle $j \in R$ schedules a rebroadcast for m with a time delay T_{Sij}. If any vehicle $j \in R$ receives an echo of m before T_{Sij} expires, it cancels its rebroadcast and ignores future duplicates of m.

The process for defining T_{Sij} consists of two main tasks: (i) estimating which vehicles are within the transmission range of the sender and received m, i.e., belong to set R; and (ii) sorting the entries of every vehicle $j \in R$ in table T_n based on its geographical position relatively to the sender. The first task is achieved by using the power level included in m when transmitted by i. We elaborate on such an estimation in Section 3.2.2.3. In the second task, based on the transmission range estimation of the sender, each vehicle receiving m makes a list $\vec{v}$ with all its neighbors in T_n that also belong to set R. These vehicles are then sorted by their distance relatively to sender i, where the farthest vehicle is the first element in $\vec{v}$. In case different vehicles are equally distant from the sender, they are sorted also by their vehicle ID, where lower ID values are placed in front positions in $\vec{v}$.

Figure 3.15 exemplifies this distributed sorting algorithm. In this example, vehicle v_2 receives a message from the sender and calculates its order among the neighbors in its table T_n that may also be in the range of the sender, namely, vehicles v_1, v_3 and v_4. With the geographical position of these vehicles in T_n, v_2 sorts these vehicles as $\vec{v} =< v_4, v_3, v_2, v_1 >$.

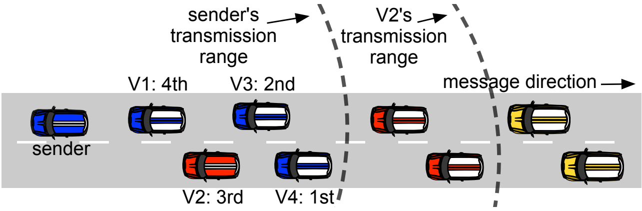


Figure 3.15: Distributed sorting algorithm

With the sorted list of vehicles $\vec{v}$, each vehicle $j \in R$ finds its own position in $\vec{v}$. We denote this position as $S_{ij} \in [0, n-1]$, where n is the total number of elements in $\vec{v}$. Next, the time that vehicles have to wait before rebroadcasting is given by:

$$T_{Sij} = st\left(\left\lceil \frac{(S_{ij}+1)}{ts_d} \right\rceil - 1\right) + AD_{ij}, \tag{3.8}$$

where the main parameter ts_d determines the number of vehicles that are allowed to be assigned to a single time slot. In other words, this parameter enables the control of time slots' density. The slot time st is an estimated value of the total time taken for the transmission to complete and the message be fully received by others, accounting for medium access delay, transmission delay and propagation delay.

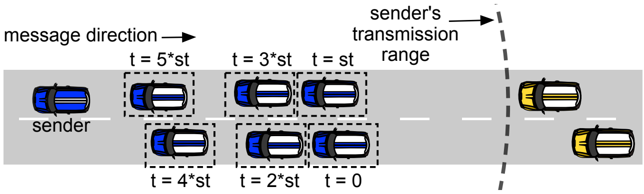
Just as proposed in Section 3.1.2.2, we introduce a small additional delay to break the synchronization between the transmissions of vehicles that are assigned to a single time slot. This occurs in our approach when $ts_d > 1$. The additional delay AD_{ij} is defined as:

$$AD_{ij} = d\left(S_{ij} \bmod ts_d\right), \tag{3.9}$$

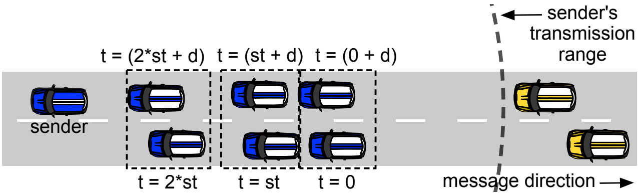
where d is a time delay sufficiently long for vehicles assigned to the same time slot to sense if other vehicle has already started its transmission and at the same time sufficiently low not to overlap with the beginning of later time slots, i.e., $d \ll st$. Example of possible values that meet these requirements are the SIFS and DIFS parameters in the MAC 802.11p.

Figure 3.16 shows how our mechanism works when different values for ts_d are used. With $ts_d = 1$, all vehicles in the range of the sender are assigned to individual time slots based on their distance to the sender, as shown in Figure 3.16(a). Thus, rebroadcasts are separated in time by multiples of slot time st. In our second example in Figure 3.16(b), $ts_d = 2$ is used. In this case, two vehicles are assigned to each time slot. To prevent nearly simultaneous rebroadcasts among the two vehicles in each time slot, the vehicle with higher S_{ij} value, i.e., nearer to the sender, waits the additional delay $AD_{ij} = d$.

With an accurate estimation of set R, optimal results in terms of transmission redundancy and end-to-end delay are achieved when $ts_d = 1$. This leads to the minimum number of rebroadcasts and also to the lowest end-to-end delay, since only optimal relay vehicles, i.e., farthest vehicles from the previous sender, rebroadcast in the earliest time slot. Vehicles assigned to later time slots would cancel their rebroadcasts upon receiving an echo of the message being propagated. However, there are a few factors that can prevent the optimal estimation of set R, as we discuss in the following section.



(a) $ts_d = 1$



(b) $ts_d = 2$

Figure 3.16: Examples of different settings for the time slot density parameter

3.2.2.3 Estimating vehicles in the sender's range

As discussed in Section 3.2.2.2, DOT depends on accurately estimating which vehicles are within the transmission range of the sender, i.e., belong to set R. On the one hand, underestimated transmission range values may lead to an excessive number of vehicles assigned to earlier time slots. This occurs because all vehicles beyond the underestimated range are assigned to the first position in list $\vec{v}$. On the other hand, overestimated values may result in longer delays, since vehicles unnecessarily wait for the rebroadcast of other vehicles that actually did not receive any message.

Just as with many vehicles assigned to a single time slot, underestimating the transmission range can lead to multiple vehicles transmitting nearly simultaneously. This may result in collisions and mean the end of a message's dissemination. To prevent this effect, we introduce the following policy. If a vehicle j is beyond the range estimated, it is assigned to the last position in list $\vec{v}$. If $\vec{v}$ is empty, j transmits immediately after a random small delay taken from the interval $[0, d]$. This policy may increase the end-to-end delay but it maintains the protocol robust against collisions and contention. On the other hand, we tackle overestimated values by being conservative when assuming the maximum distance from the sender that neighbors are still able to receive a message. This can be done by requiring a low outage probability in the propagation model assumed as we show in the following.

There are two main factors that can affect the estimation of vehicles that belong to set R: (i) error in positioning information and, thus, inaccurate positioning of vehicles in T_n; and (ii) path loss affects in wireless communication such as free-space loss, shadowing, and Doppler effect. While the accuracy of a positioning device such as GPS is generally fixed in the order of a few meters, i.e., 5 meters in outdoor environments [86], in wireless communication the communication range estimation mainly depends on the radio propagation model assumed. Although choosing an appropriate propagation model depends on the current scenario, e.g., if it is urban or a highway, we briefly show in the following how the *outage probability* can be used to estimate the transmission range when the simple log-normal shadowing model is assumed [87]. Our analysis is thus limited to show the feasibility of using the power level employed to derive the transmission range achieved. It is importance to notice, however, that for a realistic transmission range estimation, further evaluation is required to determine proper propagation models for different vehicular networks scenarios.

According to the log-normal shadowing model, the received power in dB is calculated as:

$$[P_r]_{dB} = [P_r]_{dB} + K_{\mathrm{dB}} - 10\gamma \log_{10}(d/d_0) - \psi_{\mathrm{dB}}, \tag{3.10}$$

where P_r is the received power at distance d; P_t is the transmit power (included in the sender's message); K_{dB} is the unit power loss in dB which depends on the antenna properties of the transceivers; γ is the path loss coefficient which depends on the radio environment; d_0 is the reference distance; ψ_{dB} is a Gauss distributed random variable with zero mean and with $\sigma^2_{\psi_{\mathrm{dB}}}$ as the variance generated due to the shadowing effect. We consider the channel as either slow fading or as very fast fading. Slow fading and very fast fading channels have almost the same performance as additive white Gaussian noise (AWGN) channels. To analyze the performance of wireless communication in AWGN we have to consider two criteria of interest: the bit error probability and the outage probability. In AWGN, for the BPSK modulation the bit error probability is defined as:

$$P_b = Q\left(\sqrt{2\gamma_0}\right). \tag{3.11}$$

The outage probability P_{out} is the probability that the received signal's average SNR $\bar{\gamma}_s$ falls below the minimum required SNR for the pre-defined acceptable communication performance γ_0 [87, 88]. Mathematically,

$$P_{out} = p\left(\bar{\gamma}_s < \gamma_0\right) = \int_0^{\gamma_0} p_{\gamma_0}\left(\gamma\right) d\gamma. \tag{3.12}$$

From Equations 3.11 and 3.12 the required average SNR is:

$$\bar{\gamma}_s = \frac{\left(Q^{-1}\left(P_b\right)\right)^2}{-2\ln\left(1 - P_{out}\right)}. \tag{3.13}$$

In AWGN, the carrier-to-noise ratio of the received signal is:

$$\left(\frac{C}{N}\right)_{dB} = \left[\frac{E_b}{N_0}\right]_{dB} + \left[\frac{f_b}{B}\right]_{dB}, \tag{3.14}$$

where $\frac{E_b}{N_0} = \bar{\gamma}_b$ is SNR per bit; f_b is the channel data rate (net bitrate); and B is the channel bandwidth [87]. As $C_{dB} = 10\log\left(P_r\right)$ and for the BPSK modulation $\bar{\gamma}_b = \bar{\gamma}_s$, Equation 3.14 can be re-written as:

$$\left[P_r\right]_{dB} = \left[\bar{\gamma}_s\right]_{dB} + \left[\frac{f_b}{B}\right]_{dB} + N_{dB} \tag{3.15}$$

Finally, using Equations 3.10, 3.13 and 3.15 the transmission range can be estimated by:

$$d = d_0 \times 10^{\frac{[P_t]_{dB} + K_{\mathrm{dB}} - \left[\frac{\left(Q^{-1}\left(P_b\right)\right)^2}{-2\ln\left(1 - P_{out}\right)}\right]_{dB} - \left[\frac{f_b}{B}\right]_{dB} - N_{dB}}{10\gamma}} \tag{3.16}$$

In wireless transceiver design, a typical BER of 10^{-4} and 2% outage probability are considered acceptable in performance. Figure 3.17 shows estimated transmission range values for increasing outage probabilities. We assume in our simulation, transmission range values with 2% of outage probability. Thus, values are approximately 100, 150, 200 and 300 meters for power values of 300, 800, 2800 and 10000 mW, respectively. The effects of transmission range estimation errors is further evaluated in Section 3.2.3.4.

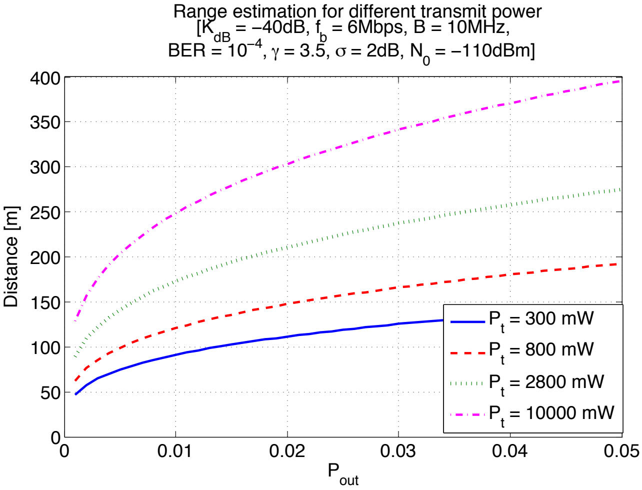


Figure 3.17: Transmission range estimation for increasing accepted outage probabilities and power levels

3.2.3 Performance evaluation

The performance evaluation of DOT is carried out by means of simulations. Our goal is to study the scalability of DOT under diverse scenarios by comparing it with two state-of-the-art suppression techniques, namely:

- ***Slotted 1-Persistence***: it is the mechanism that achieved best performance in terms of end-to-end delay among the two slotted schemes proposed in [32].
- ***Optimized Slotted 1-Persistence***: it relies on an optimized version of the Slotted 1-Persistence suppression method to prevent nearly simultaneous rebroadcasts in a single time slot in dense networks as presented in Section 3.1.

We utilize the MiXiM Framework [81] and adjust the available implementation of the IEEE 802.11b protocol to comply with basic specifications of the 802.11p version. Table 3.2 contains a summary of the simulation parameters.

In the MAC layer, we set the bit rate to 6 Mbit/s, the Contention Window (CW) to values between 15 and 1023, the slot time to 13 μs, the SIFS to 32 μs, and the DIFS to 58 μs. In the physical layer, we operate on the 5.88 GHz frequency band, with 10 MHz of bandwidth. Based on our estimates in Section 3.2.2.3, we set the transmission power to 800 mW to achieve approximately 150 meters of communication range with outage probability of 2%. We use the Friis Free Space Path Loss (FSPL) propagation model with exponent α equal to 3.5, as it is within the range 2.7 to 5, estimated for outdoor shadowed urban areas in [89]. In addition, we include shadowing effects that are modeled following a log-normal distribution with zero mean and standard deviation $\sigma = 6.25$ dB, as it is within the range 4 to 12 dB for outdoor propagation conditions according to [89]. The bit error model used is the one provided by the Veins project [90], which is based on measurements from [91] for the 6 Mbit/s bitrate.

For all suppression mechanisms, we set the slot time st to 5 ms. We define the total number of time slots for Slotted 1-Persistence NS_{std} to 3 and for Optimized Slotted 1-Persistence we set NS_{opt} to 6 (3 slots for each road direction as defined in Section 3.1). The value chosen for Slotted 1-Persistence is based on simulation parameters used in [47]. The maximum additional delay D_{max} used by Optimized Slotted 1-Persistence is set to 1 ms. Finally, for the DOT mechanism we set the time slot density ts_d to 1 and additional delay d to DIFS.

For all simulation scenarios the message size is 2312 bytes large, the maximum allowed by the 802.11p standard. Data messages are generated every 2 seconds, i.e., message frequency of 0.5 Hz. Each message is generated by one fixed vehicle positioned in one end of the road and gathered by another fixed vehicle in the other end of road. For each simulation scenario 20 runs of 100 seconds are executed. Finally, *beacons* are 24 bytes large and sent at 1 Hz. This represents the worst-case scenario in terms of freshness of the one-hop neighborhood information, since *beacons* are usually assumed to be sent at from 1 to 10 Hz [8]. Furthermore, varying the beaconing rate in early experiments has not led to significant changes in our simulation results, expect for an equally lower delivery ratio for all evaluated schemes due to a higher network load.

We consider a scenario with a 1-kilometer straight highway with two lanes in each road direction. This scenario was created with SUMO [83]. Therefore, it includes realistic mobility patterns such as vehicle overtaking, lane changing, and relies on the well-known car-following mobility model. Vehicles' speeds vary according to the density considered by following the Krauß mobility model, i.e., the higher the density is, the slower vehicles move.

Our evaluation considers the following metrics:

Table 3.2: Simulation parameters

Physical layer	Frequency band	5.88 GHz
	Bandwidth	10 MHz
	Transmission range	~150 m
	FSPL exponent α	3.5
	Log-normal σ	6.25 dB
	Receiver sensitivity	-119.5 dBm
	Thermal noise	-110 dBm
	Bit error model	Based on [91]
Link layer	Bit rate	6 Mbit/s
	CW	[15,1023]
	Slot time	13 μs
	SIFS	32 μs
	DIFS	58 μs
Suppression mechanisms	st	5 ms
	ts_d	1
	d	DIFS
	NS_{std}	3
	NS_{opt}	6
	D_{max}	1 ms
	Beacon size	24 Bytes
	Beacon frequency	1 Hz
Scenarios	Data message size	2312 bytes
	Data message freq.	0.5 Hz
	Network density	50 veh./km/lane
	# Runs	20

- ***Delivery ratio***: the percentage of messages generated by the farthest vehicle in one end of the road which fully propagate and are received by a vehicle in the extreme opposite end of the road. Ideally, dissemination protocols must achieve a delivery ratio percentage close to 100% in dense networks.

- ***Delay***: the total time taken for a message to propagate from one end to the other of the road length. This is particularly important for critical safety messages that must be disseminated as quickly as possible. We additionally compare the performance of each protocol with a theoretical optimum which serves as lower bound. This value is simply calculated as the minimum number of hops that a message must travel times the transmission delay, given the transmission range employed.

- ***Total number of transmissions***: the total number of transmissions performed on average by an arbitrary vehicle. We consider only data messages in these results, thereby excluding transmissions of *beacons*. This value is normalized by the total number of vehicles in each scenario. In order to be scalable, protocols must keep a low number of transmissions during a message's dissemination.

3.2.3.1 Network density

We first study the performance of the protocols with increasing network densities. Since we focus on dense networks, we fix the parameters in Table 3.2 and vary the density from 20 to 100 vehicles/km/lane.

As shown in Figure 3.18(a), Slotted 1-Persistence improves its delivery ratio up to 60% as the network density increases. This is explained by the extra rebroadcast redundancy which occurs when more vehicles are assigned to a single time slot. In contrast, DOT maintains performance of near 100% for all density values, whereas Optimized Slotted 1-Persistence reaches 100% up to density of 60 vehicles/km/lane and suffers a decrease of up to 10% with density of 100 vehicles/km/lane.

Figure 3.18(b) shows the performance with respect to the end-to-end delay. The end-to-end delay tends to increase with density for protocols that rely on a fixed number of time slots such as Slotted 1-Persistence and Optimized Slotted 1-Persistence. This can be reasoned by the higher contention delay generated when more vehicles attempt to rebroadcast in a single time slot. In contrast, DOT scales properly with increasing densities. In fact, the higher the density of the network, the higher the chance is that a vehicle is positioned closer to the border of the transmission range. Thus, delay values with DOT are close to the theoretical optimum in densities ranging from 30 to 100 vehicles/km/lane.

The number of transmissions performed by Slotted 1-Persistence and Optimized Slotted 1-Persistence also increases with higher densities, as shown in Figure 3.18(c). This is due to the higher number of vehicles positioned in the geographical region corresponding to a single time slot. By relying on the control of time slot density, DOT scales properly with higher densities. In fact, in proportion with the total number vehicles in each density, the number of transmissions tends to decrease.

In general, DOT scales more efficiently with increasing network densities when compared with traditional methods that employ fixed time slots such as

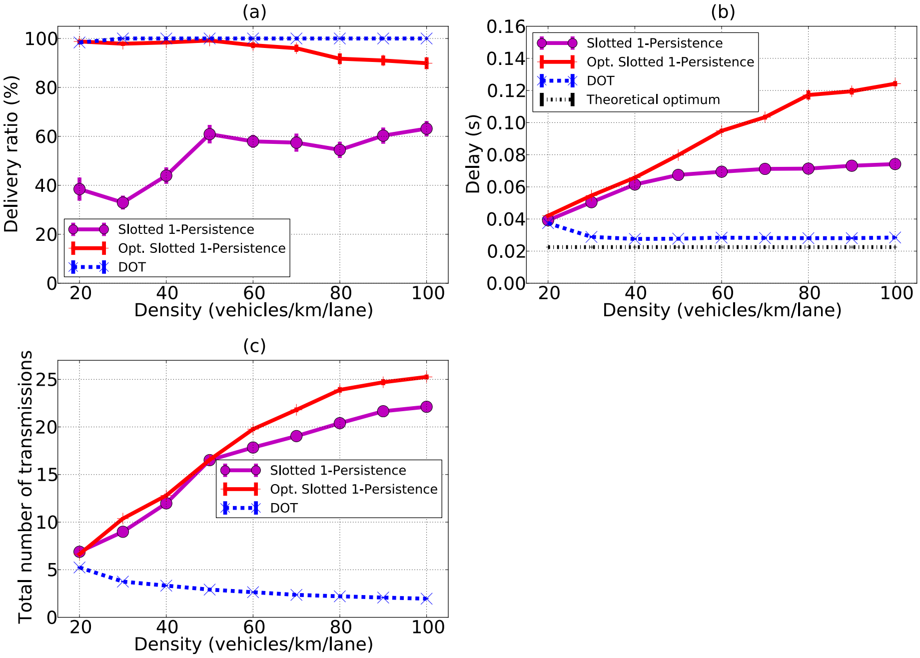


Figure 3.18: Results with 95% confidence intervals for increasing network densities

Slotted 1-Persistence and Optimized Slotted 1-Persistence.

3.2.3.2 Transmission range

Another important aspect is the performance of protocols when different transmission ranges are employed by vehicles. Specially for approaches that employ a fixed number of time slots, increasing the transmission range affects directly the size of each time slot and, thus, the performance of protocols. In addition, it affects the number of hops required for a message to travel the complete highway considered. We fix the parameters in Table 3.2 and vary the transmission range from 100 to 300 meters. Additionally, we consider the scenario *mix* where different vehicles employ different transmission ranges. More specifically, each of the ranges 100, 150, 200 and 300 meters is used by 25% of the vehicles. Each vehicle takes a range value in the beginning of the simulation run and employ it until the simulation ends.

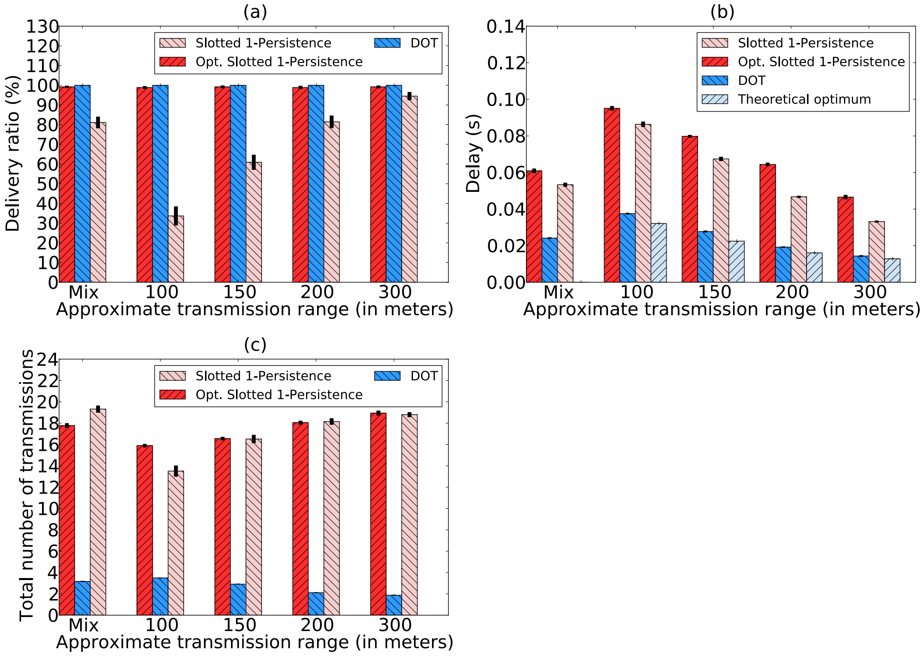


Figure 3.19: Results with 95% confidence intervals for different transmission range settings

Figure 3.19(a) shows the performance of protocols with respect to the delivery ratio. Both Optimized Slotted 1-Persistence and DOT protocols achieve near 100% in every transmission range setting. In contrast, Slotted 1-Persistence shows higher delivery ratio when considering higher transmission range values. This is explained by the extra rebroadcast redundancy and fewer hops needed for a message to be fully disseminated when higher transmission ranges are employed. Furthermore, Slotted 1-Persistence is affected when different ranges are employed by different vehicles, which can also be explained by the higher number of hops required on average for a message's dissemination.

With fewer hops needed for a message to travel, the end-to-end delay presented by each protocol also decreases when higher transmission ranges are employed (Figure 3.19(b)). DOT presents the lowest delay, with values near the theoretical optimum for each range setting.

Figure 3.19(c) shows the total number of transmissions performed by each protocol. Since Slotted 1-Persistence and Optimized Slotted 1-Persistence adopt a fixed time slot approach, higher transmission ranges means more vehicles assigned to a single time slot. Therefore, more rebroadcast redundancy and thus more transmissions are expected. On the other hand, DOT controls the time slot density regardlessly of the current density of vehicles within the transmission range. Therefore, fewer hops is translated to fewer transmissions.

Results show that not only DOT scales properly with increasing and heterogeneous transmission range settings, but also achieves near optimum performance in terms of end-to-end delay.

3.2.3.3 Time slot parameter

In the following, we analyze the performance of protocols when varying their main parameters, namely, the total number of time slots ts_n (used by Slotted 1-Persistence and Optimized Slotted 1-Persistence) and the time slot density ts_d (used by DOT). Other parameters are fixed as shown in Table 3.2. In particular, Optimized Slotted 1-Persistence uses doubled values of ts_n to distribute the number of time slots equally among the two road directions, as detailed in Section 3.1.

With regard to the delivery ratio, both Optimized Slotted 1-Persistence and Slotted 1-Persistence achieve higher delivery ratio when increasing the total number of time slots, as shown in Figure 3.20(a). In fact, employing more time slots leads to a lower number of vehicles assigned to a single time slot. Therefore, a lower level of rebroadcast redundancy is expected and messages can travel with less interference throughout the road length. The opposite effect occurs when the time slot density is increased in DOT. Higher values for ts_d means more vehicles within a single time slot, which leads to a decrease in delivery ratio from $ts_d = 4$ in this scenario.

Equivalently to what occurs when varying the network density, there is an increase in delay when more vehicles attempt to transmit nearly simultaneously in a single time slot (Figure 3.20(b)). This occurs when decreasing ts_n (Optimized Slotted 1-Persistence and Slotted 1-Persistence) or increasing ts_d (DOT). Such an increase in the number of transmissions can be confirmed in Figure 3.20(c). In general, the increase in delay is upper bounded by the network density in the scenario considered, which consequently limits the maximum number of vehicles that are within the transmission range of 150 meters.

In general, all protocols perform best when fewer vehicles attempt to transmit nearly simultaneously. This means $ts_d = 1$ for DOT and $ts_n = 8$ for Optimized Slotted 1-Persistence and Slotted 1-Persistence. However, while find-

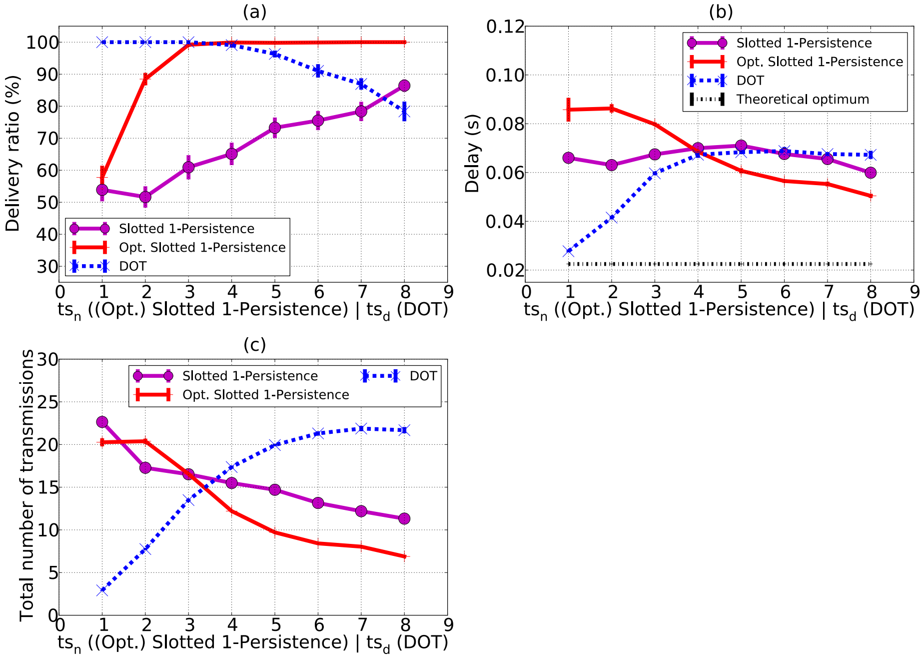


Figure 3.20: Results with 95% confidence intervals for different time slot parameters used by each protocol

ing the optimal value for ts_n in Optimized Slotted 1-Persistence and Slotted 1-Persistence depends on accurately knowing the current network density, DOT with $ts_d = 1$ scales independently from other factors.

3.2.3.4 Transmission range error

All protocols considered in our evaluation depend on accurately estimating which vehicles are within the sender's transmission range in order to distribute time slots among vehicles efficiently. As discussed in Section 3.2.2.3, due to a certain error probability in the wireless communication and inaccurate positioning estimation (GPS), the transmission range might be either underestimated or overestimated. Thus, we study the effects of such errors on the performance of each protocol. With an outage probability of 2%, the central point zero in the x-axis represents an accurate estimation of the transmission

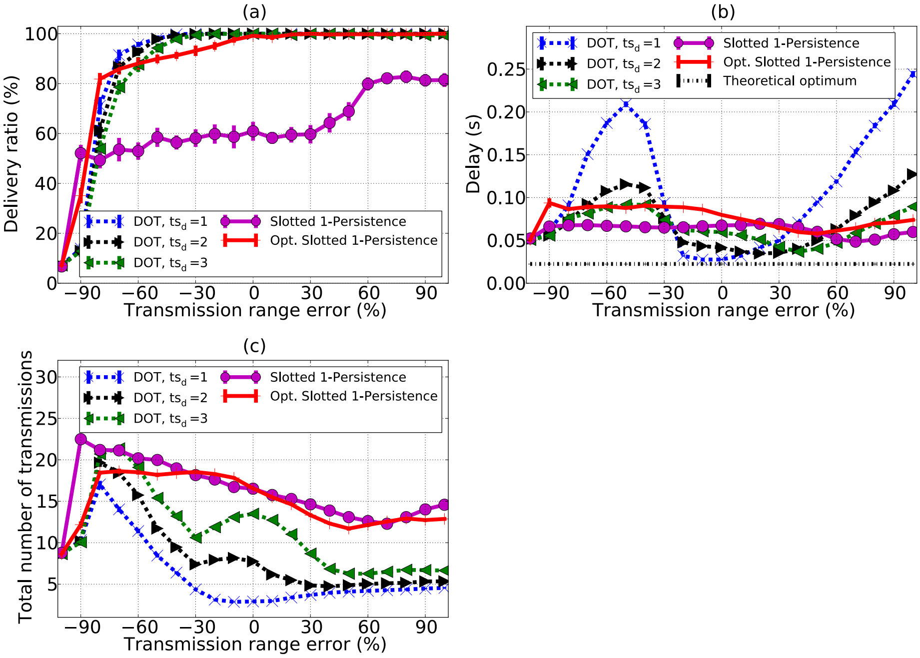


Figure 3.21: Results with 95% confidence intervals for increasing underestimation and overestimation errors in the transmission range estimation

range, which is approximately 150 meters. Negative and positive values in the x-axis are underestimated and overestimated percentage values with respect to point zero, respectively. Other parameters are fixed as shown in Table 3.2. We additionally consider results of running DOT with $ts_d = 2$ and $ts_d = 3$.

Figure 3.21(a) shows the performance of protocols with respect to the delivery ratio. For all protocols, an inaccurate transmission range estimation may result in vehicles being assigned to a sub-optimal time slot. Nevertheless, every vehicle still schedules a rebroadcast, which helps prevent the dissemination of messages from being stopped. When the transmission range is underestimated, time slots are mapped to smaller geographical regions. However, vehicles positioned beyond the underestimated range still receive and rebroadcast messages. This leads to a high level of transmission redundancy in a single time slot and, thus, to a lower delivery ratio down to 5% when the complete range is underestimated.

The results with respect to the end-to-end delay are shown in Figure 3.21(b). For protocols relying on fixed time slots such as Optimized Slotted 1-Persistence and Slotted 1-Persistence, changing the boundary of the time slots does not considerably affect the expected end-to-end delay. With a density of 50 vehicles/km/lane used in this scenario, the chance that at least one vehicle is positioned in the geographical region mapping the earliest time slot is high. One variation that can be observed in these protocols is with regard to the number of transmissions (Figure 3.21(c)). With underestimated range values, more vehicles positioned beyond the underestimated range are assigned to the earliest time slot, thereby resulting in more transmissions.

In contrast, inaccurate range estimations directly affect the expected end-to-end delay in DOT. As discussed in Section 3.2.2.3, to prevent an increase in number of transmissions when the transmission range in underestimated, vehicles positioned beyond the estimated range border are placed in the back of the sorted list $\vec{v}$. This results in increasing the end-to-end delay, as all vehicles will rely on such an underestimated range and, thus, more hops will be needed for a message to be fully disseminated along the road. For underestimated values higher than 60%, the end-to-end delay starts to decrease as a consequence of the lower delivery ratio present in this range for all protocols. On the other hand, higher delay values can also be expected with an overestimation of the transmission range, since vehicles may unnecessarily expect other vehicles farther in the message direction to rebroadcast. When more vehicles are assigned to a single time slot, i.e., $ts_d > 1$, both effects can be minimized as shown in Figure 3.21(b). This is explained by the higher chance that an inaccurate estimation is compensated by another vehicle also assigned to the same time slot but positioned farther or nearer the sender. Although the number of transmissions also increases with higher underestimated ranges, the values achieved are considerably lower when compared with Optimized Slotted 1-Persistence and Slotted 1-Persistence as shown in Figure 3.21(c).

Results show that overestimating values for the transmission range is less harmful for all protocols with regard to delivery ratio and number of transmissions. For all levels of estimation errors, DOT presents better performance results with regard to delivery ratio and number of transmissions. Despite the effects of inaccurate range estimations, DOT still presents lower end-to-end delay values compared with Optimized Slotted 1-Persistence and Slotted 1-Persistence considering a range of error up to 30%. Nevertheless, these effects are minimized when higher time slot density values are allowed.

3.2.4 Conclusion

In this section, we have presented a broadcast suppression scheme that is scalable to diverse network densities. We addressed major problems in current delay-based techniques and designed the **D**istributed **O**ptimized **T**ime (DOT) slot scheme. By exploiting the presence of 1-hop neighbor information contained in periodic safety *beacons*, DOT is capable of controlling with high precision the density of vehicles within each time slot. By means of simulations, we showed that DOT is scalable to increasing network densities, achieves near optimum delay results, and is robust to errors caused by possible inaccurate transmission range estimations. Furthermore, DOT outperformed other delay-based schemes in diverse network densities.

3.3 A scalable protocol for both highway and urban scenarios

3.3.1 Introduction

As motivated in Chapter 2, despite numerous suppression techniques to cope with dense networks and store-carry-forward models to cope with sparse networks, most related works focus on either highway or urban scenarios, but not both. On the one hand, highways are most commonly addressed with a single directional dissemination, as the data generated is assumed to only affect vehicles in one road direction, e.g., upon the event of an accident. However, such an assumption is not valid in urban scenarios, where a complex road grid with multiple road directions must be considered when relaying data messages. On the other hand, protocols designed specifically for urban scenarios usually concentrate on methods for selecting vehicles to perform the store-carry-forward task or rely on infrastructure to support the data dissemination. Nevertheless, in both types of scenarios, protocols still rely on suppression techniques that are not optimal for multi-directional dissemination.

In this section, we present the infrastructure-less **A**daptive **M**ulti-directional data **D**issemination (AMD) protocol that works seamlessly in both highway and urban scenarios. Our key contributions can be summarized as follows:

- We propose a store-carry-forward algorithm to support multi-directional data dissemination. To this end, we borrow concepts first introduced in our method for a single directional dissemination presented in Section 3.1.

- We present a generalized time slot scheme based on directional sectors to support multi-directional data dissemination. In each sector, the density of time slots is precisely controlled based on our method for single directional dissemination presented in Section 3.2.

- We perform a comprehensive simulation campaign with a direct comparison against three state-of-the-art protocols, namely, DV-CAST [47], SRD [22] (Section 3.1), and UV-CAST [51], under both realistic highway and urban scenarios. In particular, we take a real map fragment from the Manhattan area in New York City, USA, including the shape of buildings that are used to model radio obstacles.

3.3.2 Adaptive multi-directional data dissemination

In order to work seamlessly in both highway and urban scenarios, AMD incorporates the following aspects:

- ***Adaptive multi-directional dissemination***: to achieve an efficient data dissemination, each data message is simultaneously disseminated to multiple directions that are adaptively adjusted according to the local map of the road provided, for example, by a GPS navigation system. In highway scenarios, this usually means disseminating a message to both directions of the road, whereas in urban scenarios a message is disseminated towards all possible directions in the road grid. For instance, a Manhattan-like grid would have four possible directions in a region comprising an intersection.
- ***Time slot density control***: to cope with *dense* networks, we propose a time slot suppression scheme, where the final goal is to select only the farthest vehicles in each direction considered for dissemination. This time slot assignment is done by following our solution previously presented in Section 3.2, where we exploit positioning information of 1-hop neighbors to control with precision the time slots' *density*. Since the suppression of rebroadcasts is done separately for each possible direction, we guarantee a proper dissemination to all directions and prevent situations where the dissemination is hindered due to an early broadcast suppression, as shown previously in Figure 2.3.
- ***Store-carry-forward***: to cope with disconnected *sparse* networks, vehicles that are furthest away in one of the dissemination directions assume the responsibility of carrying, storing, and rebroadcasting the messages received forward to new vehicles that are encountered.

3.3.2.1 Concept definitions

To better understand the protocol, we define the following concepts which are used throughout the remaining sections:

Definition 1 (*Directional Sector*). The directional sectors of a vehicle are defined as the virtual geographical sectors within the vehicle's transmission range to which a data message must be disseminated. Each vehicle automatically adjusts its number of directional sectors according to: (i) the current local road map, e.g., two-directional highway or road intersection with four or more directions; and (ii) whether there are vehicles present in each of these possible

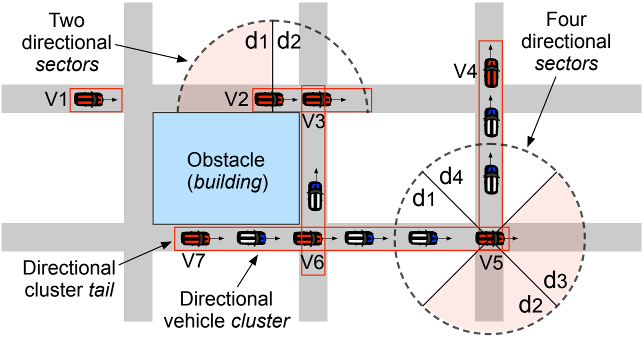


Figure 3.22: Protocol concepts applied in an urban enrironment

directional sectors. The second condition serves to prevent unnecessary divisions with empty sectors, e.g., in an intersection where no vehicles are present in one of the two crossing roads.

Definition 2 (*Directional Vehicle Cluster*). Given a directional sector, a directional vehicle cluster is defined as the group of vehicles with multi-hop connectivity that are positioned farther in the direction of the sector considered.

Definition 3 (*Directional Cluster Tail*). Given a directional vehicle cluster, the directional cluster tail is defined as the vehicle within the cluster with no radio connectivity with other vehicles positioned farther in the direction considered. Since the dissemination is multi-directional, a vehicle might be the cluster tail of multiple directional sectors simultaneously.

Figure 3.22 shows an example of how these concepts are applied. For the sake of simplicity, we limit to show the directional sectors for vehicles v_2 and v_5 only. Even though vehicle v_2 is close to an intersection, it divides its transmission range into only two sectors. This is due to the building that serves as radio obstacle and, consequently, v_2 can only detect vehicle v_3 as a neighbor, which resembles a two-direction road as normally occurs in a highway scenario. In contrast, v_5 has four directional sectors as it is in an intersection point and has neighbors positioned in orthogonal directional sectors. Each directional cluster is highlighted with a surrounding rectangle whereas cluster tails are indicated with a vehicle number, namely, vehicles v_1, v_2, v_3, v_4, v_5, v_6, and v_7. We can observe that vehicle v_2 is the cluster tail of sector d_1 in contrast to vehicle v_5 being the tail of both sectors d_2 and d_3.

3.3.2.2 Requirements and assumptions

Similarly to what has been defined as requirements in Section 3.2.2.1, AMD requires the presence of periodic *beacons* that are continually transmitted by each vehicle at a certain rate. However, AMD additionally requires that vehicles include a message list in their *beacons*, containing their last k data messages received. This serves to prevent loops in the network, i.e., a continuous rebroadcasting to new vehicles encountered that already received the data message being disseminated.

In order to accurately define the directions of dissemination, we assume that a vehicle is equipped with a device that provides road mapping information, such as a GPS navigation system. In this way, a vehicle can identify the correct number of directions in its local road context, for example, if it is an intersection in an urban setting or a highway. In addition, such mapping information also serves to identify the boundaries of the region that a message is related to, which is assumed to be defined by the application when a message is generated.

3.3.2.3 Time slot scheme

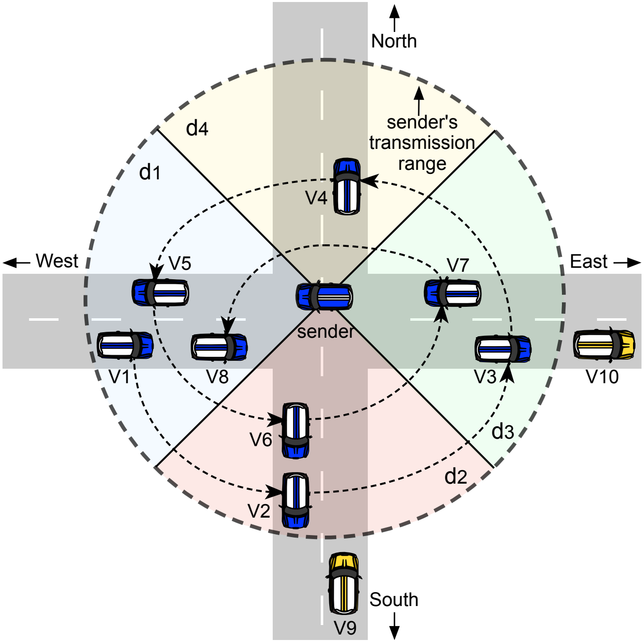
By gathering the information contained in *beacons*, each vehicle keeps a table of one-hop neighbors T_n containing the latest information about the vicinity. Each entry in T_n contains the following information: <*Vehicle ID, Expiration Time, Vehicle's Geographical Coordinates, Message List*>. The Expiration Time field is used to remove vehicles from the table that are no longer in the vicinity. Since there may be failures (e.g., collisions) when sending these *beacons*, we introduce a time tolerance before removing an entry defined as $t_t = 2.5(\frac{1}{b_f})$, where b_f is the beaconing rate, e.g., 10 Hz. This accounts for failure in one beaconing period plus possible extra delay. The message list keeps track of the k last messages received by each neighbor.

The time slot scheme works as follows. Let i be the vehicle sender of message m, and D be the set of directional sectors to which m must be disseminated. In addition, let R be the set of vehicles receiving m and $R_d \in R$ be the sub-set of vehicles receiving m within directional sector $d \in D$. Every vehicle $j \in R$ receiving m for the first time schedules a rebroadcast for m with a time delay T_{Sij}. Whenever a vehicle $j \in R_d$ receives an echo of m before T_{Sij} expires from another vehicle $k \in R_d$ that is farther in the directional sector d, it cancels (suppresses) its rebroadcast. Otherwise, the rebroadcast is performed when T_{Sij} expires.

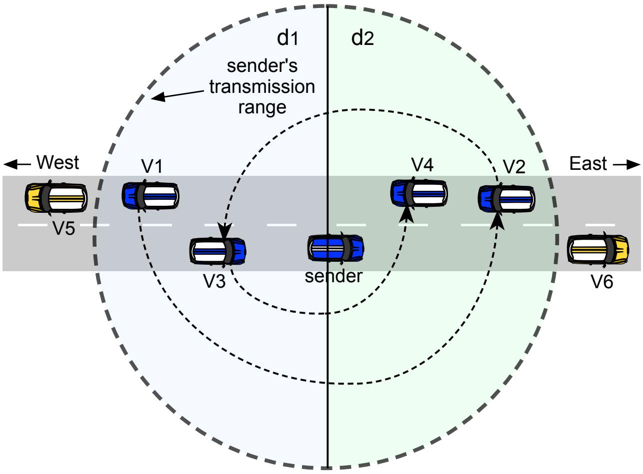
The process of defining T_{Sij} involves two tasks performed by the sender before transmitting m. The first task involves estimating which vehicles in the neighborhood will receive m, i.e., belong to set R. This is achieved by using the power level used to send message m, which allows to estimate the distance that m will travel and, thus, which neighbors in T_n will be reached by m. The second task involves defining the order in terms of priority that each vehicle in R should attempt to rebroadcast m. For this purpose, we use the common criteria of assigning a higher priority to the most distant vehicles relatively to the sender. However, in order to give equal importance to each directional sector $d \in D$ considered, the final order of rebroadcasts is defined in a round-robin fashion where the farthest vehicle in directional sector d_1 transmits first, followed by the farthest vehicle in d_2, and so forth. The final order is stored in list $\vec{v}$ and included in message m. In case different vehicles are equally distant from the sender in a single directional sector, they are then additionally sorted by their vehicle ID, where lower ID values are placed in front positions in $\vec{v}$.

Figure 3.23(a) and 3.23(b) exemplify this sorting algorithm for both an urban and highway scenarios. In Figure 3.23(a), the transmission range of the sender is divided into four directional sectors, since there are vehicles positioned in each possible directional sector in the intersection. The final order of transmission is defined as $\vec{v} =< v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8 >$. Vehicles v_9 and v_{10} are not included, since they are out of the estimated set R. In this way, the farthest vehicles in each directional sector have the highest priority. The same pattern is shown for a highway scenario in Figure 3.23(b). The difference lies in dividing the transmission range into only two directional sectors.

The decision of centralizing both tasks in the sender contrasts with our approach in Section 3.2, where we presented a distributed sorting algorithm for a single directional dissemination. However, when considering a multi-directional dissemination, the use of a centralized decision is paramount to cope with the hidden terminal problem, as motivated in [43]. Figure 3.24 shows the same highway scenario previously shown in Figure 3.23(b) but now immediately later in time after all vehicle have already received the message from the sender. If the farthest vehicles v_1 and v_2 were to estimate in a distributed fashion which other neighbors also received the message from the sender, they would clearly not include each other in set R, as they are out of range. This would result in both vehicles rebroadcasting simultaneously, thereby leading to a collision in the sender. With no echo correctly received, the sender would in turn assume that its previous broadcast failed and the same message would be broadcast once again. To prevent such collisions, the priority list $\vec{v}$ as estimated by the sender is included in m to guarantee a consistent assignment



(a) Time slot scheme in an urban scenario



(b) Time slot scheme in a highway scenario

Figure 3.23: The multi-directional time slot scheme

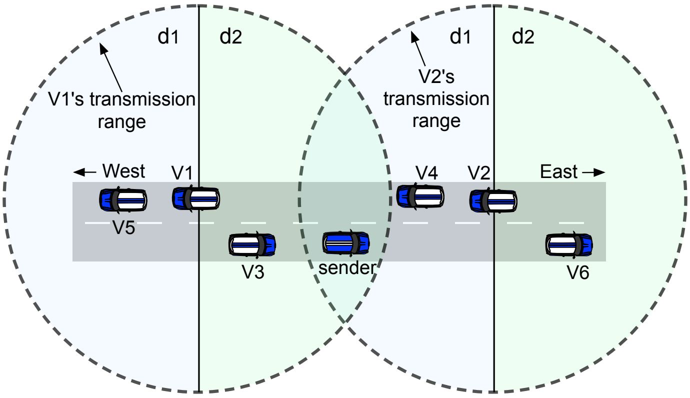


Figure 3.24: The hidden terminal problem when relying on a distributed sorting algorithm

of priority among all receiving neighbors. This comes at the cost of an extra overhead but only in the data message, thereby not including this information in periodic *beacons*.

Upon receiving message m, each vehicle $j \in R$ finds its own position in the received $\vec{v}$. We denote this position as $S_{ij} \in [0, n-1]$, where n is the total number of elements in $\vec{v}$. Just as defined in the DOT scheme, each vehicle calculates the waiting time before rebroadcasting T_{Sij} as follows:

$$T_{Sij} = st\left(\left\lceil\frac{(S_{ij}+1)}{ts_d}\right\rceil - 1\right) + AD_{ij}. \tag{3.17}$$

With Equation 3.17, each vehicle is assigned to a time slot that is proportional to its priority in the neighborhood, where high priorities are translated into early time slots. Vehicles with lower priority can cancel their rebroadcasts as soon as they hear an earlier transmission of the same message scheduled. However, such suppression is only done if the echo was sent by another vehicle residing in the same directional sector, as previously mentioned. With these measures, we guarantee the message dissemination in each possible direction while minimizing the delay and number of transmissions.

Also analogous to the DOT scheme, the main parameter ts_d determines the number of vehicles that are allowed to be assigned simultaneously to a single time slot. In other words, this parameter enables the control of time

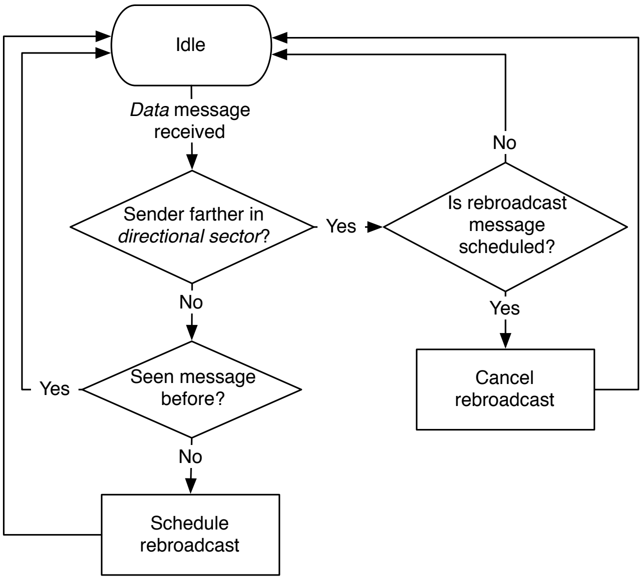


Figure 3.25: Time slot scheme used by AMD

slots' density. To avoid to nearly simultaneous broadcasts in a single time slot, we introduce an additional delay AD_{ij} defined as:

$$AD_{ij} = d\left(S_{ij} \bmod ts_d\right), \tag{3.18}$$

where d is a time delay sufficiently long for vehicles assigned to the same time slot to sense if other vehicle has already started its transmission and sufficiently low not to overlap with the beginning of later time slots, i.e., $d \ll st$.

With an accurate estimation of set R, optimal results in terms of end-to-end delay are achieved when ts_d matches the number of directional sectors: $ts_d = 4$ in Figure 3.23(a) and $ts_d = 2$ in Figure 3.23(b). This is expected since the farthest vehicles, i.e., one in each direction, rebroadcast almost immediately. As explained, their transmissions are separated in time only by the additional delay AD_{ij}. This may come at the cost of a lower delivery ratio and higher number of transmissions due to possible negative effects of the hidden terminal problem.

The complete broadcast suppression scheme is shown in Figure 3.25. Whenever a vehicle receives a data message, it first checks whether this message comes from a vehicle that is farther in the directional sector defined by the previous sender. The goal is to identify if the vehicle receiving the message is situated in between the current and previous sender, which would indicate that this message has already been disseminated in this geographical region and that a rebroadcast scheduled can be safely canceled (suppressed). In this way, we guarantee that only transmissions scheduled in directional sectors already covered by the message are cancelled. This verification is possible, since we include positioning and directional sector information of the previous sender in every data message rebroadcast, as we elaborate later in this section. If this verification returns false, it means that the receiving vehicle is farther in the directional sector of the previous sender and can schedule a rebroadcast at time T_{Sij} with respect to directional sector of the current sender, if the message has been received for the first time.

3.3.2.4 Dealing with estimation errors

As discussed previously, our time slot scheme depends on accurately estimating which vehicles are within the transmission range of the sender, i.e., belong to set R. In Section 3.2, we have elaborated on positioning and transmission range estimation and showed by means of simulation that our approach is robust against errors for different time slot density values of ts_d. Therefore, we take similar measures to cope with both underestimated and overestimated transmission range values. Specifically, we introduce the following policy to cope with underestimated values. If a vehicle j is beyond the range estimated, it is assigned to the last position in list $\vec{v}$. If $\vec{v}$ is empty, j transmits immediately after a random small delay taken from the interval $[0, d]$. This policy may increase the end-to-end delay but it maintains the protocol robust against collisions and contention. On the other hand, we tackle overestimated values by being conservative when assuming the maximum distance from the sender that neighbors are still able to receive a message.

Another potential source of error lies in the estimation of the number of directional sectors whenever the required mapping information (Section 3.3.2.2) is inaccurate or unavailable. Inaccurate estimates of this value may lead to sub-optimal performance. On the one hand, if an excessive high number of sectors is employed, vehicles are assigned to sectors that do not represent any real road, thereby unnecessarily increasing the number of transmissions. On the other hand, choosing an excessive low number of sectors may lead to the suppression of transmissions of vehicles driving in potential road directions of dissemination, thereby causing higher delays and lower delivery ratio. We elaborate further on these consequences in Section 3.3.3.1.

Finally, although mapping information is assumed to be potentially available, 3D shapes of buildings are generally unknown a priori by most navigation systems. Therefore, we do not consider obstacles in the calculation of the rebroadcast priority in the neighborhood. However, as highlighted in Figure 3.22, obstacles will also hinder the reception of beacons sent by vehicles which are directly blocked by them, e.g., vehicles behind buildings. The consequence is that only neighbors previously detected are considered, thereby minimizing estimation inaccuracies when shadowing is present.

3.3.2.5 The protocol

With our proposed time slot scheme, selected vehicles are chosen to rebroadcast whenever new messages are received. In this way, messages are immediately disseminated throughout the network to every possible road direction. However, such a scheme still depends on additional measures to cope with disconnected networks when the transmission range does not reach farther vehicles in the each directional sector.

To cope with radio gaps in the network, we rely on a store-carry-forward approach that is based on our previous single directional dissemination scheme named **S**imple and **R**obust **D**issemination (SRD) protocol, presented in Section 3.1. The general idea lies in assigning the responsibility of storing, carrying, and forwarding to vehicles located at the tail of a directional cluster, since these vehicles have the highest probability of meeting later other vehicles farther in the cluster direction. As we exemplified in Figure 3.22, a vehicle is the tail of a directional cluster if there is no other vehicle farther in that direction. A vehicle can in fact be the tail of multiple directional clusters simultaneously, for example, when a vehicle divides its transmission range into four sectors in an intersection as it occurs with v_5 in Figure 3.22.

The complete AMD protocol combines both our proposed time slot broadcast suppression and store-carry-forward schemes, as shown in Figure 3.26. Every vehicle updates its local neighborhood information T_n with the content received from either a *beacon* or a data message. When a data message is received, our time slot scheme is executed as defined by the diagram in Figure 3.25. On the other side of the diagram, *beacons* are used to update the *tail* status of the receiving vehicle for each of its directional sectors. When a vehicle makes the transition from *tail* to *non-tail* in one of its directional sectors, it is an indication that there is now connectivity to farther vehicles in that direction

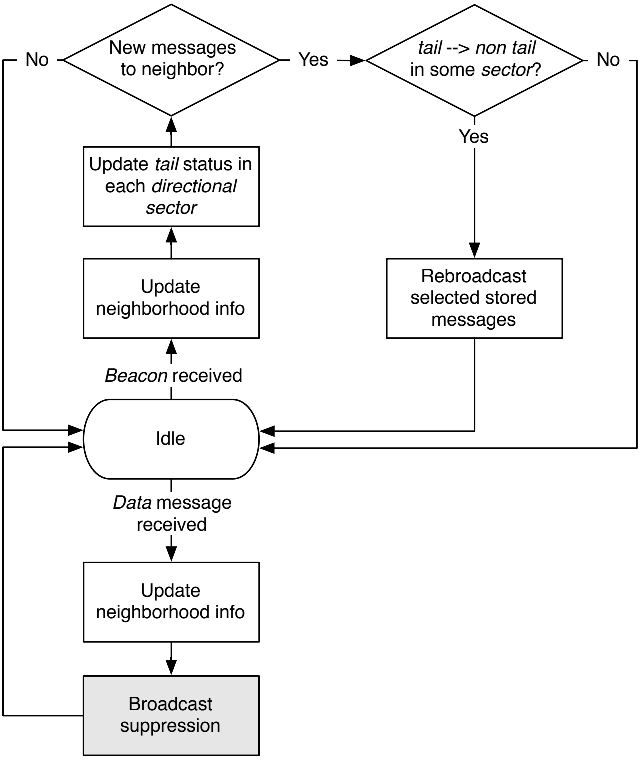


Figure 3.26: The complete AMD protocol diagram

and that previously stored messages can be relayed. To prevent unnecessary rebroadcasts, the message list received in the neighbor's *beacon* is examined and only messages not yet received by the neighbor are rebroadcast.

3.3.2.6 Defining directional sectors

Dividing the transmission range into directional sectors is a crucial task done by the sender in order to determine the rebroadcast priority of the receiving neighbors. Such division is achieved by means of the reference vector $\vec{a}$ and the total number of sectors b, where b is defined according to the total number of road directions in the region nearby the sender and whether vehicles in the neighborhood have been detected (via beacons) in each road direction.

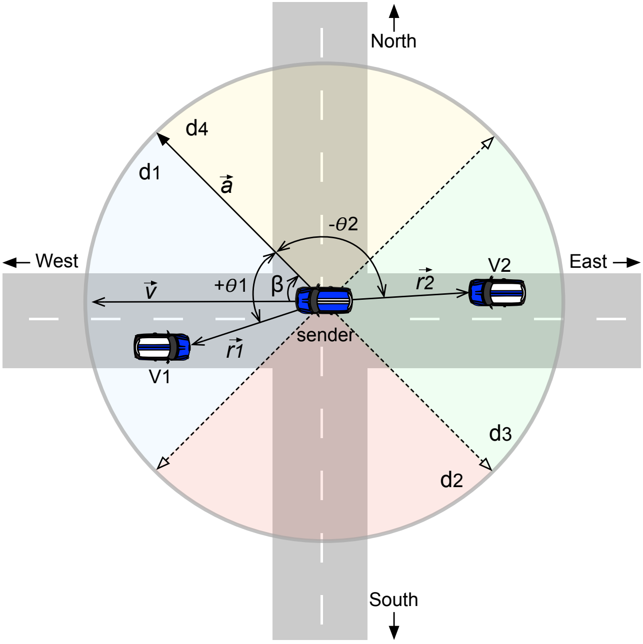


Figure 3.27: Example of how directional sectors are defined

Figure 3.27 shows how such division is done. The sender uses its previous and current geographical positions to establish its velocity vector $\vec{v}$. Depending on the number of directional sectors considered b, $\vec{v}$ is rotated in β degrees to maximize the road area covered by the sector. We consider a rotation defined as $\beta = 360/2b$, although more appropriate rotation formulas may be considered when more complex road shapes are present. The directional sector that each receiving neighbor belongs is the defined by the angle between the rotated vector $\vec{a}$ and the direction vector $\vec{r}$ with respect to the sender's position. In this example, four directional sectors are considered in the intersection, which yields a rotation of $\beta = 45$ degrees. Vehicles v_1 and v_2 have an angle of θ_1 and θ_2 between $\vec{a}$ and their vector with respect to the sender $\vec{r}_1$ and $\vec{r}_2$, respectively. By convention, we define that the index number of directional sectors increases anti-clock wise as in the regular unit trigonometric circle.

3.3.2.7 Message structure

Both data messages and *beacons* have vehicle and message IDs to enable vehicles to distinguish different broadcast messages. An example of vehicle ID is the MAC address, while the message ID can either be a sequence number or a timestamp of the message generation time.

The complete data message structure comprises the following information: *<Vehicle ID, Message ID, Vehicle's Geographical Coordinates, Time Stamp, Event's Geographical Coordinates, Priority List* $\vec{v}$*, Previous Direction Reference Vector* $\vec{a}_p$*, Previous Number of Sectors* b_p*>*. The *Time Stamp* and *Event's Geographical Coordinates* fields are used to set the validity for the message in terms of time and distance, respectively. This prevents both the circulation of old messages and that messages travel beyond the boundaries defined by the application. As explained previously, in order for vehicles to suppress their scheduled rebroadcasts correctly, they must know whether an echo of a message comes from a vehicle that is farther in the directional sector defined by the previous sender. For this purpose, every data message also includes the directional sector data of the previous sender, namely, *Previous Direction Reference Vector* $\vec{a}_p$ and *Previous Number of Sectors* b_p fields.

The complete structure of *beacons* is defined as previously motivated in our requirements section: *<Vehicle ID, Message ID, Vehicle's Geographical Coordinates, Message List>*.

3.3.3 Performance evaluation

The performance evaluation of AMD is carried out by means of simulations. Our goal is to study the scalability of AMD under both highway and urban realistic scenarios. We select three state-of-the-art protocols for comparison, namely:

- ***DV-CAST***: it is a protocol designed to cope with both sparse and dense networks in *highways* [47]. It uses one of the three suppression techniques proposed in [32]. In particular, we set DV-CAST to use the Slotted 1-Persistence suppression technique, which is the mechanism that has shown to achieve best performance in terms of end-to-end delay.

- ***SRD***: it is a protocol that we previously proposed for *highway* scenarios in Section 3.1. Just as with DV-CAST, it combines both a store-carry-forward approach and suppression technique to tackle disconnected and dense networks, respectively. Its suppression technique, Optimized Slotted 1-Persistence, relies on an optimized version of the Slotted 1-Persistence suppression method to prevent nearly simultaneous rebroadcasts in a single time slot in dense networks.

- ***UV-CAST***: it is a protocol that specifically addresses *urban* scenarios with zero infrastructure support [51]. It combines: (i) a suppression technique for dense networks that gives higher priority to vehicles near intersection points; (ii) and a gift-wrapping algorithm to select vehicles to store, carry, and forward messages.

We do not include DOT directly in the comparison, since most of its functioning is incorporated by AMD. In fact, despite the differences regarding where the sorting procedure of vehicles in the neighborhood is done, i.e., centralized with AMD and distributed with DOT, early results show that the performance of the two protocols is analogous in highway scenarios, which is the only applicable scenario for the DOT scheme.

In our simulations, we utilize the MiXiM Framework [81] and adjust the available implementation of the IEEE 802.11b protocol to comply with basic specifications of the 802.11p version. Table 3.3 contains a summary of the simulation parameters. In the MAC layer, we set the bit rate to 6 Mbit/s, the Contention Window (CW) to values between 15 and 1023, the slot time to 13 μs, the SIFS to 32 μs, and the DIFS to 58 μs. In the physical layer, we operate on the 5.88 GHz frequency band, with 10 MHz of bandwidth. We set the transmission power to 300 mW to achieve approximately 230 meters of communication range. The bit error rate (BER) model used is the one provided by the Veins project [90], which is based on measurements from [91] for the 6 Mbit/s bitrate. We use the Friis Free Space Path Loss (FSPL) propagation model with exponent α equal to 3.0, as it is within the range 2.7 to 5, estimated for outdoor shadowed urban areas in [89]. We include shadowing effects that are modeled following a log-normal distribution with zero mean and standard deviation $\sigma = 6.25$ dB, as it is within the range 4 to 12 dB for outdoor propagation conditions according to [89]. Finally, we use the shadowing obstacle model proposed in [92] to simulate obstacles caused by the presence of buildings in urban scenarios.

For all suppression mechanisms, we set the slot time st to 5 ms. We define the total number of time slots for Slotted 1-Persistence used by DV-CAST NS_{std} to 3 and for Optimized Slotted 1-Persistence used by SRD we set NS_{opt} to 6 (3 slots for each road direction as defined in Section 3.1). The value chosen for Slotted 1-Persistence is based on simulation parameters used in [47]. The maximum additional delay D_{max} used by Optimized Slotted 1-Persistence is

Table 3.3: Simulation parameters

Physical layer	Frequency band	5.88 GHz
	Bandwidth	10 MHz
	Transmission range	$\sim$230 m
	FSPL exponent α	3.0
	Log-normal σ	6.25 dB
	Obstacle model	Defined in [92]
	Receiver sensitivity	-119.5 dBm
	Thermal noise	-110 dBm
	Bit Error Rate (BER)	Based on [93]
Link layer	Bit rate	6 Mbit/s
	CW	[15,1023]
	Slot time	13 μs
	SIFS	32 μs
	DIFS	58 μs
Suppression mechanisms	st	5 ms
	ts_d	1
	d	DIFS
	NS_{std}	3
	NS_{opt}	6
	D_{max}	1 ms
	τ_{max}	500 ms
	Beacon frequency	1 Hz
	Beacon size	$\geq$ 24 Bytes
	Message list's k	25
Scenarios	Data message size	2312 bytes
	Data message freq.	0.5 Hz
	# Runs	20

set to 1 ms. For our suppression mechanism we set the time slot density ts_d to 1 and additional delay d to DIFS. For UV-CAST, we set the maximum waiting time parameter τ_{max} to 500 ms, as suggested in [51]. Finally, we also map all the intersection points in our urban scenario to allow for a higher priority broadcast by vehicles near intersections, as required by UV-CAST.

For all simulation scenarios the data message size is 2312 bytes large, the maximum allowed by the 802.11p standard. Data messages are generated every 2 seconds, i.e., message frequency of 0.5 Hz. The size of *beacons* can vary from 24 bytes to the maximum message size depending on the message list

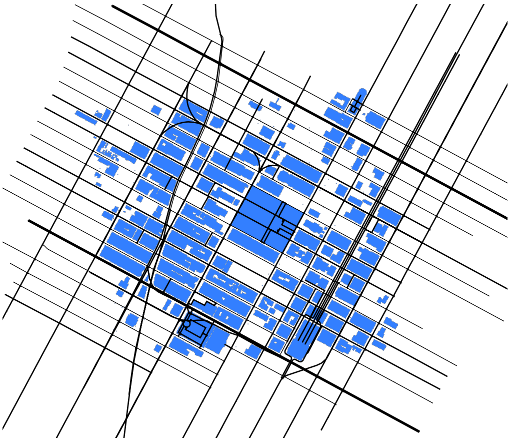


Figure 3.28: Urban scenario: map fragment of Manhattan, New York City, USA

included. We consider that each new entry in the message list is 12 bytes large, thereby leading to final *beacon* size of $s(w) = 12w + 24$ bytes, where w is the number of entries in the list. Since we limit the total number of entries to $k = 25$, the maximum size that each *beacon* can have in our simulations is limited to 324 bytes. Such limit is chosen based on the proper balance achieved between the number of unnecessary transmissions avoided due to loops in the network and beacon size in the scenarios considered in our simulations. However, further analysis is required to determine the most appropriate value for a wider variety of scenarios. *Beacons* are sent at the frequency of 1 Hz, which gives the worst case scenario in terms of freshness of the one-hop neighborhood information. Furthermore, varying the beaconing rate in our experiments has not led to significant changes in our simulation results, except for more message collisions.

We consider one highway scenario and one urban scenario. The highway consists of a 1-kilometer straight road with two lanes in each road direction. Each message is generated by one fixed vehicle positioned in one end of the road and gathered by another fixed vehicle in the other end of road. For this scenario, in total 20 runs of 100 seconds are executed. As urban scenario, we select a map fragment from Manhattan, New York City, USA. This segment has an area of 1.5 x 2 km^2 and was retrieved with OpenStreetMaps [94]. Messages are generated by one fixed vehicle in the center of the map and gathered by one of the four fixed vehicles that are positioned in each corner of the map.

Figure 3.28 shows the complete map fragment considered, where buildings represented by dark rectangles serve as radio obstacles. Simulations for this urban scenario consist of 20 runs of 300 seconds.

Both scenarios were created with SUMO [83]. Therefore, they includes realistic mobility patterns such as vehicle overtaking, lane changing, and relies on the well-known car-following mobility model. Vehicles' speeds vary according to the density considered by following the Krauß mobility model, i.e., the higher the density is, the slower vehicles move.

Our evaluation considers the following metrics:

- ***Delivery ratio***: the percentage of data messages generated that fully propagate the scenario considered until they are received by one of the fixed vehicle responsible to gather data messages. Ideally, dissemination protocols must achieve a delivery ratio percentage close to 100% in dense networks.

- ***Delay***: the total time taken for a data message generated to fully propagate the scenario considered until it is received by one of the vehicles responsible to gather data messages. This is particularly important for critical safety messages that must be disseminated as quickly as possible. We additionally compare the performance of each protocol with a theoretical optimum which serves as lower bound. This value is simply calculated as the minimum number of hops that a message must travel times the transmission delay, given the transmission range employed. We limit this estimation of the theoretical optimum to our highway scenario, since each message has a clear straight trajectory to travel, which leads to a predictable optimum end-to-end delay. The same does not occur for urban scenarios, due to its complexity in terms of multiple possible trajectories, mobility of vehicles and radio obstacles.

- ***Total number of transmissions***: the total number of transmissions performed on average by an arbitrary vehicle. We consider only data messages in these results, thereby excluding transmissions of *beacons*. This value is normalized by the total number of vehicles in each scenario. In order to be scalable, protocols must keep a low number of transmissions during a message's dissemination.

3.3.3.1 Number of directional sectors

When not properly chosen, the total number of directional sectors can negatively affect the performance of the AMD protocol. As previously explained,

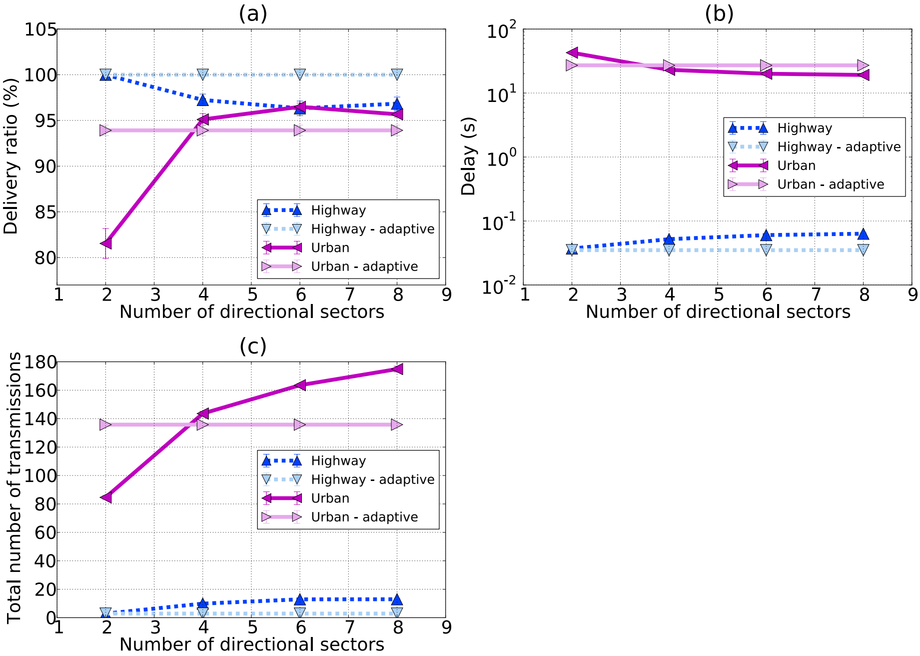


Figure 3.29: Results with 95% confidence intervals for different number of directional sectors used by AMD in highway and urban scenarios

AMD uses mapping information provided by a GPS navigation system to adaptively adjust the number of directional sectors according to the number of road directions and the presence of vehicles in the local region. In our simulations, each vehicle is pre-loaded with a simplified version of the scenario map. Such map contains the geographical positions corresponding to the center of each road intersection. Since the scenarios considered contain either straight roads (highway) or follow a Manhattan grid shape (urban scenarios), we define that the number of sectors can be either two or four. More specifically, the number of sectors is four whenever (i) the vehicle about to broadcast is within a radius of 15 meters from the center point of the nearest intersection and (ii) at least one neighboring vehicle has been previously detected via the reception of beacons in one of the orthogonal road directions relatively to the velocity vector of the sender. Otherwise, two directional sectors are employed.

In the following, we analyze the effects of varying the number of directional sectors in both highway and urban scenarios when compared to the adaptive algorithm used by AMD. Figure 3.29 shows the results for varying the total number of directional sectors from 2 to 8. Each number is fixed during the whole simulation run regardless of the number of road directions in the map. In Figures 3.29(a) and 3.29(b), we can observe that choosing a number higher than two for the highway scenario has a negative impact in the delivery ratio and delay. The same occurs for the urban scenario when fixing a number of sectors lower than four. Both results are explained by the fact that choosing two sectors for highways and four or more for urban scenarios provides a better matching to the actual road mapping. On the one hand, an excessive number of sectors leads to too many vehicles being assigned to a different sector. Since the transmission of vehicles can only be suppressed by other vehicles in the same sector, this results in a high number of transmissions and possible collisions in the network (Figure 3.29(c)). On the other hand, an excessive low number of sectors leads to an inefficient division of sectors, thereby causing higher delays and lower delivery ratio.

Overall, using mapping information to adaptively choosing the number of directional sectors provides a performance near or equal the best result achieved when fixing the number of sectors beforehand for the whole simulation.

3.3.3.2 Network density

In Figures 3.30 and 3.31, we show the results for each protocol when varying the network density. Varying the network density evaluates the protocols in terms of scalability, which is crucial in vehicular networks due to its dynamic nature. We additionally show the results for the suppression techniques used by each protocol separately in order to isolate the gains in performance when employing store-carry-forward mechanisms in very low densities.

Figure 3.30 shows the results for highway scenarios when varying the network density from 1 to 100 km/h/lane. We compare AMD with two other protocols designed specifically for highway scenarios, namely, DV-CAST and SRD. As shown in Figure 3.30(a), AMD achieves near 100% in delivery ratio for densities higher than 15 vehicles/km/lane. In contrast, DV-CAST and SRD present lower delivery ratio, especially in high densities. These protocols lack a means to control the time slots' density, thereby leading to extra rebroadcast redundancy and collisions when many vehicles are assigned to a single time slot. For lower densities, the delivery ratio is lower for all protocols because at the moment that a message is generated there are cases when no vehicle is in

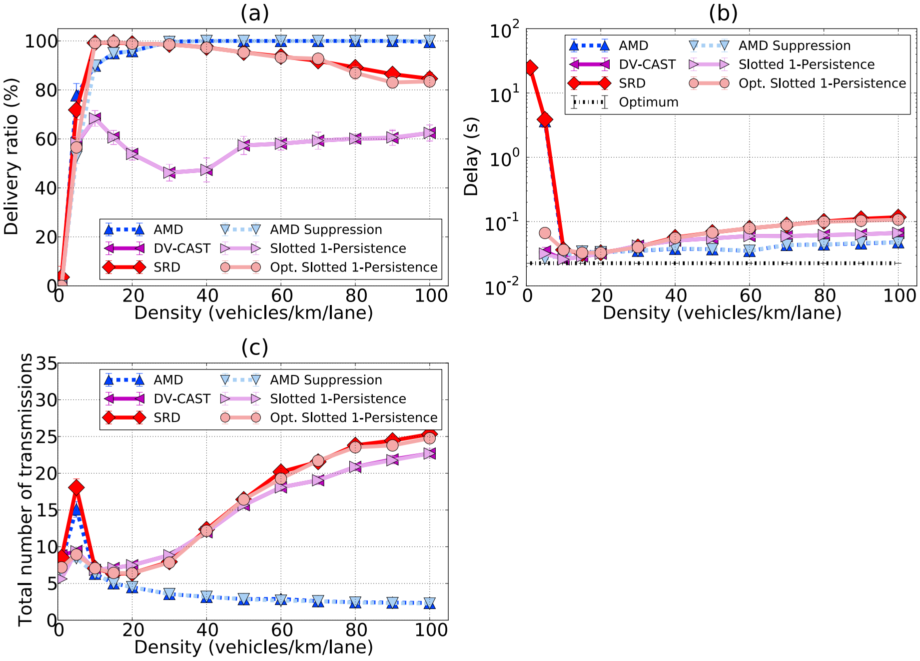


Figure 3.30: Results with 95% confidence intervals for increasing network densities in highway scenarios

neighborhood to received and disseminate the message to other vehicles in the road. Nevertheless, both AMD and SRD protocols present an improvement of near 45% in very low densities, namely, density of 5 vehicles/km/lane, compared to their suppression techniques alone.

The end-to-end delay tends to increase with density, especially for protocols that rely on a fixed number of time slots such as DV-CAST and SRD, as shown in Figure 3.30(b). The reason lies in the higher contention delay generated when more vehicles attempt to rebroadcast in a single time slot. This can be verified in Figure 3.30(c), where the total number of transmissions is shown to be significantly higher for DV-CAST and SRD. In contrast, in proportion with the total number vehicles in each density, the number of transmissions tends to decrease with AMD thanks to its control of the time slots' density.

The results for our urban scenario when varying the network density from 25 to 150 vehicles/km^2 are shown in Figure 3.31. AMD is compared against

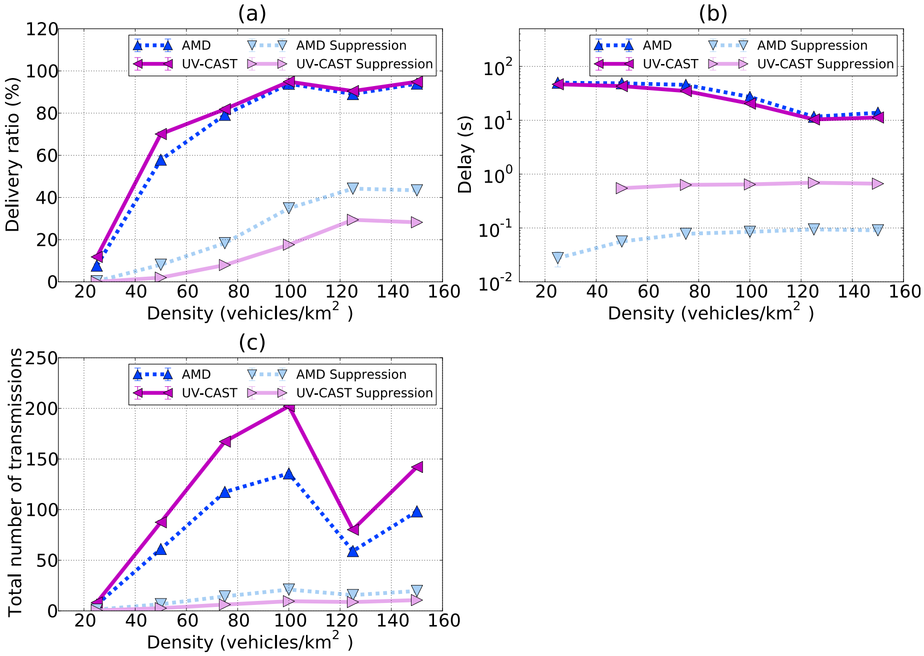


Figure 3.31: Results with 95% confidence intervals for increasing network densities in urban scenarios

UV-CAST; a protocol designed especially for urban environments. Both protocols achieve similar performance in terms of delivery ratio and end-to-end delay, as shown in Figures 3.31(a) and 3.31(b). This is explained by the fact that when protocols have to resort to using their store-carry-forward mechanisms, their performance in terms of delay and delivery ratio becomes dependent on the movement of vehicles, which is equal for both protocols. However, when verifying the performance of the suppression techniques used by each protocol alone, AMD's suppression clear outperforms the suppression used by UV-CAST in both metrics. This shows that AMD is able to quickly disseminate messages whenever there exist end-to-end connectivity to one of the fixed vehicles responsible for gathering data messages. We can observe that the suppression techniques alone present a lower delivery ratio when compared with their complete protocols. This behavior is particularly expected in urban scenarios where radio obstacles make disconnections predominant, thereby increasing the dependency on store-carry-forward strategies.

In terms of number of transmissions, AMD introduces a lower overhead in the network compared to UV-CAST, as shown in Figure 3.31(c). The reason lies in the ability of AMD to correctly select vehicles to perform the task of carrying and forwarding messages as well as in the ability of its suppression technique to separate vehicles in independent directional sectors, which allows vehicles to properly rebroadcast and suppress transmissions. We can also observe the trend of an increasing number of transmissions from densities 25 to 100 vehicles/km^2. After this point, the network becomes mostly connected and fewer transmissions are needed due to the more frequent use of each suppression technique.

In general, AMD scales more efficiently with increasing network densities when compared with protocols especially designed for either highway or urban scenarios. Compared to these solutions, AMD presents up to 7× lower number of transmissions in dense highway scenarios.

3.3.3.3 Time slot parameter

In the following, we analyze the performance of protocols when varying their main parameters, namely, the total number of time slots (used by DV-CAST and SRD), τ_{max} (used by UV-CAST) and the time slot density ts_d (used by AMD). In particular, SRD uses doubled number of time slots to distribute the number of time slots equally among the two road directions, as detailed in Section 3.1. Contrary to the other protocols, UV-CAST does not define a fixed number of time slots but rather a maximum delay τ_{max}. In this case, we define that each value in our plot assumes a value of $\tau_{max} = 0.0625\ i$, where i falls in the interval from 1 to 8 that is used in the evaluation of each protocol's time slot parameter.

Figure 3.32 shows the results when varying the time slot parameter of each protocol for highway scenarios. With regard to the delivery ratio, both SRD and DV-CAST achieve higher delivery ratio when increasing the total number of time slots, as shown in Figure 3.32(a). With more time slots, a lower number of vehicles is assigned to a single time slot. Therefore, a lower level of rebroadcast redundancy is expected and messages can travel with less interference throughout the road length. The opposite effect occurs when the time slot density is increased in AMD. Higher values for the number of time slots means more vehicles within a single time slot, which leads to a decrease in delivery ratio from $ts_d = 4$ in this scenario.

Similarly to what occurs when varying the network density, the end-to-end

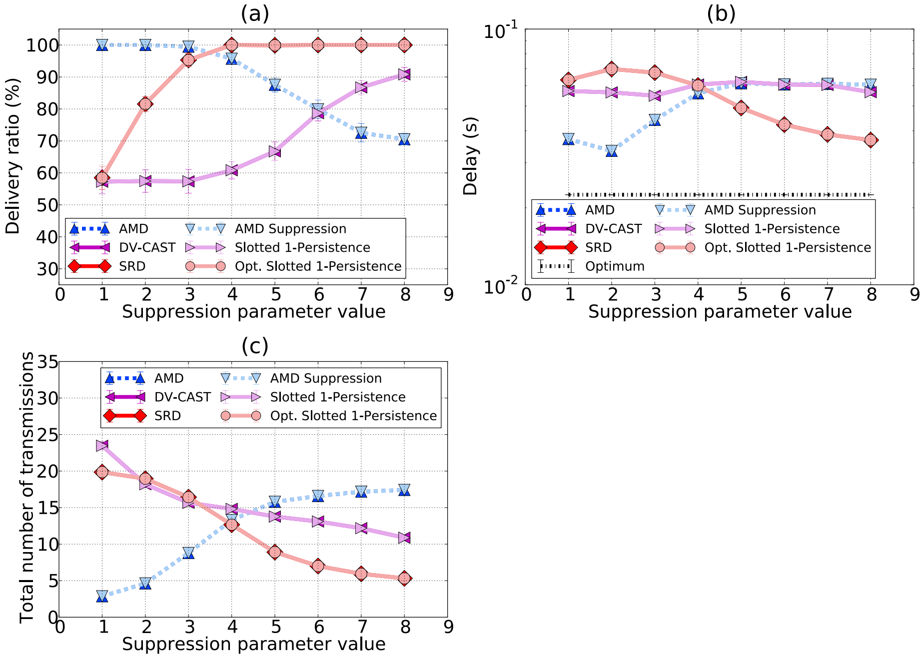


Figure 3.32: Results with 95% confidence intervals for different time slot parameters in highway scenarios

delay tends to increase when more vehicles attempt to transmit nearly simultaneously in a single time slot (Figure 3.32(b)). This occurs when decreasing the number of time slots (SRD and DV-CAST) or increasing ts_d (AMD). Such increase in the number of transmissions can be verified in Figure 3.32(c). One interesting remark is that AMD achieves its lowest delay when $ts_d = 2$, since this allows one transmission in each road direction to occur simultaneously.

With regard to our urban scenario (Figure 3.33), varying the time slot parameter for both AMD and UV-CAST shows to have little impact when considering their complete protocol with a store-carry-forward mechanism. Their performance is again dependent on the movement of vehicles, which is equal for both protocols.

When looking at each protocol's suppression technique, however, increasing ts_d in AMD results in more vehicles being assigned to a single time slot and, thus, in a lower delivery ratio (Figure 3.33(a)). Differently from the results

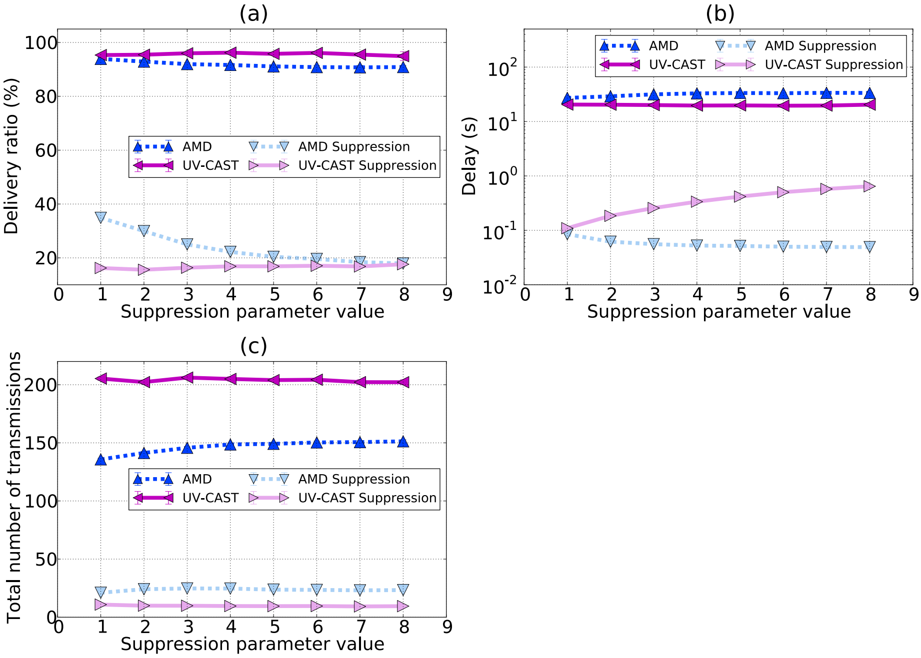


Figure 3.33: Results with 95% confidence intervals for different time slot parameters in urban scenarios

for highway scenarios, delay values are lower with higher ts_d (Figure 3.33(b)), which is a result of the better matching of the number of simultaneous transmission allowed with the multiple road directions present in more complex urban scenarios. In contrast, increasing the τ_{max} parameter in UV-CAST implicitly works as increasing the number of time slots used, since it effectively helps spreading the transmissions of vehicles in time. However, because the suppression technique used by UV-CAST is not designed for multi-directional dissemination, such increase in τ_{max} has little impact on the metrics evaluated, apart from the obvious increase in delay.

Overall, all protocols perform best when fewer vehicles attempt to transmit nearly simultaneously. AMD, in particular, presents best performance in terms of delay when the number of simultaneous transmission allowed ts_d equals the number of road directions, however, at the cost of a lower delivery ratio in urban scenarios.

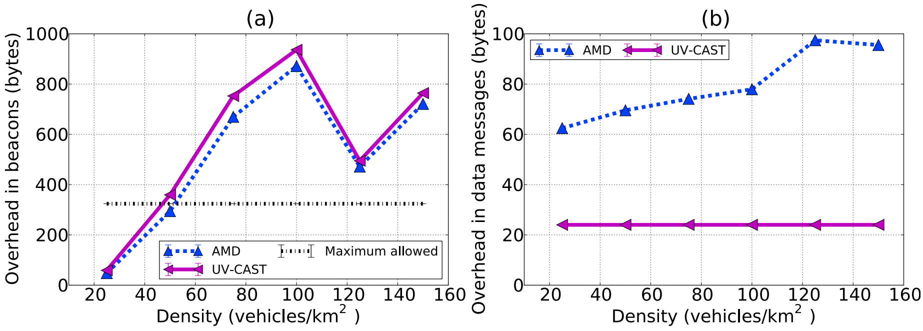


Figure 3.34: Results with 95% confidence intervals for the message overhead introduced by AMD and UV-CAST

3.3.3.4 Message overhead

All protocols considered in this evaluation require that a certain overhead is added into beacons or data messages in order to guarantee their proper functioning. Such overhead is generally translated into a fixed number of bytes which correspond to extra fields appended to either beacon or data messages. However, both AMD and UV-CAST resort to appending a message list with variable length to beacons in order to prevent that repeated messages are unnecessarily disseminated in the network. In particular, AMD also includes a small variable list in data messages to guarantee that the order of rebroadcast in the neighborhood is achieved. Therefore, in this last section, we measure the message overhead required by these two protocols when increasing network densities are considered.

Figure 3.34 shows the message overhead in number of bytes for both protocols in our urban scenario. As explained previously, the size of beacons can vary from 24 bytes to a final beacon size of $s(w) = 12w + 24$ bytes, where w is the number of entries in the list. Since we limit the total number of entries to $k = 25$, the maximum size that each beacon can have in our simulations is limited to 324 bytes. In Figure 3.34(a), we can observe that this upper bound value is reached when the network density is around 50 vehicles/km^2. Although setting an unbounded value for the number of entries k is obviously unadvisable, we additionally evaluate in this section the total overhead when the maximum list size possible is allowed for each network density. As shown in the same figure, the maximum overhead reached for each density follows a similar pattern as the number of transmissions (Figure 3.31(c)). In particular, UV-CAST presents a slightly higher overhead compared with AMD, reaching a maximum of 950 bytes.

With regard to data messages, we compare the extra variable overhead required by AMD with the fixed number of bytes used by UV-CAST. As shown in Figure 3.34(b), such overhead is much lower compared to the message list included in beacons, since it depends only the number of neighbors participating in the rebroadcast operation defined by AMD's suppression technique.

Overall, both AMD and UV-CAST protocols require additional message overhead of variable length to guarantee a proper functioning and to prevent unnecessary transmissions due to potential dissemination loops in the network. Especially for emergency applications, we expect that the message list size introduced in beacons be much lower than what has been considered here, since a single message might be repeated over time by the source vehicle, thereby reducing the number of entries of unique messages in the message list.

3.3.4 Conclusion

We have presented a data dissemination protocol that works seamlessly in both highway and urban scenarios: the **A**daptive **M**ulti-directional data **D**issemination (AMD) protocol. AMD combines a generalized time slot scheme based on directional sectors and a store-carry-forward algorithm to support multi-directional data dissemination.

By means of simulation, we showed that AMD scales properly in various network densities in both highway and urban scenarios. We considered in our simulation scenarios realistic features such as a real map fragment of the Manhattan area in New York City with buildings serving as radio obstacles. Compared with protocols especially designed for either highway or urban scenarios, namely, DV-CAST (highway), SRD (highway), and UV-CAST (urban), AMD obtained higher delivery ratio, lower end-to-end delay, and lower number of transmissions. In particular, AMD presented up to 7× lower number of transmissions in dense highway scenarios.

Table 3.4: Overview of solutions presented in the chapter

Solution	Goal	Requirements	Target scenario
SRD	Cope with disconnections and broadcast storm in single-directional dissemination	Position info	Highway
DOT	Achieve scalability in dense networks in single-directional dissemination	Position info	Highway
AMD	Deal with both disconnections and scalability issues in multi-directional dissemination	Position & mapping info	Highway & urban

3.4 Concluding remarks

In this chapter, we have proposed protocols to enable the dissemination of data related to safety applications in both highway and urban scenarios. Table 3.4 shows an overview of these solutions and their evolution in terms of goals achieved and scenario covered throughout the chapter.

In Section 3.1, we presented the SRD protocol that is able to cope with disconnected highways while preventing the negative effects of the broadcast storm problem. The use of suppression techniques has been motivated and employed in dense networks while the store-carry-forward communication model has been used in sparse networks. Our simulation results show that SRD outperforms another state-of-the-art protocol in terms of delivery ratio and introduces a lower load into the network.

In Section 3.2, we presented the DOT scheme to further address the broadcast storm problem, especially in dense networks. The main idea behind DOT stems from the fact that *beacons* provide one-hop neighborhood awareness that can be exploited to precisely assign vehicles to different time slots. Although we advocated a *suppressed hello* message mechanism in Section 3.1, further developments in standardization, e.g., the European ETSI standard [95], makes it evident that beaconing is expected to be present to support a wide variety of safety-related applications. By means of simulations, we showed that DOT is scalable to increasing network densities, achieves near optimum delay results, and is robust to errors caused by possible inaccurate transmission range estimations. Furthermore, DOT outperformed other delay-based schemes in diverse network densities.

Finally, Section 3.3 presents the AMD protocol. AMD inherits concepts from both SRD and DOT protocols to combine a generalized time slot scheme based on directional sectors and a store-carry-forward algorithm to support multi-directional data dissemination, thereby working seamlessly in both highway and urban scenarios. By means of simulation, we showed that AMD scales properly in various network densities in both highway and urban scenarios. Compared with protocols especially designed for either highway or urban scenarios, AMD obtained higher delivery ratio, lower end-to-end delay, and lower number of transmissions.

Based on the results presented in this chapter, we can conclude that AMD can serve as an efficient multi-purpose protocol for safety-related data in both highway and urban scenarios. AMD incorporates concepts and evolves from SRD and DOT and, therefore, represents the most complete solution of this chapter. Nevertheless, if the application only requires the dissemination of messages to a specific direction, DOT may be a better candidate. This comes from the fact that DOT works in a distributed manner, thereby inserting no overhead to data messages for the sorting of vehicles' priority in the neighborhood.

CHAPTER 4

Data dissemination for non-safety applications

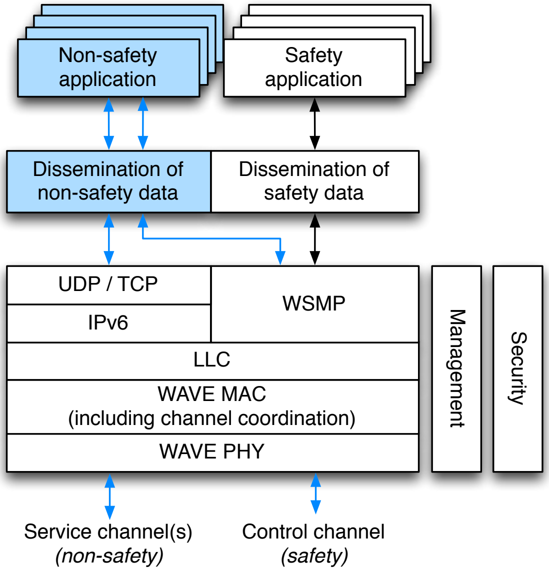


Figure 4.1: The WAVE stack with highlighted module for non-safety applications

In this chapter[1], we present solutions for the dissemination of data related to non-safety applications. Applications in this category are expected to gen-

---

[1] This chapter is based on the following publications: (i) *Analysis of Utility-Based Data Dissemination Approaches in VANETs*, 4th International Symposium on Wireless Vehicular Communications (WIVEC) - VTC Fall 2011 [24]; (ii) *Achieving Data Utility Fairness in Periodic Dissemination for VANETs*, IEEE 75th Vehicular Technology Conference (VTC Spring) 2012 [25]; (iii) *Fair and adaptive data dissemination for traffic information systems*, 4th IEEE Vehicular Networking Conference (VNC) 2012 [26]; and (iv) *On the applicability of fair and adaptive data dissemination in traffic information systems*, Elsevier Ad Hoc Networks (accepted for publication).

erate large amounts of data that, due to a limited channel capacity and high speed of vehicles, cannot be fully shared in the neighborhood. Thus, we concentrate our efforts into the process of selecting and disseminating the most relevant data to interested vehicles while at the same time controlling the network load.

Figure 4.1 highlights the components used for non-safety data dissemination as defined by our approach described in Chapter 1. A module placed between non-safety applications and the WAVE protocol stack is used to coordinate messages in the neighborhood. In the first two sections of this chapter, we assume a radio set-up with one transceiver where messages are sent to a single channel. Later on, we explore a more realistic setting where both control and service channels are used. In addition, we assume that every vehicle is able to determine its current geographical position on the road using, for example, the Global Positioning System (GPS). Finally, evaluation parameters such as transmission range and scenario size are adjusted throughout the sections according to their suitability to meet scalability requirements in terms of simulation execution time.

The remainder of the chapter is organized as follows. In Section 4.1, we present a study of the trade-offs between fairness and efficiency when used as goals for selecting data. After verifying the advantages of relying on fairness for selecting data, Section 4.2 presents a data dissemination protocol that distributes data utility fairly among vehicles in the neighborhood. In Section 4.3, we describe a data dissemination protocol that combines the fair approach presented in the previous section with a method for controlling the transmission rate of messages, thereby distributing data utility fairly over vehicles while adaptively controlling the network load. Finally, Section 4.4 finalizes this chapter with concluding remarks.

4.1 Exploring fairness vs. efficiency as goals for data selection

4.1.1 Introduction

Due to the continuous collection, processing, and dissemination of data, one crucial requirement non-safety applications is the efficient use of the available bandwidth. The amount of data collected can increase quickly even with aggregation algorithms. In addition, the time window for data exchange can be very limited due to the rapidly changing road environment. In such a scenario, data dissemination protocols must incorporate efficient mechanisms to select the most relevant data to maximize the utility (benefit) gain of vehicles, where data utility is measured on the basis of available contextual information. In this line, two fundamentally distinct approaches can be identified in the literature. The first aims to maximize the system *efficiency*. Vehicles select data with the goal of maximizing the total utility gained by all vehicles in the vicinity [60]. The second focuses on a *fair* distribution of utility among vehicles [69]. As previously shown in Figure 2.4, a fair approach is particularly suitable for situations where vehicles have conflicting data interests. For example, two vehicles moving in opposite directions may be potentially interested in each other's data, since one holds data related to the destination of the other.

In this section, we explore the trade-off between efficiency and fairness as goals for data selection. In particular, we analyze this trade-off in terms of various performance metrics, paying special attention to Jain's fairness index and the sum of utility gains.

4.1.2 Utility function

Throughout this chapter, we refer to the *utility* of a data message as the benefit that a vehicle can have by receiving that message. A message's utility is calculated by each corresponding application based on the current level of "interest" that a vehicle has in the message's content depending on the vehicle's current context. For instance, if a message contains information about the vehicle's final destination, the application may consider giving a high utility to this message. However, from the perspective of another vehicle moving towards a different destination, the same information might be considered almost irrelevant. We classify this contextual knowledge into the following categories:

- ***Mobility context***: ranges from the complete route of a vehicle to the vehicle direction, speed, mobility history, etc.

- ***Data context***: includes the priority of the data message, age, geographical region, etc.

This contextual information can be weighted in a function which attributes a value u_{ij} to each data message m_j in view of vehicle v_i. The normalized utility value is given by:

$$u_{ij}(\alpha_1 z_1^i(m_j), \alpha_2 z_2^i(m_j), ..., \alpha_l z_l^i(m_j)), \tag{4.1}$$

where z_k^i with $k = 1, 2, ..., l$ are the functions of each type of contextual information k for vehicle v_i weighted by parameters α_k. These functions are normalized with values falling in a predetermined interval, e.g., $[0, 1]$. The application is responsible for defining how these functions are combined in u_{ij}.

4.1.3 Data selection models

We evaluate two fundamentally distinct optimization models for data selection which we refer to as *Total Sum Optimization (TSO)* and *Fair Sum Optimization (FSO)*. TSO aims to maximize the system *efficiency*. In [60] this is referred to as being an *altruistic* approach, since vehicles select data with only the goal of maximizing the total sum of utilities gained by all vehicles regardless of how much they profit individually. In contrast, FSO focuses on a *fair* distribution of utility among all vehicles.

To achieve fairness, we rely on concepts of bargaining from game theory. The goal is to reach an agreement on the utility outcome when vehicles are willing to cooperate. The key foundation of bargaining models is the the Nash Bargaining solution [96], which has been widely used in fields such as network bandwidth allocation and has been recently proposed for use in vehicular networks for data exchange in [69]. In [96] it is proved that in a convex, closed and bounded set the solution is unique for the following axioms:

i) *Pareto optimality*: at the bargaining outcome, no player can improve without decreasing the other player's utility.

ii) *Symmetry*: if the utility region is symmetric around a line with slope 45 degrees then the outcome will lie on the line of symmetry.

iii) *Invariance to linear transformation*: the bargaining outcome varies linearly if the utilities are scaled using an affine transformation.

iv) *Independence of irrelevant alternatives*: if the solution of the bargaining problem lies in a subset U' of S, then the outcome does not vary if the bargaining is performed on U' instead of S.

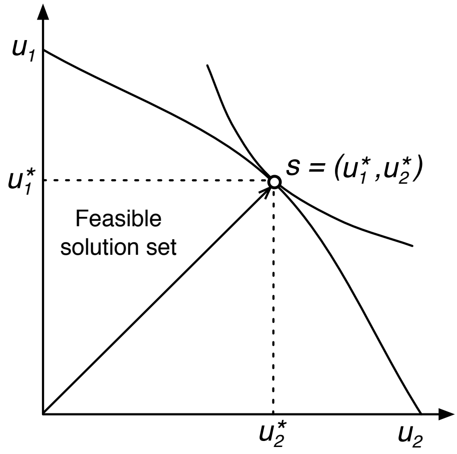


Figure 4.2: Overview of the Nash Bargaining solution

The Nash Bargaining solution is exemplified in Figure 4.2. The solution $s = (u_1^*, u_2^*)$ is the point in the boundary of the feasible set that maximizes the product of the vehicles' utility functions, thereby satisfying all four axioms.

Among alternatives to the Nash Bargaining solution are the Kalai-Smorodinsky [97] and the Egalitarian [98] bargaining solutions. Kalai-Smorodinsky preserves all axioms from the Nash Bargaining solution except for the independence of irrelevant alternatives that is replaced by the axiom of individual monotonicity. The reason for this modification comes from the criticism related to certain problem formulations, especially in economics, where this axiom would not hold. However, in this thesis we choose the Nash Bargaining solution as means to achieve fairness, since we assume that each application defines the utility of a generated message solely based on the current contextual information. In other words, each message is given a utility value that is independent from changes in the set of messages available in the neighborhood. Later in this chapter, we explore the Max-min [99] solution that seeks to achieve an Egalitarian distribution of data utility.

The proposed optimization models are defined as follows. A vehicle v_i independently stores its *local* knowledge of the neighborhood into the utility matrix U. Let U be utility matrix for h vehicles and n data messages,

$$U = \begin{array}{c c} & \begin{array}{cccc} m_1 & m_2 & \dots & m_n \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ \\ v_h \end{array} & \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ u_{21} & u_{22} & \dots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{h1} & u_{h2} & \dots & u_{hn} \end{pmatrix} \end{array}, \tag{4.2}$$

where u_{ij} is given by (4.1). In matrix U, the utility value for each pair (v_i, m_j) is given. There are n potential distinct data messages to be sent in the neighborhood. For a message to appear in U, there is at least one vehicle that has not received it yet. If vehicle v_i already has message j, then $u_{ij} = 0$.

TSO and FSO are then defined respectively by (4.3) and (4.4). The binary vector $x = x_1, ..., x_n$ selects the messages m_j which will be transmitted. Given the total data exchange duration time, the maximum number of messages which can be transmitted is k_{max}.

$$\max \sum_{i=1}^{m} \sum_{j=1}^{n} [u_{ij} x_j] ; \tag{4.3}$$

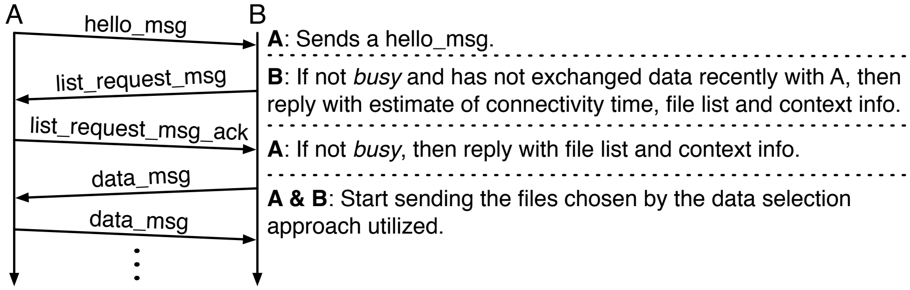
$$\max \prod_{i=1}^{m} \sum_{j=1}^{n} [u_{ij} x_j] ; \tag{4.4}$$

$$\text{with} \quad x_j \in \{0, 1\} \quad \text{s.t.} \quad \sum_{j=1}^{n} x_j \leq k_{max}.$$

With this formulation, TSO and FSO select the message that yields the maximum *sum* and *product* of utility gains in the neighborhood, respectively.

4.1.4 Basic protocol

To compare the data selection models, we define a basic protocol for the data exchange of every pair of vehicle. The decision to restrict the protocol to only two vehicles is because of the extra robustness achieved when using RTS/CTS of the 802.11p MAC layer. The basic protocol works as shown in Figure 4.3. Vehicles which are not currently exchanging data (because they are busy) and which have not recently exchanged data with each other start a simple handshaking process after a periodic *hello* message is received. This involves the exchange of the list of messages that they possess and the contextual information used by utility function u_{ij}. This guarantees that both vehicles generate the same utility matrix U. The connectivity time is estimated by vehicle B and included in *list_request_msg*. This estimation is based on the current route information available about both vehicles. Finally, the data selection model is run by both vehicles individually and messages are exchanged by means of data messages.



A: Sends a hello_msg.

B: If not *busy* and has not exchanged data recently with A, then reply with estimate of connectivity time, file list and context info.

A: If not *busy*, then reply with file list and context info.

A & B: Start sending the files chosen by the data selection approach utilized.

Figure 4.3: Basic protocol for data exchange between every pair of vehicle

4.1.5 Performance evaluation

The performance evaluation of both TSO and FSO is carried out by means of simulations. Our goal is to understand the fundamental differences between the two approaches in various performance metrics and scenarios.

We utilize OMNeT++ 4.1 [80] with MiXiM v2.0.1. We adjust the available implementation of IEEE 802.11b to comply with basic specifications of the 802.11p version. In the MAC layer, we set the bit rate to 6 Mbit/s, the Contention Window (CW) to values between 15 and 1023, the slot time to 13 μs, the SIFS to 32 μs, and the DIFS to 58 μs. In the physical layer, we operate in the 5.88 GHz frequency band, using 10 MHz of bandwidth.

With regard to the transmission power employed, different values may be used according to the application's priority. Efforts put on selecting a proper transmission power value include the Decentralized Congestion Control (DCC) mechanism as defined by the ETSI European standardization [12] that controls the network load by adjusting the transmission power level and transmission rate. However, our goal here is limited to achieving a proper balance between choosing realistic values (i.e., up to 500 meters of range) and achieving scalability in the simulations in terms of the overall processing time. Above all, we are particularly interested in data exchanges with limited connectivity duration time in order to properly understand the contrasts in terms of performance between the data selection mechanisms. Despite leading to higher delay and lower number of messages received per vehicle, lower transmission ranges are clearly more suitable to meet this goal. They are also preferred in terms of scalability, since fewer vehicles would be sharing the medium in the simulation simultaneously. This explains the different power values used throughout this chapter.

In this section, we set the transmission power to 168.98 mW to achieve approximately 500 meters of interference range and 250 m of transmission range, assuming the Friis Free Space Path Loss (FSPL) propagation model with an exponent α equal to 3.5. The signal-to-noise threshold is set to 0.1259 mW, receiver sensitivity to -119.5dBm, and thermal noise to -110dBm. The simulation parameters are summarized in Table 4.1.

Regarding the utility function, different results can be expected when different contextual information and parameters are considered by an application. Our goal is to define basic functions and parameters that can be common to various applications. Thus, the utility function u_{ij} is defined as:

$$u_{ij} = \alpha_1 z_1^i(m_j) + \alpha_2 z_2^i(m_j) + \alpha_3 z_3^i(m_j), \tag{4.5}$$

which is composed by the contextual knowledge functions:

***Data priority** (*α_1*,* $z_1^i(m_j)$*):* we define three levels of priority for m_j related to its data, namely, traffic information, parking information, and general infotainment, returning respective z_1^i values of 1.0, 0.8, and 0.6. α_1 is set to 0.8.

***Closest distance to a message's region** (*α_2*,* $z_2^i(m_j)$*):*

Table 4.1: Simulation parameters

Physical layer	Frequency band	5.88 GHz
	Bandwidth	10 MHz
	Transmission range	~250 m
	FSPL exponent α	3.5
	Receiver sensitivity	-119.5 dBm
	Thermal noise	-110 dBm
Link layer	Bit rate	6 Mbit/s
	CW	[15,1023]
	Slot time	13 μs
	SIFS	32 μs
	DIFS	58 μs
Basic protocol	*Hello* message size	24 bytes
	Hello message freq.	1 Hz
	Max. msg list size in *hello*	100
Scenarios	Data message size	2312 bytes
	Initial # messages	10
	# Runs	50

$$z_2^i(m_j) = 1 - \frac{d^i(c_{m_j})}{5000}, \tag{4.6}$$

where $d^i(c_{m_j})$ is a function which calculates the shortest distance in meters to which the vehicle i approaches the message's geographical coordinates c_{mj}. The accuracy of this measure will depend on the current information the vehicle has about its route, e.g., complete route if it is set in a navigation system. Only distances within an area of 5 km^2 ($d(c_{m_j}) \leq 5000$) are considered, and α_2 is set to 0.15.

***Data age** (**α_3, $z_3^i(m_j)$**)*:

$$z_3^i(m_j) = 0.99^{t_{mj}}, \tag{4.7}$$

where t_{mj} is the time elapsed since the message's generation time and α_3 is set to 0.05.

With the weights applied, most importance is given to the priority of a message. The reasoning behind this choice lies in defining parameters that enable the assessment of the data selection approaches when a disparity of utility values exists between vehicles.

For the data dissemination approaches, *hello* messages are of 24 bytes and sent at 1 Hz and data messages are 2312 bytes (the maximum allowed by 802.11p). Every vehicle begins the simulation with 10 messages. Each message's geographical coordinates are set to one of the two diagonal extremes of the simulation map: either (0,0) or (x_{max},y_{max}). The decision of choosing either coordinates is made at the beginning of the simulation based on the vehicles's direction. More specifically, if a vehicle is traveling towards a region near (0,0), the message's coordinate will be assigned to (x_{max},y_{max}), or (0,0) otherwise. In this manner, we simulate vehicles which have already passed by the message's geographical region before and now carry the message to other regions, for example, when they acquire a message in one city and then travel to another. In addition, messages' generation times are simply defined as the beginning of the simulation, i.e., time zero. To create disparity between the utility that each vehicle may gain in a data exchange, we assign different priorities following a similar condition to the one applied for the message's geographical coordinate: 1.0 if it is traveling towards (0,0); and 0.8 or 0.6 (with 50% each) otherwise. Finally, whenever two vehicles exchange data the optimization problems defined for each data selection approach (Section 4.1.3) are solved externally by the General Algebraic Modeling System (GAMS) [100].

Since the approaches being compared depend on an accurate estimation of the connectivity duration time, we take the following two measures to improve this estimation. First, instead of sending multiple data messages that could form a file, we simulate large files by decreasing the rate at which data messages are sent to the MAC layer. Each data message is sent every 1.5 second, which simulates files of 1.125 Mbyte. Second, we assume that every vehicle is aware of its complete route followed in the simulation which is shared with other vehicles via *hello* messages. This could be thought of as if every driver had set the route in his navigation system. Although these measures are certainly not realistic, we are able to focus more closely on the data selection approaches and more accurate conclusions can be drawn about which approach an actual protocol should be based on.

Our evaluation considers the following metrics:

- ***Jain's fairness index***: calculated for each interaction between pairs of vehicles. It is defined as $(\sum_i^h x_i)^2/(h\sum_i^h x_i^2)$ (see [101]), where h is the total number of vehicles and x_i is the utility sum gained by vehicle i. It indicates how well utility gains are distributed among vehicles on average. $1/h$ and 1 are the worst and best cases, respectively.

- ***Sum of utility gains***: the average total utility sum gained by an arbitrary vehicle. It measures the overall performance of the dissemination approach.
- ***Percentage of nodes which gained utility***: also indicates how well utility gains are distributed but specifically in terms of which percentage of vehicles finish the simulation with no utility gain.
- ***Number of messages received***: average number of messages received by an arbitrary vehicle.
- ***Number of hops***: average number of hops that a message travels.
- ***Delivery ratio***: the percentage of messages received by vehicles which eventually approach a 1 km radius surrounding the message's geographical coordinates. It measures the approach's efficiency in terms of how often messages are distributed to vehicles actually interested in the messages.
- ***Delay***: the average amount of time taken from the message's generation until it is received by vehicles that will be traveling to the area to which the message relates.

4.1.5.1 Two-vehicle scenario

We first consider a simple scenario in which two vehicles drive in opposite directions at an average speed of 120 km/h. A space of 10 meters separates each road direction. This simulation consists of 50 runs. Our goal is to assess the performance of each approach in a basic scenario with increasing available connectivity time (k_{max}).

Figure 4.4(a) presents the results of applying the Jain's fairness index. FSO seeks symmetry in the distribution. Therefore, the index is above 0.9 for even numbers of k_{max} and above 0.75 for odd numbers, since with even numbers the probability of achieving symmetry is higher. On the other hand, when k_{max} is low TSO tends to increase the utility of only one vehicle (index of 0.5). It is worth noticing that with the defined start-up configuration for each message's priority, generation time, and coordinate, we provoke a utility disparity between vehicles. Therefore, with TSO one vehicle will always be able to obtain a higher utility gain when compared to the other in this scenario. However, the

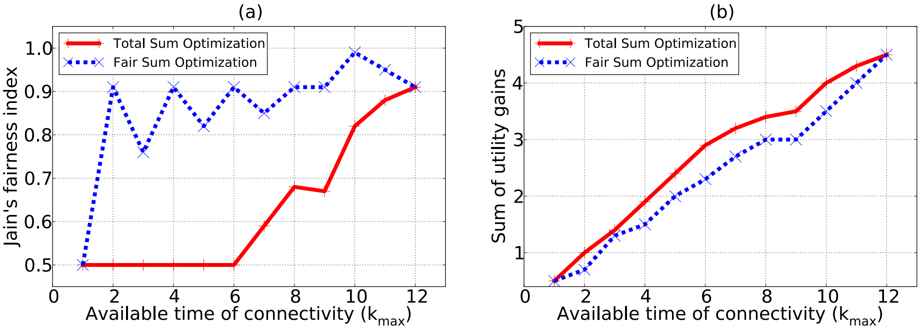


Figure 4.4: Results with 95% confidence intervals for the two-vehicle scenario

two approaches present similar values when k_{max} is higher. This is expected since with increasing time, more messages will be sent. Thus, fewer differences in the results are present.

The high level of fairness achieved with FSO comes at the price of a lower performance in terms of the total sum of utility gains, as shown in Figure 4.4(b). However, this loss in terms of the maximum difference is just 20% compared to the gain in Jain's index at 40%.

Overall, FSO presents more advantageous results when compared with TSO, since it provides a fairer distribution of utility among the vehicles while not compromising the overall performance achieved by all vehicles taken together.

4.1.5.2 Traffic simulator scenarios

In the following, we consider more realistic scenarios created with SUMO [83]. These scenarios include vehicle overtaking, lane changing, and rely on well-known car-following mobility models such as Krauß and Intelligent Driver Model (IDM). Three scenarios with an area of 6 km^2 are considered: an OpenStreetMaps [94] map fragment (2 x 3 km^2) from the urban TAPAS Cologne traffic model [102] combining both sparse and dense networks (scenario 1), a well-connected network (scenario 2), and a sparse network (scenario 3). Scenario 1 describes the traffic within the city of Cologne, Germany (Figure 4.5), with traffic demands generated by TAPAS – a system that computes mobility demands based on the population's traveling habits and the city's infrastruc-

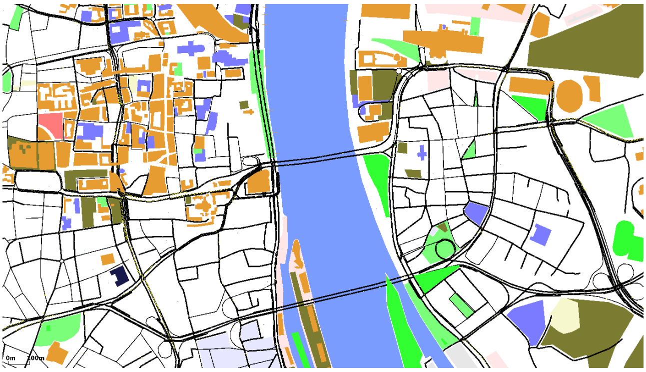


Figure 4.5: Urban scenario: map fragment of the city of Cologne, Germany

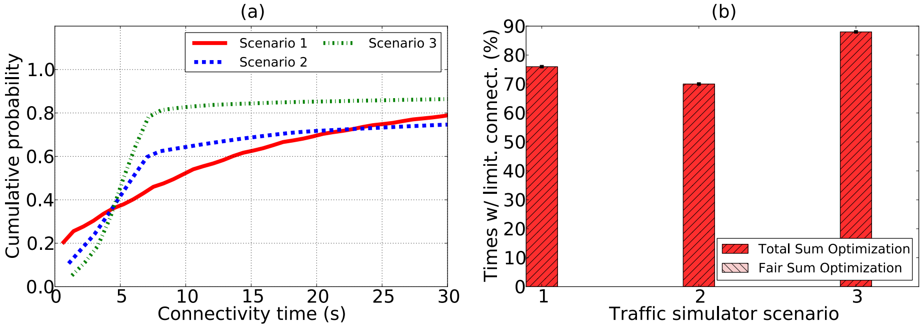


Figure 4.6: Study of the connectivity time in the scenarios used in the evaluation

ture. In this scenario, the number of vehicles simultaneously moving increases linearly with time from 10 to 470, with a total of 709 generated. Vehicles' speeds vary from 0 to 100 km/h. Scenarios 2 and 3 contain a 10-kilometer straight highway with two lanes in each road direction. Lanes are 4 meters wide with a 10-meter space between each direction. A moderate traffic flow generated for scenario 2 which leads to a density of 7.5 vehicles/km/lane and for scenario 3 a low traffic flow which leads to a density of 2.5 vehicles/km/lane. In both scenarios, vehicles' speeds vary from 80 km/h to 120 km/h. Simulations consist of 25 runs of 300 seconds.

We first study the connectivity profile of each scenario in Figures 4.6(a) and 4.6(b). Figure 4.6(a) shows the cumulative distribution function (CDF) for the connectivity time of any interaction between a pair of vehicles with both approaches. We consider a restricted version of the figure limiting the x-axis to just 30 seconds to focus on a limited period, since changes are nearly unnoticeable after this point. In the sparse topology present in scenario 3, 80% of interactions are performed under 10 seconds. In the remaining scenarios, 60% of interactions are within 10 seconds in scenario 2 while 50% are within 10 seconds in scenario 1. Figure 4.6(b) shows the percentage of times a data exchange is performed with limited connectivity time, i.e., not all files that are available can be exchanged. While these percentages depend significantly on the amount of data initially carried by each vehicle, in our configuration with only 10 messages, each simulation 1.125 Mbyte, values are equal or greater than 70% for all scenarios. This verifies that vehicles often have to deal with limited connectivity time.

Figure 4.7 shows the results of applying the Jain's fairness index and the sum of utility gains. We first narrow down the results to only cases with time-limited data exchanges in Figures 4.7(a) and 4.7(b). As with the results in the two-vehicle scenario, FSO achieves a Jain's index which is 20% to 65% higher than TSO. The difference is most noticeable in scenario 3 because of the presence of a sparse network in which vehicles can mostly communicate with other vehicles traveling in the opposite direction. This reflects the results shown in Figure 4.4(a) where with a short connectivity time results between the two approaches diverge in a great extent. In addition, despite the greater sum of utility gains achieved by TSO, the difference compared with FSO is less than 7%. When considering all data exchanges in each scenario (Figures 4.7(c) and 4.7(d)), the Jain's index is still higher with FSO; however, it has a lower difference compared to TSO. Especially in data exchanges with abundant time, vehicles are likely to gain more in terms of utility, which directly influences the average value of the Jain's index.

Interestingly, with all data exchanges considered, FSO outperforms TSO in terms of the sum of utility gains. To understand the reasons for this result, we evaluate four other metrics, as shown in Figure 4.8. The first metric is the percentage of vehicles which gained utility during the simulation. As shown in Figure 4.8(a), while FSO seeks symmetry in the utility distribution and almost guarantees that every vehicle receives some utility, with TSO more than 15% of the vehicles finish the simulation without any message having been received. This is also reflected in the total number of messages received (Figure 4.8(b)) with more messages being received when FSO is utilized. FSO also has advan-

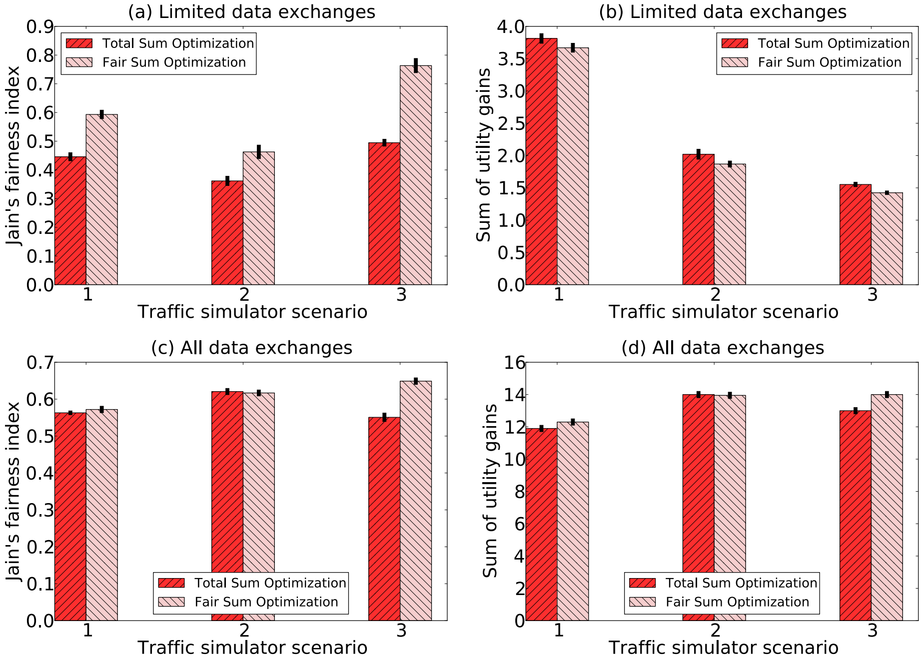


Figure 4.7: Results with 95% confidence intervals considering data exchanges with limited connectivity time in (a) and (b); all data exchanges in (c) and (d)

tages in terms of the average number of hops a message travels and especially in terms of the delivery ratio (Figures 4.8(c) and 4.8(d)). This is a direct consequence of the previous results: with more vehicles receiving messages, messages are more likely to spread throughout the network and reach interested vehicles. Finally, given that messages are better distributed among vehicles a shorter delays is expected with FSO. However, due to the much lower delivery ratio achieved by TSO, the results are not comparable. Therefore, the results for both approaches are almost identical with small variations. For scenarios 2 and 3, in which vehicles move at high speeds the average delay is within the range of 110 to 120 seconds. For scenario 1, the delay is higher (170 to 180 seconds) since vehicles are moving through an urban area and thus at lower speeds.

Overall, improving fairness while disseminating data can lead to a better overall Jain's fairness index, comparable or increased sum of utility gains,

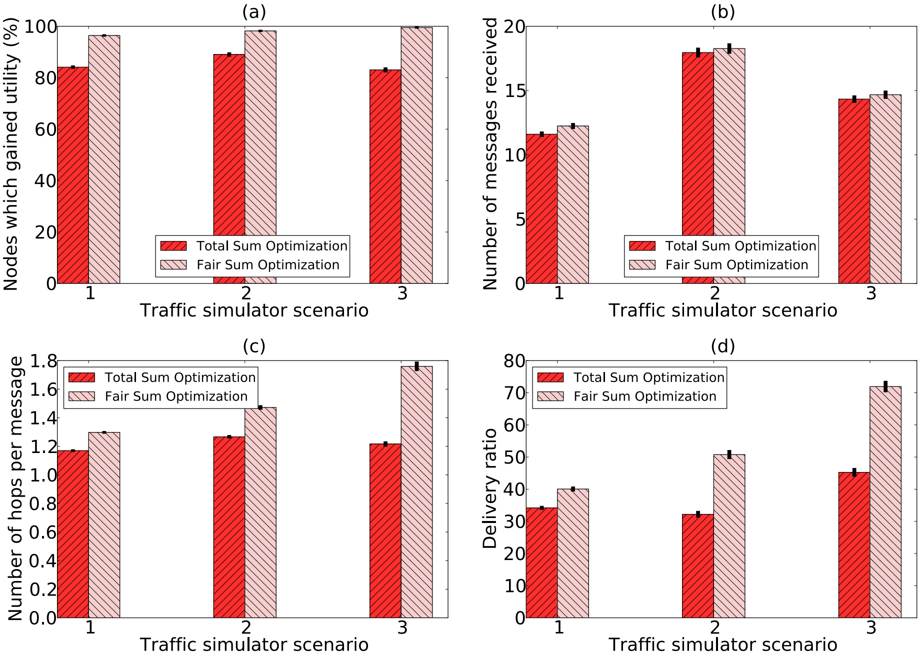


Figure 4.8: Results with 95% confidence intervals for various metrics for the traffic simulator scenarios

number of hops and number of messages received. Most importantly, as a direct consequence of these results, a fair approach has the potential to significantly reach more interested vehicles.

4.1.6 Conclusion

This section has presented a study of the trade-off between efficiency and fairness when disseminating data in vehicular environments. By means of simulation, we have evaluated both approaches using a variety of performance metrics and scenarios. Overall, the use of a data selection strategy which seeks fairness has shown to reach more interested vehicles, to present higher Jain's fairness index, and comparable or higher sum of utility gains.

We concentrated on comparing *optimal* results of both efficient and fair strategies. Therefore, a few simplifications were proposed which deserve further attention. First, the basic protocol used in our comparison prioritizes communication robustness with the use of RTS/CTS of 802.11p. Hence, it allows only the communication between every pair of vehicles. However, in various situations multiple vehicles will be simultaneously available for exchanging data. Although more unreliable, broadcast communication may effectively improve the protocol's performance by making better use of the available bandwidth, and consequently of time. Second, the data selection model defined to achieve fairness relies on the Nash Bargaining solution for binary variables. This solution falls in the class of Mixed-Integer Non-Linear Programming (MINLP) which are NP-hard problems. Finally, we considered that the connectivity time is calculated before the actual data exchange. Such calculation can be difficult due to unpredictable variations of busy radio medium and incomplete information about vehicle routes. To overcome these limitations, in the following sections we present protocols that rely on broadcast communication to distribute data utility fairly in the neighborhood while at the same time being able to adapt data selection decisions during the data exchange as new context knowledge about the vicinity is acquired.

4.2 Achieving fairness via synchronous periodic dissemination

4.2.1 Introduction

For many non-safety applications such as traffic monitoring, the dissemination of data can be accomplished by means of periodic broadcast with longer intervals compared to safety beaconing [10]. Periodic dissemination is especially suitable to dynamic scenarios, since there is no need for changing the protocol's operation mode to suit the current environment [17].

In view of the advantages of relying on fairness for selecting and sharing data in the neighborhood, we present in this section the FairDD: *Fair D*ata *D*issemination protocol. FairDD focuses on periodically disseminating data utility fairly in the neighborhood according to a defined application cycle and provide means for keeping the network load under a defined value by suppressing only the least relevant data messages.

4.2.2 Fair data dissemination

FairDD aims to achieve a *fair* distribution of data utility throughout the network while keeping the network load under a defined level. FairDD consists of two main components: (i) a distributed fair data selection mechanism and (ii) a synchronized suppression mechanism to cancel only the least relevant data messages.

4.2.2.1 Data selection

To achieve utility fairness in the neighborhood, we propose a distributed data selection mechanism that considers the individual interests of vehicles. FairDD relies on the Nash Bargaining [96] solution from game theory. This solution achieves a compromise between fairness and efficiency. Fairness refers to the symmetry of utility distribution among vehicles and efficiency refers to the total utility distributed.

A vehicle v_i employing FairDD independently stores its *local* knowledge of the neighborhood into two variables: utility matrix U and vector of accumulated utility c_i. Let U be utility matrix for h vehicles and n data messages,

$$U = \begin{array}{c c} & \begin{array}{cccc} m_1 & m_2 & \dots & m_n \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ \\ v_h \end{array} & \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ u_{21} & u_{22} & \dots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{h1} & u_{h2} & \dots & u_{hn} \end{pmatrix} \end{array}, \tag{4.8}$$

where u_{ij} is given by (4.1). In matrix U, the utility value for each pair (v_i, m_j) is given. There are n distinct data messages to be sent in the neighborhood. For a message to appear in U, there is at least one vehicle that has not received it yet. If vehicle i already has message j, then $u_{ij} = 0$.

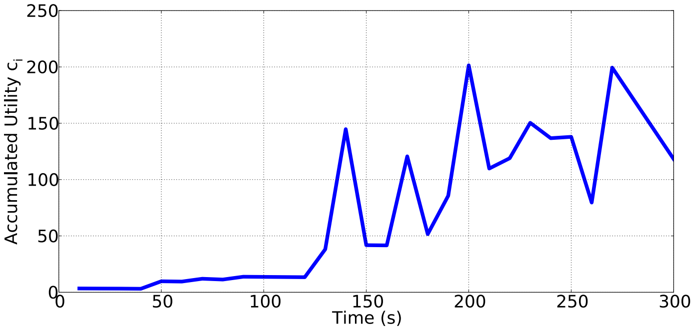


Figure 4.9: Example of the accumulated utility (c_i) concept for a random vehicle moving in the city of Enschede, The Netherlands

One main feature of FairDD is that we take into account the accumulated utility c_i of each vehicle v_i. In this way, a vehicle that gained more in previous opportunities will have a lower priority to increase its c_i in the next data exchange. Nevertheless, since the communication is broadcast-based, such a vehicle might still gain non-zero utility from overhearing. Another property of c_i is that it continually changes depending on the current context of v_i. A change of context might lead to a change of the message's utility (see Equation (4.1)), thereby affecting c_i. For example, when a vehicle moves from one geographical region to another or when a message becomes old. Figure 4.9 shows the evolution of c_i when a random vehicle i moves in one of our simulation scenarios. The utility function considered takes into account the vehicle direction, closest distance to message's region, message age, and data priority (detailed in Section 4.2.3). A vehicles starts receiving utility but as time goes by or as the vehicles changes its direction, its accumulated utility c_i begins to fluctuate.

Algorithm 4.1 FairDD_DataSelection

```
Input: $U$, $\vec{c}$ // matrix and vector of accumulated utility
Output: $\vec{q}$ // vehicle's queue of selected messages
 1: $q \leftarrow \oslash$
 2: $r \leftarrow 0$ // message's order to be sent in neighborhood
 3: $J \leftarrow \{0, 1, ..., n\}$
 4: while $U \neq \oslash$ do
 5:     $t \leftarrow \arg\max_{j \in J} \prod_{i=1}^{h} [u_{ij} + c_i]$
 6:     if vehicle has $m_t$ and is farthest from last sender then
 7:         $q.add(m_t, r)$ // add $m_t$ in $\vec{q}$, store its order $r$
 8:     end if
 9:     for each neighbor $i$ do
10:         $c_i \leftarrow c_i + u_{it}$
11:     end for
12:     remove $m_t$ from $U$
13:     remove $t$ from $J$
14:     $r \leftarrow r + 1$
15: end while
```

The data selection process defines in a distributed manner the order in which messages are sent in the neighborhood and the vehicles to send these messages. Each vehicle calculates its local optimum solution based on the information received from one-hop neighbors, since acquiring global information is infeasible. This process is defined by Algorithm 4.1. U and $\vec{c}$ are the utility matrix and the vector of accumulated utility values for each vehicle, respectively. The core function is described in line 5. The Nash Bargaining solution maximizes the product of the sum of the utility gain u_{ij} and accumulated utility c_i of each vehicle. Therefore, in matrix U, message m_t maximizing $\prod_{i=1}^{h} [u_{ij} + c_i]$ will be selected. To guarantee that this product is higher when more neighbors are profiting, we set a lower bound $\varepsilon = 1$ for c_i. If a vehicle has m_t, then m_t is added to its queue of selected messages $\vec{q}$. However, to prevent transmission redundancies, each message selected should be sent by only one vehicle. Thus, in case there are multiple vehicles carrying m_t, the one farthest away from the previous sender of m_t will be selected, thereby allowing for a quick message dissemination. In each iteration, U is updated (lines 12 and 13) and the next optimum result is calculated. In the end, queue $\vec{q}$ defines which messages carried by each vehicle will be broadcast and at which order in the neighborhood.

The complexity of Algorithm 4.1 is upper-bounded by the search of the maximum product in line 5. In total, $h\sum_{a=0}^{n}[n-a]$ operations are performed, where h and n are the number of vehicles and messages in the neighborhood, respectively. Since h is always limited by the transmission range, the complexity comes down to $O(n^2)$.

4.2.2.2 Protocol

We propose a network protocol that translates the message ordering defined by the data selection to the network. We consider periodic applications using broadcast communication, where the periodicity of an application is referred to as the application cycle. We define that vehicles are capable to synchronize their cycles to the Coordinated Universal Time (UTC), such as with a GPS device. This is in accordance to the IEEE WAVE standard [103] that defines that devices not capable of operating on multi-channels simultaneously must rely on such synchronization for vehicles to access the control channel (CCH) at specified time intervals.

The protocol is shown in Figure 4.10. In the beginning of each cycle, each vehicle calculates its queue of selected messages $\vec{q}$ with Algorithm 4.1. The application cycle is divided in equal time slots of size defined as the maximum time taken for each transmission. Each message is transmitted in the time slot number corresponding to the message's order r. If r exceeds the total number of time slots $n = (cycle/slottime)$, the message is scheduled to $[n-1]$. With this scheme, messages with higher priority are sent in earlier time slots compared with low priority messages. Since this order reflects the local optimum calculated by each vehicle individually, there is a chance that more than one message is sent in a single time slot. In this case, we introduce a small random delay before each transmission to prevent collisions. To keep the network load under a specified level, we define a suppression line that represents the total number of messages allowed to be sent in the neighborhood. Therefore, only messages scheduled after the suppression line are canceled.

As explained previously, the data selection mechanism depends on the current contextual knowledge acquired by each vehicle in order to build matrix U. For such purpose, we define periodic *hello* messages that are sent asynchronously in parallel with data messages. Each *hello* message sent by vehicle

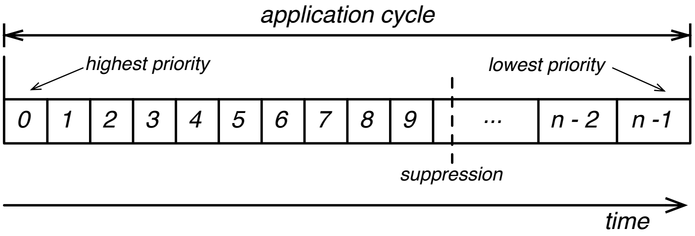


Figure 4.10: FairDD – synchronized suppression mechanism

i contains a summarized list of messages carried by i with information such as message age and geographical region where it was generated. In addition, the following information is included: vehicle's ID, direction, final destination and accumulated utility c_i.

4.2.3 Performance evaluation

The performance evaluation of FairDD is carried out by means of simulations. Our goal is two-fold: (i) evaluate the advantages of employing data selection mechanisms to use the bandwidth more efficiently and (ii) compare FairDD's data selection with other approaches, namely:

- ***Altruistic***: based on the work presented in [60], this approach maximizes the total utility gain for all neighbors as a whole. Thus, it does not consider individual interest. It gives an upper-bound in terms of efficiency for individual message selections.
- ***No selection***: no utility is considered when selecting a data message. We simply define that messages with lower ID are sent first. Thus, messages with higher ID numbers are more likely to be suppressed.
- ***No suppression***: just as with *No selection*, no utility is used in the data selection. However, the maximum number of messages allowed by the application is sent. We define this maximum to be equal to the total number of neighbors.

We use OMNeT++ 4.1 simulator [80] with MiXiM v2.0.1. We adjust the implementation of IEEE 802.11b to comply with basic specifications of 802.11p.

Table 4.2: Simulation parameters

Physical layer	Frequency band	5.88 GHz
	Bandwidth	10 MHz
	Transmission range	∼100 m
	FSPL exponent α	3.5
	Receiver sensitivity	-119.5 dBm
	Thermal noise	-110 dBm
Link layer	Bit rate	6 Mbit/s
	CW	[15,1023]
	Slot time	13 μs
	SIFS	32 μs
	DIFS	58 μs
FairDD	*Hello* message size	2312 bytes
	Hello message freq.	1 Hz
	Data message freq.	1 Hz
	Max. msg list size in *hello*	100
	Slot time for data messages	10 ms
Scenarios	Data message size	2312 bytes
	Initial # messages	10
	# Runs	50

In the MAC layer, we set the bit rate to 6 Mbit/s, the Contention Window (CW) to values between 15 and 1023, the slot time to 13 μs, the SIFS to 32 μs, and the DIFS to 58 μs. In the physical layer, we operate in the 5.88 GHz frequency band, using 10 MHz of bandwidth.

Due to the use of broadcast communication, more messages are received by each vehicle compared to using the basic protocol defined in Section 4.1.4. Following the reasoning explained in Section 4.1.5, a lower transmission power is used in the remaining sections in order to compensate for the higher simulation processing time. More specifically, we set the transmission power to 10 mW to achieve approximately 200 meters of interference range and 100 m of transmission range, assuming the Friis Free Space Path Loss (FSPL) propagation model with path loss coefficient equal to 3.5. The signal-to-noise threshold is set to 0.1259 mW, receiver sensitivity to -119.5dBm, and thermal noise to -110dBm. The simulation parameters are summarized in Table 4.2.

The utility function u_{ij} is defined as:

$$u_{ij} = p(\alpha_1 z_1^i(m_j) + \alpha_2 z_2^i(m_j) + \alpha_3 z_3^i(m_j)), \tag{4.9}$$

which is composed by the contextual knowledge functions:

***Vehicle direction** (α_1, $z_1^i(m_j)$)*: if the vehicle i is going towards the data message's geographical region, z_1^i returns 1, otherwise it returns zero. α_1 is set to 0.3.

***Closest distance to a message's region** (α_2, $z_2^i(m_j)$)*:

$$z_2^i(m_j) = 1 - \frac{d^i(c_{m_j})}{5000}, \tag{4.10}$$

where $d^i(c_{m_j})$ is a function which calculates the shortest distance in meters to which vehicle i approaches the message's geographical coordinates c_{mj}. α_2 is set to 0.6.

***Data age** (α_3, $z_3^i(m_j)$)*:

$$z_3^i(m_j) = 0.99^{t_{mj}}, \tag{4.11}$$

where t_{mj} is the time elapsed since the message's generation time and α_3 is set to 0.1.

***Data priority** (p)*: we define three levels of data priority for m_j: $p \in \{1.0, 0.5, 0.1\}$.

For all data selection mechanisms the protocol described in 4.2.2.2 is used. Both *hello* messages and data messages are 2312 bytes large (the maximum allowed by 802.11p) and sent at 1 Hz. As explained, *hello* messages are sent in in parallel without synchronization with the application cycle. Each slot time is set to 10 milliseconds, which is an overestimate of the transmission time with a bit rate of 3 Mbit/s (the lowest data rate of the 802.11p standard).

Every vehicle begins the simulation with 10 data messages. Each message's geographical coordinates are set to the Cartesian point corresponding to 500 meters away from the vehicle in the opposite vehicle's direction. In this manner, we simulate vehicles that have already passed by the message's geographical region and now carry the message to other regions. The start age of messages is defined as a random number in $[0, 300]$ seconds. The three levels of data priority are assigned for each message according to lane ID number ln at which the vehicle begin in the simulation by: $ln \bmod 3$.

In Section 4.1, we explored a wide range o metrics to assess the differences in performance between prioritizing fairness or efficiency in the data selection. However, when considering broadcast communication, some metrics such as the percentage of vehicles that received some utility become less meaningful, since all vehicles receive utility by overhearing transmissions in the neighborhood. Therefore, we narrow down our metrics to the following list:

- ***Jain's fairness index***: defined as $(\sum_i^h x_i)^2/(h \sum_i^h x_i^2)$ (see [101]), where h is the total number of vehicles and x_i is the sum of utility gained by vehicle i. It indicates how well utility gains are distributed among vehicles. $1/h$ and 1 are the worst and best cases, respectively.
- ***Sum of utility gains***: the total utility sum gained by an arbitrary vehicle on average. It measures the overall performance of the dissemination approach.
- ***Utility per data message received***: shows the bandwidth utilization efficiency of the approach in terms of how much utility is gained per each data message received on average.
- ***Delay***: the average amount of time taken from the message's generation until it is received by vehicles that will be traveling to the area (1 km^2) to which the message relates.

In the following, we describe the results for an urban scenario with increasing network load levels (Section 4.2.3.1) and a highway scenario with increasing network densities (Section 4.2.3.2). Both scenarios were created with SUMO [83]. Therefore, they include realistic mobility patterns such as vehicle overtaking, lane changing, and rely on the well-known Krauß car-following mobility model.

4.2.3.1 Urban scenario with increasing controlled network load

We start by comparing FairDD with approaches 1–3 when increasing network load levels are allowed by the suppression mechanism presented in Section 4.2.2.2. The network load is defined as the percentage of neighbors that are allowed to send a data message, varying from 1 to 100%. Therefore, suppression is applied whenever the combined number of messages transmitted and received by a vehicle exceeds such defined percentage. In this case, the results for the approach with no suppression are fixed to 100%.

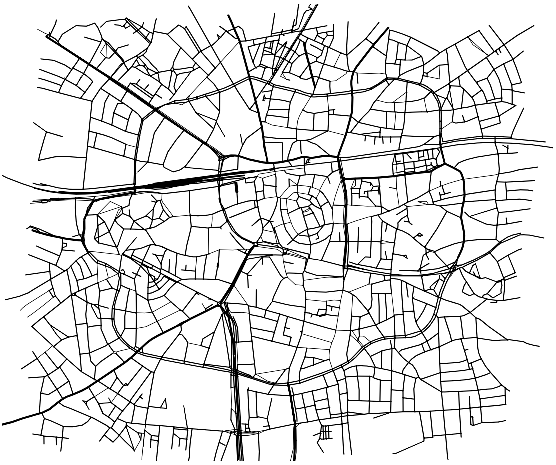


Figure 4.11: Urban scenario: map fragment of Enschede, The Netherlands

We consider a sparse urban scenario by taking a map fragment of the city of Enschede, The Netherlands (shown in Figure 4.11). This segment has an area of 3.5 x 4 km^2 and was retrieved with OpenStreetMaps [94]. The number of vehicles simultaneously moving increases linearly with time from 0 to 200, with a total of 300 generated. Vehicles' speeds vary from 0 to 100 km/h. Simulations consist of 20 runs of 300 seconds.

Figure 4.12(a) shows the results of applying the Jain's fairness index. Since it is broadcast communication, as the network load allowed increases more messages are sent and even vehicles with high accumulated utilities may still increase their utilities. Thus, the level of fairness tends to decrease. For all network load levels allowed, FairDD presents a higher fairness index compared with Altruistic. However, employing no selection shows an even higher value compared with FairDD. In fact, this is simply a result of the criteria used in this approach for the ordering of messages: messages with lower ID are always selected first and thus similar utility values are distributed.

Although with a higher fairness index, approaches with no data selection present lower values of the sum of utility gains as shown in Figure 4.12(b). In fact, even restricting the number of messages transmitted both FairDD and Altruistic achieved higher values of utility gain when compared with the approach with no suppression. FairDD and Altruistic present almost identical

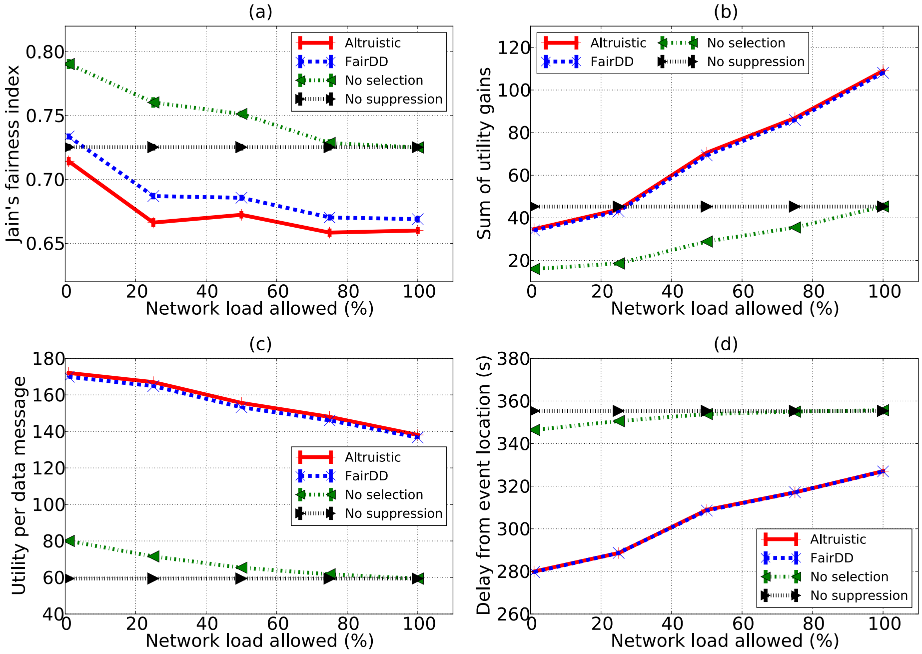


Figure 4.12: Results with 95% confidence intervals for increasing network load levels allowed by FairDD's suppression mechanism

results. Generally, when higher loads are allowed more messages are sent and, thus, more utility is gained on average per vehicle.

When looking at the utility per data message received (Figure 4.12(c)), there is a clear advantage when using data selection mechanisms. Such a ratio is more than the double compared with approaches with no data selection. With higher network load levels allowed, this ratio decreases as messages with less priorities are selected later on.

Finally, in Figure 4.12(d) the results for the average delay are presented. Applying data selection shows again a clear gain in performance compared to no use of selection. Since the utility is calculated based on the direction and final destination of vehicles, maximizing the utility gain of vehicles leads to a more quickly distribution of relevant data to those "interested" vehicles actually traveling towards the geographical region of the messages.

These results show the advantages of employing data selection: the network is utilized more efficiently in terms of utility gain per message received and relevant data is more quickly spread to interested vehicles. Also, FairDD presents a higher fairness index compared with Altruistic and yet maintaining equivalent results in the remaining parameters.

4.2.3.2 Highway scenario with increasing network densities

We consider a highway scenario with densities varying from 1 to 80 vehicles/km/lane. For all approaches employing data selection the maximum network load allowed is set to the minimum. Therefore, only data messages assigned to the earliest time slot (highest priority) are allowed to be sent.

The road is a 1-kilometer straight highway with two lanes in each road direction. Vehicles' speeds vary according to the density considered by following the Krauß mobility model, i.e., the higher the density is, the slower vehicles move. Simulations consist of 20 runs of 100 seconds.

Figures 4.13(a) and 4.13(b) show the results of applying the Jain's fairness index and the sum of utility gains for various densities. When higher densities are considered, more messages are transmitted due to the higher number of vehicles. Thus, the sum of utility gain per vehicle tends to increase. Also, in higher densities vehicles move more slowly, which gives more time for a complete data exchange among adjacent vehicles. For this reason, the index of fairness becomes higher and similar for all approaches. Notably, FairDD presents up to 20% higher index of fairness compared to Altruistic. Although the approach with no data selection presents higher values of fairness compared to approaches employing data selection, its values in terms of the sum of utility gains (efficiency) are considerably lower.

When no suppression is applied, the number of messages inserted into the network increases almost at the same rate as the density increases. Thus, the sum of utility gain is higher than all other approaches. However, as shown in Figure 4.13(c), the utility per message ratio of such an approach is 50% lower than approaches with data selection. Results for all approaches also show a decrease in the ratio up to an almost constant level as density increases. In fact, with more vehicles moving at slow speeds, the same group of vehicles tends to remain together. Thus, it is more likely that after some time messages with lower utility values are exchanged.

Finally, as observed in the previous section, employing data selection mechanisms brings the average delay down to almost 50% compared with approaches with no data selection (Figure 4.13(d)). In this scenario, such delay

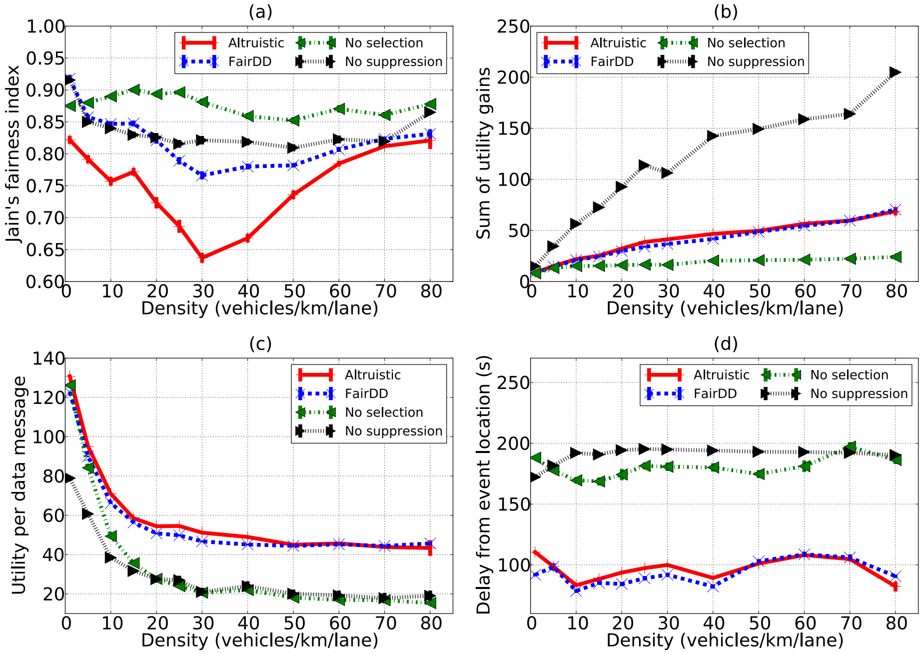


Figure 4.13: Results with 95% confidence intervals for increasing network densities

values remained around a constant upper-bound for all densities.

Overall, the advantages of employing data selection mechanisms remain valid for various network densities. More importantly, FairDD shows a gain up to 20% in fairness compared with Altruistic and yet it presents equivalent results in sum of utility gains, utility per message ratio and delay.

4.2.4 Conclusion

This section has presented FairDD, a periodic dissemination protocol that achieves a fair distribution of application data utility throughout the network while keeping the network load under a defined level. With simulation, we have shown that: (i) by employing data selection the network is utilized more efficiently in terms of utility gain per data message received and relevant data is more quickly spread to interested vehicles; and more importantly (ii) FairDD presents a higher fairness index compared with other approaches and yet it maintains a high level of bandwidth utilization efficiency.

4.3 A fair and adaptive data dissemination protocol

4.3.1 Introduction

In the previous section, we defined a protocol that periodically disseminates data fairly in the neighborhood according to a pre-defined application cycle. However, defining the periodicity of dissemination a priori is clearly not suitable when the number of vehicles in the neighborhood varies continually. Instead, the order of transmission for different messages should change adaptively according to the most up-to-date state of the neighborhood. To this end, we present in this section the FairAD: *Fair* and *A*daptive data *D*issemination protocol. Our key contributions can be summarized as follows:

- We present FairAD: a broadcast-based protocol that distributes data utility fairly over vehicles while *adaptively* controlling the network load. The protocol relies only on local knowledge to achieve fairness with concepts of Nash Bargaining from game theory. FairAD is a result of combining two independent lines of work, namely, the data selection mechanism discussed previously in Section 4.2 and the adaptive beaconing control proposed in [17, 77]

- We validate FairAD and other data selection approaches with simulations in large-scale networks. In particular, as urban scenario, we take a real map fragment from the Manhattan area in New York City, USA, including the shape of buildings that are used to model radio obstacles.

- We further demonstrate the applicability of FairAD by giving example of more realistic utility functions for two TIS applications: (i) parking-related and (ii) traffic information applications. We additionally study the effects when both applications are considered simultaneously in our performance evaluation.

- We perform real-world experiments with two vehicles moving in opposite directions in a highway at high speeds. We validate the behavior of FairAD and other data selection approaches and study aspects such as the average connectivity time, transmission range achieved, packet loss and throughput.

4.3.2 Fair and adaptive data dissemination

FairAD aims to achieve a *fair* distribution of data utility throughout the network while controlling the network load. It consists of two main components:

(i) a distributed fair data selection mechanism based on FairDD (Section 4.2) and (ii) an adaptive periodic protocol based on ATB [17, 77] to control the rate at which messages are broadcast into the network.

4.3.2.1 Data selection

Just as FairDD, FairAD relies on the Nash Bargaining [96] solution from game theory. This solution achieves a compromise between fairness and efficiency. A vehicle v_i employing FairAD independently stores its *local* knowledge of the neighborhood into two variables: utility matrix U and vector of accumulated utility c_i.

Let U be utility matrix for h vehicles and n data messages,

$$U = \begin{array}{c} \\ v_1 \\ v_2 \\ \\ v_h \end{array} \begin{array}{c} \begin{array}{cccc} m_1 & m_2 & \dots & m_n \end{array} \\ \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ u_{21} & u_{22} & \dots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{h1} & u_{h2} & \dots & u_{hn} \end{pmatrix} \end{array}, \tag{4.12}$$

where u_{ij} is given by (4.1). In matrix U, the utility value for each pair (v_i, m_j) is given. There are n distinct data messages to be sent in the neighborhood. For a message to appear in U, there is at least one vehicle that has not received it yet. If vehicle v_i already has message j, then $u_{ij} = 0$.

FairAD also takes into account the accumulated utility c_i of each vehicle v_i. In this way, a vehicle that gained more in previous opportunities will have a lower priority to increase its c_i in the next data exchange. Figure 4.14 shows the evolution of c_i when a random vehicle i moves in one of our simulation scenarios. The utility function considered takes into account the vehicle speed, distance to message's region and message age (detailed in Section 4.3.4.2). A vehicles starts receiving utility but as time goes by or as the vehicles changes its speed or direction, its accumulated utility c_i begins to fluctuate.

The data selection process defines in a distributed manner the next message each vehicles sends and its priority in terms of fairness, given the accumulated utility and messages carried by neighbors in the neighborhood. Each vehicle calculates its optimum solution locally, based on the information received from one-hop neighbors only. This process is defined by Algorithm 4.2. The input values U and $\vec{c}$ are the utility matrix and a vector containing the accumulated utility values c_i of each vehicle, respectively. The algorithm gives as output the

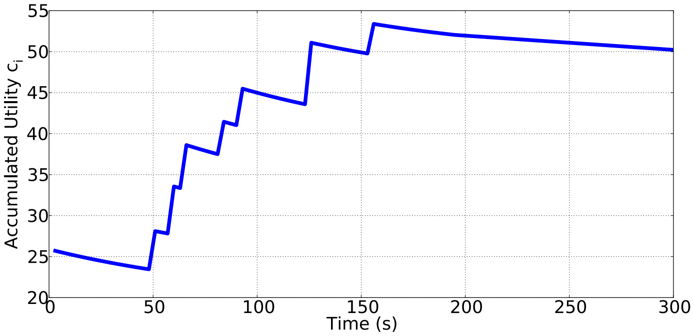


Figure 4.14: Example of the accumulated utility (c_i) concept for a random vehicle moving in Manhattan, New York City, USA

message selected m_t having the highest priority P among the messages carried by the local vehicle, where lower values of P indicate higher priority.

The core function is described in line 4. The Nash Bargaining solution maximizes the product of the sum of the utility gain u_{ij} and accumulated utility c_i of each vehicle. Therefore, in matrix U, message m_t maximizing $\prod_{i=1}^{h} [u_{ij} + c_i]$ will be selected. To guarantee that this product is higher when more neighbors are profiting, we set a lower bound $\varepsilon = 1$ for c_i. Each vehicle stops its search when it has the m_t of the current loop iteration r, where r represents the rank of the message with respect to other messages in the neighborhood. However, to prevent transmission redundancies when multiple vehicles have m_t, a small extra value $S_v\delta$ is considered for the final priority P (line 8), where δ is a constant value (e.g., 0.1) and S_v is the order of the local vehicle in the list of one-hop neighbors sorted by their distance to the location where m_t was generated. The goal is to give higher chance for vehicles farther away from the message's event location to broadcast the message first, thereby allowing for a quick data dissemination. Other vehicles carrying m_t but with lower priority could then cancel and reselect their messages.

Whenever a message is not selected, U is updated (lines 13–14) and the next optimum result is calculated in the following iteration. The final value of P lying in the interval $[0, 1]$ is defined in line 10. The maximum message rank r_{max} serves to limit the number of messages considered in each data selection in order to: (i) control how spread messages are in the interval $[0, 1]$; and (ii) prevent long processing time when a large number of messages is available in the

Algorithm 4.2 FairAD_DataSelection

```
Input: $U$, $\vec{c}$ // matrix and vector of accumulated utility
 1: $r \leftarrow 0$ // counter to define the final message rank
 2: $J \leftarrow \{0, 1, ..., n\}$
 3: while $U \neq \oslash$ and $r < r_{max}$ do
 4:     $t \leftarrow \arg\max_{j \in J} \prod_{i=1}^{h} [u_{ij} + c_i]$
 5:     if this vehicle has $m_t$ then
 6:         if number of neighbors with $m_t > 0$ then
 7:             sort vehicles by distance from event location
 8:             $r \leftarrow r + (S_v \delta)$ // $S_v$ is the order of this vehicle
 9:         end if
10:         $P \leftarrow \left( \frac{r}{r_{max}} \right)$
11:         return $m_t$, $P$ // message selected and its priority
12:     end if
13:     remove $m_t$ from $U$
14:     remove $t$ from $J$
15:     $r \leftarrow r + 1$
16: end while // no message selected, try again later
```

neighborhood. Reaching r_{max} and not selecting a message is an indication that this vehicles has messages with lower priority compared to its neighbors and can try later. The vehicle runs the algorithm again as soon as new information about the environment is received, as we describe in the following sections.

The complexity of Algorithm 4.2 is upper-bounded by the search of the maximum product in line 4. In the worst case, i.e., when $r_{max} = n$, in total $h \sum_{a=0}^{n} [n - a]$ operations are performed, where h and n are the number of vehicles and messages in the neighborhood, respectively. As the number of vehicles h is always limited by the transmission range employed by neighbors, the complexity comes down to $O(n^2)$.

4.3.2.2 Adaptive message intervals

We propose the use of Adaptive Traffic Beacon (ATB) [17, 77] as our means to control the rate at which messages are transmitted in the network. ATB is designed to ensure a congestion-free channel by preventing packet loss (collisions) while reducing the messages's end-to-end delay. To achieve its goal, ATB adaptively controls the *interval* between transmissions of a given vehicle by relying on two metrics: (i) the *channel quality* C and (ii) the *message priority* P.

The message priority P determines the importance of each message in the current network context, i.e., in the current set of neighbors. It allows messages with higher priority to be transmitted first. As proposed in the ATB architecture in [17, 77], P combines and weighs specific metrics, namely, the data age, distance to event source, distance to the next Road-Side Unit (RSU), and how well the information has already been disseminated. However, different applications may require different metrics to be considered. In addition, one aspect missing in this calculation is the different interests that vehicles might have in a certain message. To this end, we improve the calculation of P by considering our generalized *utility* function as described in Section 4.1.2. In this manner, we provide a flexible framework for applications to define which aspects to consider according to their specific needs. More importantly, we use our algorithm described in Section 4.3.2.1 to provide a fair distribution of utility among neighbors without compromising efficiency in terms of the total utility distributed. Therefore, P is the priority of the message selected by Algorithm 4.2 according to the Nash Bargaining principle.

The channel quality C combines three different network metrics in order to estimate the availability of channel resources as detailed in [17, 77]:

i) Number of collisions or bit errors K observed in the last time interval. It gives an estimate of the recent load on the channel:

$$K = 1 - \left(\frac{1}{1 + \#\,\text{collisions}} \right). \tag{4.13}$$

ii) The current Signal to Noise Ratio (SNR) as perceived in the last transmission estimates the current transmission quality. It is denoted as S:

$$S = \max \left\{ 0, \left(1 - \frac{\text{SNR}}{\text{max. SNR}} \right)^2 \right\}. \tag{4.14}$$

iii) Finally, number of neighbors N, i.e., neighborhood density, is used to predict the probability of other transmissions in the next time interval:

$$N = \min \left\{ \left(\frac{\#\,\text{neighbors}}{\text{max. }\#\,\text{neighbors}} \right)^2, 1 \right\}. \tag{4.15}$$

In order to give higher weight to metrics K and S, factor $\omega_C \geq 1$ is used to combine the three components as follows:

$$C = \frac{N + \left[\omega_C \left(\frac{S+K}{2}\right)\right]}{1 + \omega_C}. \tag{4.16}$$

The combination of both parameters C and P is given by (4.17). Smaller values of C and P represent a better channel and a higher priority, respectively. Therefore, when both values are zero $I = I_{min}$, i.e., the shortest interval allowed, where $I \in [I_{min}, I_{max}]$. The weight of each parameter is determined by factor ω_I. The quadratic form in both parameters C and P is used to quickly reduce I when the channel quality improves and/or when the message priority increases.

$$I = I_{min} + \left[(I_{max} - I_{min})(\omega_I C^2 + (1 - \omega_I)P^2)\right]. \tag{4.17}$$

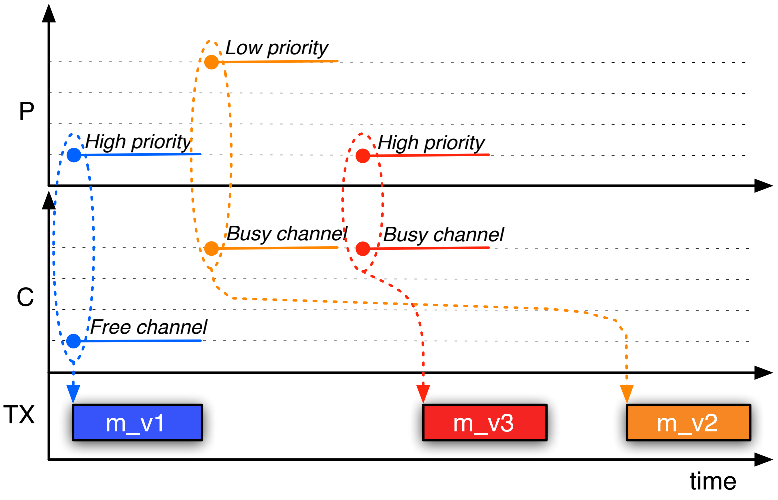


Figure 4.15: Overview of ATB

The overview of ATB is shown in Figure 4.15. In this example, vehicle v_1 sends message m_v_1 with both lower P and C values because of the high message's priority and currently free channel. As time goes by, vehicles v_2 and

v_3 find the channel busy. Due to a difference in their message priority, their transmissions are switched in time because of the higher priority given to message m_v_3.

4.3.2.3 Adaptive periodic protocol

We propose an adaptive protocol that continually reevaluates the next data message to be sent and its priority, whenever new information about the environment is received. Two types of messages are defined: *hello* messages and *data* messages.

As explained previously, the data selection mechanism proposed in Section 4.3.2.1 depends on the current contextual knowledge acquired by each vehicle to build matrix U. For this purpose, we define auxiliary *hello* messages that are broadcast continually by each vehicle. Each *hello* message sent by vehicle v_i contains a summarized list of data messages carried by v_i with information such as age and the geographical region where each message was generated. In addition, these messages include up-to-date information about the vehicle such as the vehicle's ID, direction, final destination and accumulated utility c_i. The information about the vehicle is always included in the header of each *hello* message. However, to guarantee an upper-bound for the processing time of Algorithm 4.2, the list size is kept under the maximum message size allowed by the underlying protocol, i.e., 802.11p. In such cases, vehicles are required to include in the list messages that are expected to be most important to other vehicles according to the data selection scheme. This is done by executing Algorithm 4.2 with only the messages carried by vehicle v_i, i.e., subset U_i, multiple times without repeating the messages chosen in each iteration until the maximum list size is reached. However, further study is required to determine the best criteria to select messages when exceeding the maximum limit size.

On the other hand, *data* messages carry the actual data distributed by the application. In contrast to *hello* messages, data messages are only scheduled when at least one neighbor can benefit from it, i.e., utility > 0. Therefore, if all neighbors already shared their messages and no new message is generated, then no more data messages are transmitted.

As defined in [103], vehicles shall be able to accommodate an architecture that supports a control channel (CCH) and multiple service channels (SCHs). Therefore, we define each type of message to be sent in a separate radio channel in order for *hello* messages not to interfere with the transmission of data messages. The transmission interval for both message types is defined according

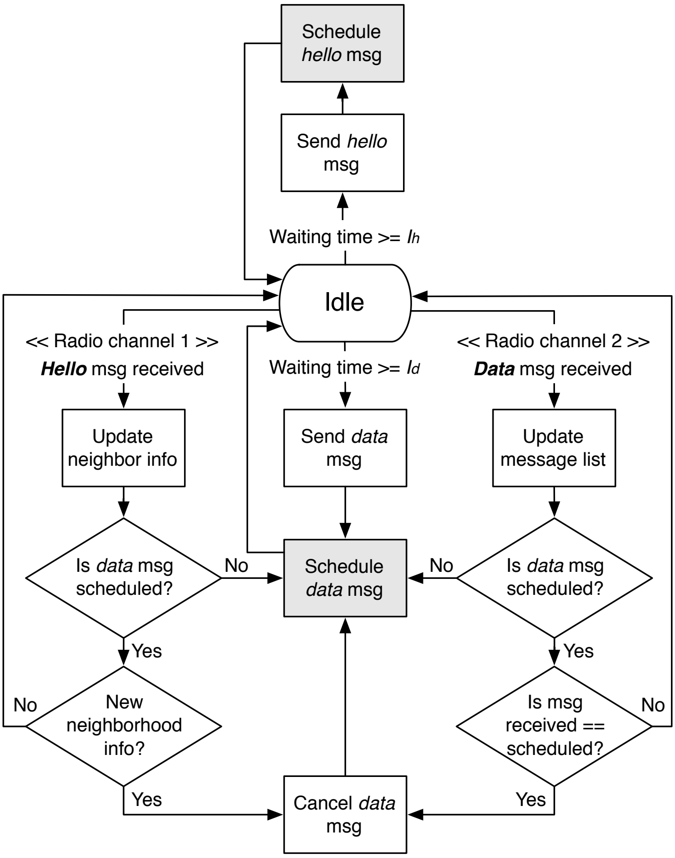


Figure 4.16: FairAD protocol diagram

to (4.17), where I_h and I_d are the intervals defined for *hello* and data messages, respectively. In particular, we define $\omega_I = 1$ for I_h. As *hello* messages are equally important, $\omega_I = 1$ guarantees that only the channel quality C is taken into account.

The complete protocol diagram is shown in Figure 4.16. The upper part of the diagram shows the process of scheduling and sending *hello* messages. Whenever I_h expires, a *hello* message is sent and a new one is scheduled. The lower part shows the decision tree for scheduling data messages. A new data message is immediately scheduled if no data message is already scheduled and a new *hello* message or data message is received from other neighbors. Every data message selection in the function *Schedule data msg* is done by Algorithm 4.2. The protocol also takes care of canceling and rescheduling messages if new data is available in the neighborhood as indicated by *hello* messages or if another neighbor farther away from the message's event location has already disseminated the data message scheduled. In this way, we guarantee an optimum message selection according to the most up-to-date contextual information. When rescheduling, the new interval defined refers always to the last time a message was sent, thereby respecting the condition $I \in [I_{min}, I_{max}]$. Since *hello* messages are sent at a low frequency, i.e., at least 1 Hz, this measure does not incur excessive additional processing.

4.3.3 Performance evaluation

The performance evaluation of FairAD is carried out by means of simulations. Our goal is two-fold: (i) verify the benefits of employing data selection mechanisms in the adaptive periodic data dissemination protocol described in Section 4.3.2.3 and (ii) compare FairAD's data selection with other data selection approaches, namely:

- ***Altruistic***: based on [60], it maximizes the total utility gain for all neighbors as a whole. Thus, it does not consider individual interest. It gives an upper-bound in terms of efficiency for individual message selections.
- ***Max-min***: maximizes the utility of vehicles with the lowest accumulated utility. It is an alternative to Nash Bargaining with respect to achieving fairness [99]. It gives an upper-bound in terms of fairness for individual message selections.
- ***No selection***: no utility is considered when selecting a data message. We simply define that messages with lower ID are sent with higher priority.

We use the Veins framework [90] version 2.0-rc2, which is based on both OMNeT++ 4.2.2 [80] event-driven network simulator and SUMO [83] for road traffic microsimulation. Veins provides realistic models for the 802.11p DSRC PHY and MAC layers, including multi channel operation required by our adaptive protocol in FairAD. At the same time, SUMO allows the creation of scenarios that include realistic mobility patterns such as vehicle overtaking, lane changing, and rely on the well-known Krauß car-following mobility model.

Table 4.3: Simulation parameters

Physical layer	Frequency band	5.88, 5.89 GHz
	Bandwidth	10 MHz
	Transmission range	$\sim$100 m
	Tx power	10 mW
	FSPL exponent α	2.2
	Receiver sensitivity	-85 dBm
	Thermal noise	-110 dBm
	Bit Error Rate (BER)	Based on [93]
Link layer	Bit rate	18 Mbit/s
	CW	[15,1023]
	Slot time	13 μs
	SIFS	32 μs
	DIFS	58 μs
FairAD	r_{max}	5
	δ	0.1
	max. SNR (S)	50 dB
	max. # neighbors (N)	50
	ω_C	2
	I_{min} (*hello* msg)	1 s
	I_{max} (*hello* msg)	5 s
	ω_I (*hello* msg)	1
	I_{min} (data msg)	30 ms
	I_{max} (data msg)	60 s
	ω_I (data msg)	0.5
	Max. msg list size in *hello*	100
Scenarios	Data message size	2312 bytes
	Initial # messages	5
	# runs	30

The complete list of simulation parameters is shown in Table 4.3. The parameters for the PHY and MAC layers are defined in such a way that complies with the 802.11p standard. We use channels 5.88 and 5.89 GHz for *hello* and data messages, respectively. In FairAD, we choose $r_{max} = 5$ to provide a large separation in time between messages selected by different vehicles in the interval $[I_{min}, I_{max}]$ and $\delta = 0.1$ to let vehicles farther away from the message's event location broadcast first. Since *hello* and data messages are used for different purposes, we set a different interval $[I_{min}, I_{max}]$ for each type. On the one hand, *hello* messages should be always broadcast to provide neighborhood awareness. Therefore, we limit the range to $[1, 5]$. On the other hand, the interval for data messages should be large enough to allow for a separation in time between messages of different priorities. Hence, we set this interval to $[0.03, 60]$, as proposed in [17, 77]. We also set a different value to ω_I for each message type, namely, $\omega_I = 1$ and $\omega_I = 0.5$ for *hello* and data messages, respectively. $\omega_I = 0.5$ assigns equal importance to both channel quality C and message priority P. Giving a higher weight to P is particularly useful for the evaluation of different data selection mechanisms, since differences in priority will be quickly reflected in the interval assigned.

The utility function u_{ij} is defined as the product:

$$u_{ij} = \beta \left(\alpha_1 z_1^i(m_j)\right) \left(\alpha_2 z_2^i(m_j)\right) \left(\alpha_3 z_3^i(m_j)\right), \tag{4.18}$$

which is composed by the contextual knowledge functions:

Vehicle direction **(**α_1**,** $z_1^i(m_j)$**)**: if the vehicle v_i is going towards the data message's geographical region, z_1^i returns 1, otherwise it returns 0.1. α_1 is set to 3.

Closest distance to a message's region **(**α_2**,** $z_2^i(m_j)$**)**:

$$z_2^i(m_j) = 1 - \frac{d^i(c_{m_j})}{\sqrt{x_{max}^2 + y_{max}^2}}, \tag{4.19}$$

where $d^i(c_{m_j})$ is a function which calculates the shortest distance in meters to which vehicle v_i approaches the message's geographical coordinates c_{mj} and x_{max}, y_{max} are the maximum x and y cartesian values of the scenario being considered. α_2 is set to 6.

Data age **(**α_3**,** $z_3^i(m_j)$**)**:

$$z_3^i(m_j) = 0.99^{t_{mj}}, \tag{4.20}$$

where t_{mj} is the time elapsed since the message's generation time and α_3 is set to 1.

Data priority **(**β**)**: we define three levels of data priority for m_j: $\beta \in \{1.0, 0.5, 0.1\}$. Note that this is a fixed value defined by the application and different from the message priority P defined in Algorithm 4.2.

Every vehicle begins the simulation with 5 data messages. Each message's geographical coordinates are set to the Cartesian point corresponding to 500 meters away from the vehicle in the opposite vehicle's direction. In this manner, we simulate vehicles that have already passed by the message's geographical region and now carry the message to other regions. The start age of messages is defined as a random number in $[0, 300]$ seconds. The three levels of data priority are assigned for each message according to lane ID number ln at which the vehicle begin in the simulation by: $ln \bmod 3$.

In the previous sections, we have relied on the sum of utility gains as means to evaluate the data selection approaches in terms of efficiency. This was possible since the number of transmissions was the same in both cases of the basic protocol for pairs of vehicles in Section 4.1.4 and the synchronized protocol described in Section 4.2.2.2. However, when exploring adaptive data selection decisions such as with FairAD, the number of transmissions may vary according to the priority defined by each data selection approach. For this reason, in the following evaluation we focus rather on the utility per data message received which eliminates the factor of vehicles receiving a different number of messages. To evaluate the FairAD's ability to control the network load, we introduce the number of transmissions as metric. The complete list of metrics is defined as follows:

- ***Jain's fairness index***: calculated each time a vehicle selects and sends a data message; defined as $(\sum_i^h c_i)^2/(h\sum_i^h c_i^2)$ (see [101]), where h is the number of vehicles in the neighborhood and c_i is the accumulated utility of each neighbor v_i after receiving the message selected. It indicates how well data utility is distributed among vehicles. $1/h$ and 1 are the worst and best cases, respectively.

- ***Utility per data message received***: shows the bandwidth utilization efficiency of the approach in terms of how much utility is gained per each data message received on average.

- ***Total number of transmissions***: the total number of *data* messages transmitted on average by an arbitrary vehicle. It indicates how well the adaptive periodic protocol copes with changes in the network load.

- ***Delay***: the average amount of time taken from the message's generation until it is received by vehicles that will be traveling to the area to which the message relates. The area radius is defined as: $\frac{1}{4}\sqrt{x_{max}^2 + y_{max}^2}$, where x_{max}

and y_{max} are the maximum x and y cartesian values of the scenario being considered.

4.3.3.1 Urban scenario with increasing data message list sizes

In the following, we compare FairAD with approaches 1–3 when increasing the maximum size allowed for the message list that is included in *hello* messages. In this way, we evaluate the impact of how much awareness is necessary for vehicles to efficiently select and distribute data in the neighborhood. We vary the maximum number of messages from 0 to 20, where zero means that vehicles are limited to select messages based only on its local list. In this scenario, vehicles receive on average from 30 to 40 messages in the simulation time.

We consider a sparse urban scenario by taking a map fragment of the city of Enschede, The Netherlands (shown in Figure 4.11). This segment has an area of 3.5 x 4 km^2 and was retrieved with OpenStreetMaps [94]. The number of vehicles simultaneously moving increases linearly with time from 0 to 200, with a total of 300 generated. Vehicles' speeds vary from 0 to 100 km/h. Simulations consist of 30 runs of 300 seconds.

Figure 4.17(a) shows the results of applying the Jain's fairness index. Specially for FairAD and Max-min, the level of fairness increases as more information about the messages available in the neighborhood is known. As these approaches focus on fairness, more contextual information enables a more precise data selection that will please individual interests of vehicles.

With more messages in the list, vehicles are also able to transmit fewer messages and more efficiently as shown in Figures 4.17(b) and 4.17(c). Notably, Altruistic and FairAD are able to increase efficiency and choose messages with highest utility to be distributed, thereby outperforming Max-min and No selection in terms of the utility per message ratio. In contrast, Max-min aims only to increase the utility gain of vehicles with lowest accumulated utility, which compromises the efficiency in terms of the total utility distributed.

Finally, Figure 4.17(d) shows the results for the average delay. All approaches benefit from increasing the message list size, as vehicles can keep track of messages that have already been received by other vehicles and avoid duplicate broadcasts. However, applying data selection leads to lower delay values compared to No selection. Since the direction and final destination of vehicles are considered in the utility calculation, mechanisms considering utility in the data selection are able to distribute messages more quickly to "interested" vehicles that are actually traveling towards the message's event region.

Overall, increasing the message list size allows for data selection methods

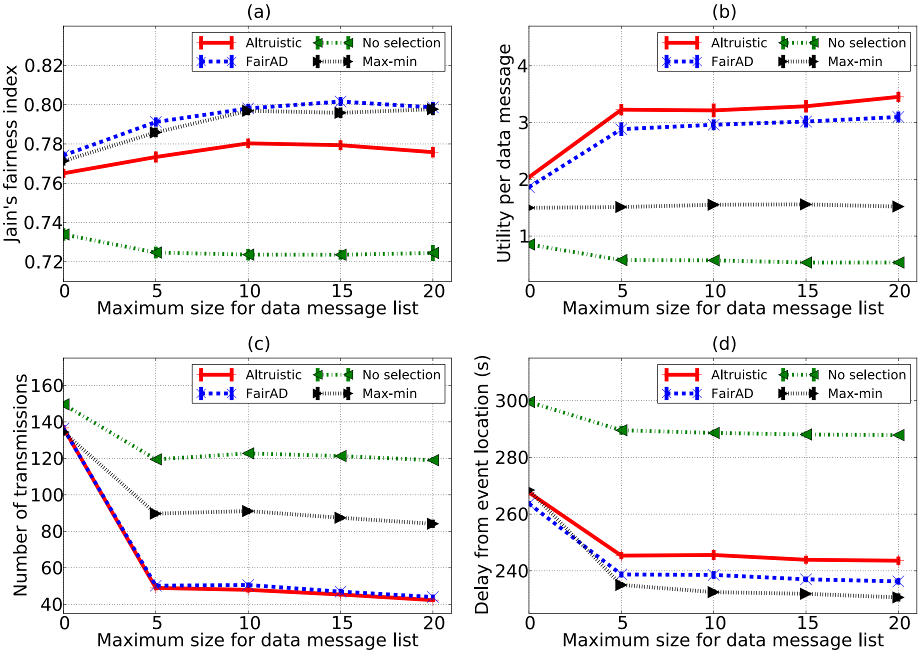


Figure 4.17: Results with 95% confidence intervals for increasing sizes of the data message list included in *hello* messages

to better achieve their specific goals of efficiency (Altruistic), fairness (Max-min), and both (FairAD). This is true even for small list sizes, thanks to the policy adopted to include messages that are predicted to be most beneficial to other neighbors. Notably, FairAD achieves the best balance between fairness and efficiency.

4.3.3.2 Highway scenario with increasing network densities

We consider a highway scenario with densities varying from 5 to 100 vehicles/km/lane. Simulations consist of 30 runs of 100 seconds. The road is a 1-kilometer straight highway with two lanes in each road direction. The speed of vehicles reaches a maximum of 120 km/h in very sparse scenarios. When increasing the density, the speed varies according to the Krauß mobility model,

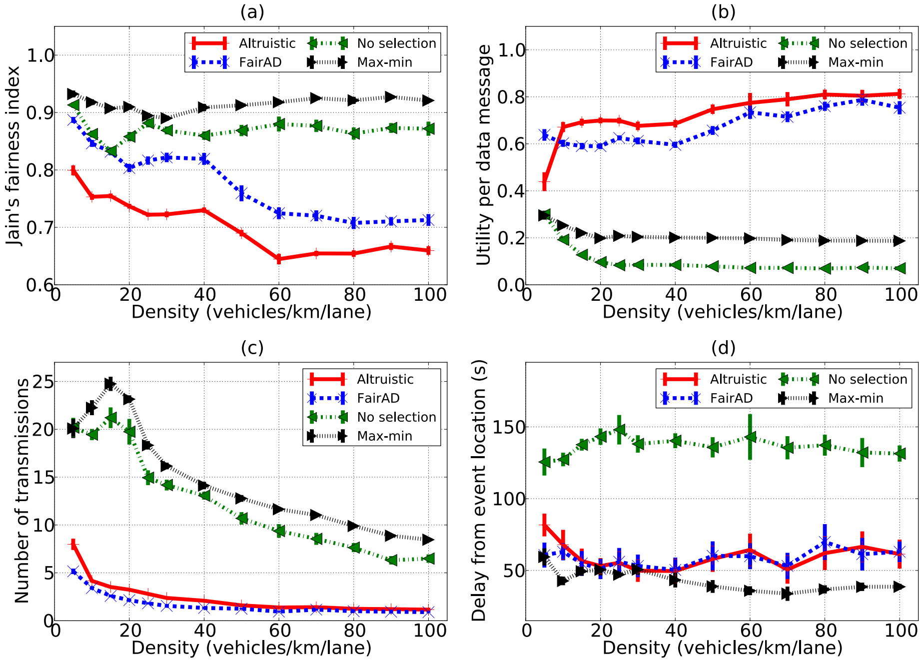


Figure 4.18: Results with 95% confidence intervals for increasing network densities

i.e., the higher the density is, the slower vehicles move. Note that the space between vehicles varies, with small traffic jams occurring in each road end. Thus, the number of data exchanges and, consequently, results do not present a perfect linear behavior with increasing densities.

Figure 4.18(a) shows the results of applying the Jain's fairness index for various densities. FairAD and Max-min show up to 15% and 25% higher fairness index compared to Altruistic, respectively. No selection shows a higher value compared to FairAD, which is simply a result of the criteria used by No selection to assign messages' priority: messages with lower ID are selected first and thus similar utility values are distributed.

As the density increases, the adaptive protocol based on ATB properly controls the network load by increasing the time interval between transmissions with higher values of C in Equation (4.17). This results in a lower number of transmissions for all methods (Figure 4.18(c)). The number of transmissions varies between data selection methods due to their differences in selecting the message priority P of the same equation. With fewer messages transmitted, methods that aim at efficiency such as Altruistic and FairAD show an improvement in the utility per message ratio and outperform Max-min and No selection, as shown in Figure 4.18(b). However, this comes at the cost of decreasing their performance in terms of fairness.

Finally, as pointed out previously, employing data selection mechanisms clearly helps decrease the average delay compared with No selection, as shown in Figure 4.18(d).

These results show that FairAD is able to adaptively distribute data utility fairly over vehicles and properly control the network load for increasing network densities.

4.3.4 Applications

Up to this point, we have considered basic utility functions with contextual information that may be common to a variety of applications. In the following, we elaborate on the utility functions of two specific basic applications: one related to (i) parking information; and another related to (ii) traffic information. In addition to evaluating FairAD with more realistic functions, we are interested in evaluating the impact of running both applications simultaneously.

These functions return values that fall in the interval $[1, 8]$, which provides enough room for utility disparity between vehicles depending on their mobility and data context. Also, we choose multiplication as the means to combine different parameters in the utility functions in order to tighten their dependence and allow for a wider variety of values between different vehicles' context. The functions that we present here are by no means sufficient to represent a full parking or traffic information application. However, we argue that the contextual information that we propose may be incorporated in more complex applications of each type alongside other factors.

4.3.4.1 Parking information

We propose a parking related application that disseminates information about the parking places currently available in a city. To this end, we propose the use of the driver's intention to park the vehicle and the age of the parking information.

The utility function u_{ij}^{P} is defined as:

$$u_{ij}^{p} = \begin{cases} 1 & \textbf{if } \text{the vehicle will } \textit{not} \text{ park;} \\ 2\; z_1^i(m_j)\; z_2^i(m_j). & \textbf{if } \text{the vehicle will park.} \end{cases} \tag{4.21}$$

u_{ij}^{P} returns a value that falls in the interval $[1, 8]$, where both contextual knowledge functions $z_1^i(m_j)$ and $z_2^i(m_j)$ return values in the interval $[1, 2]$. Effectively, vehicles that have the intention to park always receive higher values, namely, from the interval $[2, 8]$. $z_1^i(m_j)$ and $z_2^i(m_j)$ are defined as follows:

Distance to vehicle's parking destination ($z_1^i(m_j)$):

$$z_1^i(m_j) = 2 - \frac{d_i^P(c_{m_j})}{5000} \tag{4.22}$$

where $d_i^P(c_{m_j})$ is a function which calculates the distance in meters between the vehicle's final parking destination and the coordinates of the parking place where the message was generated c_{m_j}. We assume that only parking information up to 5 km of distance are interesting for a vehicle: $d_i^P(c_{m_j}) \in [0, 5000]$, based on location-based service requirements defined in [18]. With distances farther than 5000, $z_1^i(m_j)$ is given the minimum value of 1.

Data age ($z_2^i(m_j)$):

$$z_2^i(m_j) = 1 + 0.99^{t_{mj}}, \tag{4.23}$$

where t_{mj} is the time elapsed since the message's generation time. Effectively, this function return values near the minimum value of 1 when t_{mj} is close to 300 seconds.

4.3.4.2 Traffic information

We additionally propose a traffic related application that disseminates information about the current traffic situation in the city. Each vehicle periodically generates messages with their own speed and geographical coordinates. By sharing these messages, the speed profile of different regions of the city can be built. Although data aggregation could certainly be used to merge different messages as proposed in [79], this is out of the scope of this thesis. We rather concentrate here on combining the vehicles's speed, distance, and age of information into a common utility function.

The utility function u_{ij}^T is defined as the product:

$$u_{ij}^T = z_2^i(m_j)\, z_3^i(m_j)\, z_4^i(m_j). \tag{4.24}$$

u_{ij}^T returns a value that falls in the interval $[1, 8]$, where each contextual knowledge function returns values in the interval $[1, 2]$. $z_2^i(m_j)$ is used as defined previously for the parking information application, whereas $z_3^i(m_j)$ and $z_4^i(m_j)$ are defined as follows:

***Distance to vehicle (*$z_3^i(m_j)$*)*:**

$$z_3^i(m_j) = 1 - \frac{(d_i^T(c_{m_j}))^2}{6245000} + \frac{d_i^T(c_{m_j})}{1249} \tag{4.25}$$

where $d_i^T(c_{m_j})$ is the distance between the current vehicle's position and the coordinates c_{m_j} where the message was generated. This function forms an inverted parabola with roots at points 0 and 5000 in the x-axis. On the one hand, messages containing information regarding distances immediately close to the vehicle are not interesting, since the driver may be aware of the traffic situation without resorting to information from other vehicles. On the other hand, information regarding excessively long distances can become outdated or can be unimportant if the vehicle never actually reach that region. Therefore, we define that distances near the center point 2500 in the x-axis return the highest values. We assume that only traffic information up to 5 km of distance are interesting for a vehicle: $d_i^T(c_{m_j}) \in [0, 5000]$, based on road congestion information requirements defined in [18]. With distances farther than 5000, $z_3^i(m_j)$ is given the minimum value of 1.

***Traffic speed (*$z_4^i(m_j)$*)*:**

$$z_4^i(m_j) = 2 - \frac{s_{m_j}}{36} \tag{4.26}$$

where s_{m_j} is the speed of the vehicle that generated message m_j. We assume that speeds vary in meters per seconds in the interval $[0, 36]$. In this function, more importance is given to low speed values, as these indicate potential traffic jams in the city. Speeds higher than 36 m/s are given the minimum value of 1.

4.3.5 Real-world experiments and impact of applications

We further evaluate the performance of FairAD with both real-world experiments and simulations. Our goal is two-fold: (i) verify the correctness and feasibility of employing different data selection mechanisms in real-world environments and (ii) compare FairAD's data selection in large scale simulation scenarios against other data selection approaches when the applications proposed are employed. The following data selection mechanisms are used as comparison:

- ***Altruistic***: based on [60], it maximizes the total utility gain for all neighbors as a whole. Thus, it does not consider individual interest. It gives an upper-bound in terms of efficiency for individual message selections.
- ***Max-min***: maximizes the utility of vehicles with the lowest accumulated utility. It is an alternative to Nash Bargaining with respect to achieving fairness [99]. It gives an upper-bound in terms of fairness for individual message selections.
- ***No selection***: no utility is considered when selecting a data message. We simply define that messages with lower ID are sent with higher priority.

Since we have already shown the ability of FairAD to control the network load for increasing densities, we exclude the total number of transmissions from our evaluation metrics. Thus, the following list is considered:

- ***Jain's fairness index***: calculated each time a vehicle selects and sends a data message; defined as $(\sum_i^h c_i)^2/(h\sum_i^h c_i^2)$ (see [101]), where h is the number of vehicles in the neighborhood and c_i is the accumulated utility of each neighbor v_i after receiving the message selected. It indicates how well data utility is distributed among vehicles. $1/h$ and 1 are the worst and best cases, respectively.
- ***Utility per data message received***: shows the bandwidth utilization efficiency of the approach in terms of how much utility is gained per each data message received on average.
- ***Delay***: the average amount of time taken from the message's generation until it is received by vehicles that will be traveling to the area to which the message relates. The area radius is defined as: $\frac{1}{4}\sqrt{x_{max}^2 + y_{max}^2}$, where x_{max} and y_{max} are the maximum x and y cartesian values of the scenario being considered.

Table 4.4: Experiment parameters

Physical Layer	Frequency band	5.88 GHz
	Bandwidth	10 MHz
	Tx power	20 dBm
Link Layer	Bit rate	6 Mbit/s
FairAD	I_{min} (*hello* msg)	1 s
	I_{max} (*hello* msg)	1 s
	I_{min} (data msg)	30 ms
	I_{max} (data msg)	60 s
	ω_I (data msg)	0
Scenarios	Relative speed	~225 km/h
	Data message size	2312 bytes
	Initial # messages	250

4.3.5.1 Real-world experiments

In our real-world experiments, we use two vehicles equipped with a 802.11p gateway. The Atheros AR5413 802.11a radio is used with a modified driver to comply with 802.11p standard in terms of frequency band, channel width, and bit rate. We implement the FairAD protocol and the other data selection methods used for comparison in a Perl script. The standard socket library is used to broadcast UDP packets in their maximum size before fragmentation, namely, 1472 bytes. In total, around 2312 bytes are sent when taking into account extra overhead in the MAC and PHY layers. Since the experiments consist of only two vehicles, the parameters related to channel load used by FairAD are unnecessary. Specifically, the r_{max}, δ, and ω_C parameters are omitted. Therefore, we focus on the priority of messages with $\omega_I = 0$. The experiment parameters are summarized in Table 4.4.

Our scenario consists of two vehicles driving in opposite directions in one piece of the A35 highway that links the cities of Enschede and Hengelo in The Netherlands. During the day of experiments, the weather humidity was 92% with temperature around +4 degrees Celsius. Each vehicle begins in a junction point located near one of the two cities and drives 5.6 kilometers until it reaches the other junction point. The average relative speed between the vehicles is 225 km/h. In total, this process is repeated 12 times, where 4 times is reserved for experiment 1 and 8 times for experiment 2 (2 times for each data selection). Each experiment is described as follows:

Table 4.5: Experiment results

	Average	Standard deviation
Connectivity time	7.62 s	1.31 s
TX range achieved	254.1 m	25.15 m
Messages per sec. exp. 1	40.21	1.47
Messages per sec. exp. 2	17.17	1.40
Throughput exp. 1	743.8 kbit/s	27.22 kbit/s
Throughput exp. 2	317.6 kbit/s	25.98 kbit/s
Packet loss exp. 1	75.39 %	0.55 %
Packet loss exp. 2	19.42 %	11.95 %

- ***Experiment 1***: consists of one sender and one receiver only, without any sort of data selection. The sender broadcasts messages continuously with no interval between the messages. Our goal is to evaluate how much data can be received correctly when two vehicles are moving at high speeds in opposite direction.

- ***Experiment 2***: consists of comparing each data selection method. All methods are run in the same protocol as shown in Figure 4.16. *Hello* messages are sent at a fixed rate of 1 Hz and data messages are sent in the interval $\in [0.030, 60]$ seconds, as proposed in [77]. Each vehicle includes its updated accumulated utility value c_i in each message transmitted and keeps track of the accumulated utility of the other vehicle in order to make data selection decisions. After each messages is received, the priority of the message scheduled is updated and the waiting interval is defined according to each data selection method. To provoke a conflict of interests and test the behavior of each data selection method, we define that each message worths 10 of utility to one vehicle and only 1 to the other. Each vehicles begins with 250 messages to be exchanged.

The results that are common to both experiments are averaged and shown in Table 4.5. Due to the high relative speed between the vehicles and the average of 254.1 meters of transmission range achieved, the average time of connectivity is limited to only 7.62 seconds. In experiment 2, the throughput achieved is lower than with experiment 1 due to the minimum interval of 30 ms between every two transmissions performed by a vehicle. The packet loss is also lower with experiment 2, since one vehicle only begins exchanging data with another after it has correctly received a *hello* message. Figure 4.19 shows a sample of the

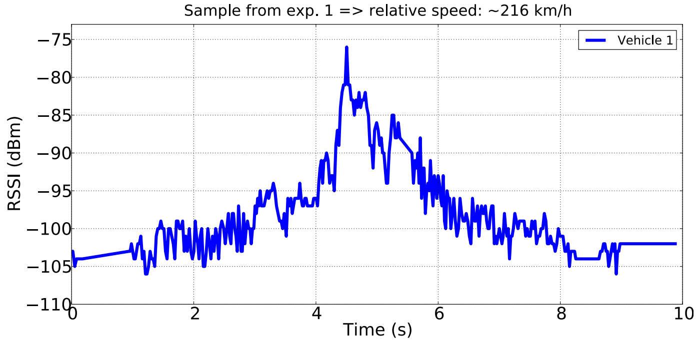


Figure 4.19: The evolution of the received signal strength during one data exchange in experiment 1

received signal strength when running experiment 1. In this sample, the connectivity time is around 10 seconds with the strongest peak lying in the center around 5 seconds when the vehicles pass by each other.

In Figure 4.20, we compare the behavior of each data selection method along time during data exchanges performed in our experiments. Since one vehicle receives 10 worth of utility and the other only 1, when employing Altruistic only one vehicle broadcasts messages (Figure 4.20(a)). For this reason, only vehicle 1 accumulates utility gains during the data exchange. With an opposite behavior, Max-min aims always to compensate differences in utility gains to achieve an equal utility gain in both vehicles as shown in Figure 4.20(c). FairAD aims at not only fairness but also efficiency in terms of the total utility distributed. Therefore, the compensation is limited and a compromise between both goals is achieved along time (Figure 4.20(b)). Finally, when no selection mechanism is used, a poor result can be achieved (Figure 4.20(d)). In particular, the latter result represents the worst case in terms of efficiency, since No selection chooses messages with the lowest IDs, which in this case are the ones with lowest utility. Since the results are shown from the point of view of vehicle 1, there are some negative fluctuations in the accumulative utility of vehicle 2 (c_2). This is explained by the fact that vehicle 1 keeps track of c_2 by increasing it every time a new message is sent. Since not every message is received correctly by vehicle 2, c_2 is corrected every time vehicle 2 sends a new *hello* message.

Figure 4.21 shows the average results in terms of fairness index and utility per message received for all runs of experiment 2. Altruistic clearly presents

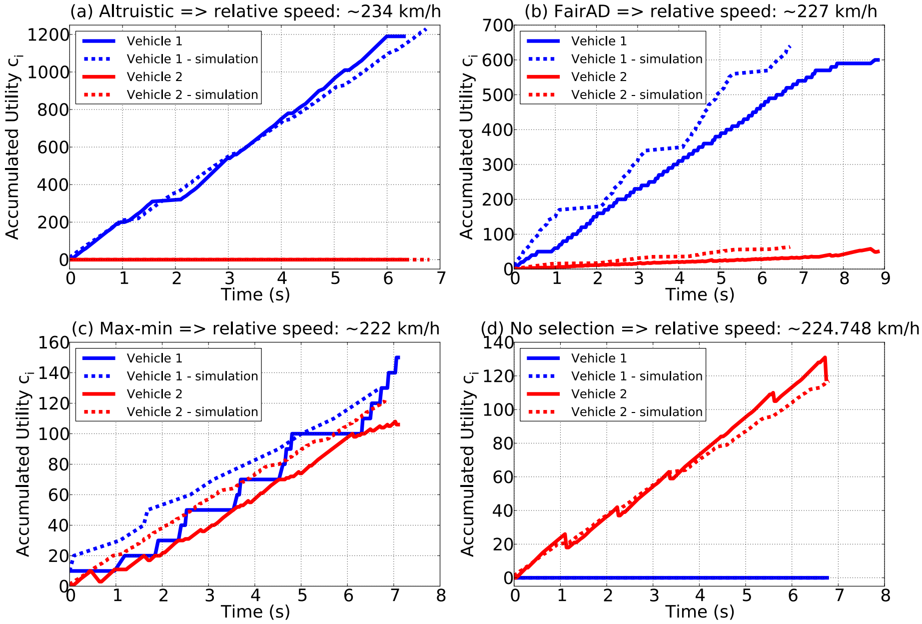


Figure 4.20: The behavior of each data selection method over time in both real-world experiments and simulations

the best result in terms of efficiency at the cost of having the worst fairness index. Conversely, Max-min achieves the best result in terms of fairness and a poor result in terms of efficiency. The dashed lines in Figure 4.21(a) indicate the minimum and maximum achievable values for the fairness index when only two vehicles are present.

All results above are in line with the expected behavior of each method, given their individual goals. In both Figures 4.20 and 4.21, we additionally verify that our simulation implementation represents a proper matching of the real-world experiments. This serves to strengthen the confidence in using our simulation implementation for large-scale scenarios as described in the next section. In the following, all simulation parameters are adjusted to match the real-world experiment results. In particular, the minimum transmission interval I_{min} is set to 50 ms in order to consider the additional overhead introduced by the application layer in the gateway before sending down broadcast mes-

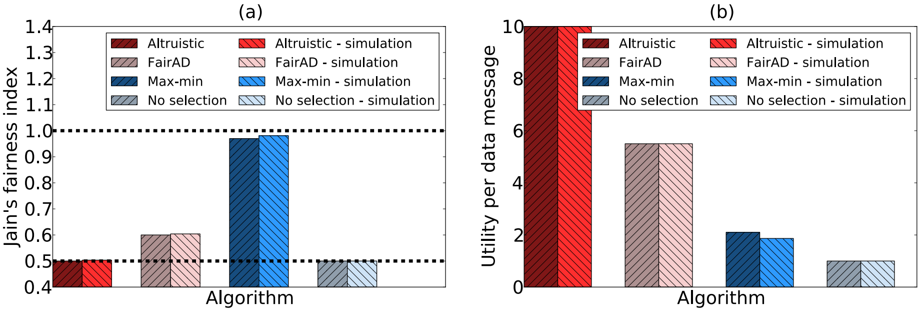


Figure 4.21: The Jain's fairness index and utility per message received averages for both real-world experiments and simulations

sages. The only difference between the simulation parameters used to validate our real-world experiments and the ones used in our large-scale simulations is with regard to the power level used. For the validation of our real-world experiments, a power level of 20 dBm was used to match the power level used by our gateway. However, for the sake of scalability, a lower transmission range was preferred in our larger-scale simulations, as summarized in Table 4.6.

4.3.5.2 Simulation

In our simulations, we evaluate the impact on each data selection method when considering both applications defined in Section 4.3.4 in large-scale scenarios. We use the Veins framework [90] version 2.0-rc2, which is based on both OMNeT++ 4.2.2 [80] event-driven network simulator and SUMO [83] for road traffic microsimulation. Veins provides realistic models for the 802.11p DSRC PHY and MAC layers, including multi channel operation required by our adaptive protocol in FairAD. At the same time, SUMO allows the creation of scenarios that include realistic mobility patterns such as vehicle overtaking, lane changing, and rely on the well-known Krauß car-following mobility model.

The complete list of simulation parameters is shown in Table 4.6. The parameters for the PHY and MAC layers are defined in such a way that complies with the 802.11p standard. We use channels 5.88 and 5.89 GHz for *hello* and data messages, respectively. In FairAD, we choose $r_{max} = 5$ to provide a large separation in time between messages selected by different vehicles in the interval $[I_{min}, I_{max}]$ and $\delta = 0.1$ to let vehicles farther away from the message's

Table 4.6: Simulation parameters

Physical Layer	Frequency band	5.88, 5.89 GHz
	Bandwidth	10 MHz
	Transmission range	$\sim$100 m
	Tx power	10 mW
	FSPL exponent α	2.5
	Obstacle model	Defined in [92]
	Receiver sensitivity	-90 dBm
	Thermal noise	-110 dBm
	Bit Error Rate (BER)	Based on [93]
Link Layer	Bit rate	6 Mbit/s
	CW	[15,1023]
	Slot time	13 μs
	SIFS	32 μs
	DIFS	58 μs
FairAD	r_{max}	5
	δ	0.1
	max. SNR (S)	50 dB
	max. # neighbors (N)	50
	ω_C	2
	I_{min} (*hello* msg)	1 s
	I_{max} (*hello* msg)	5 s
	ω_I (*hello* msg)	1
	I_{min} (data msg)	50 ms
	I_{max} (data msg)	60 s
	ω_I (data msg)	0.5
	Max. msg list size in *hello*	100
Scenarios	Data message size	2312 bytes
	Initial # messages	5
	# runs	30

event location broadcast first. Since *hello* and data messages are used for different purposes, we set a different interval $[I_{min}, I_{max}]$ for each type. On the one hand, *hello* messages should be always broadcast to provide neighborhood awareness. Therefore, we limit the range to $[1, 5]$. On the other hand, the interval for data messages should be large enough to allow for a separation in time between messages of different priorities. Hence, we set this interval to $[0.05, 60]$, where the minimum of 50 ms is used to match our real-world experiments, as explained in the previous section. We also set a different value to ω_I for each message type, namely, $\omega_I = 1$ and $\omega_I = 0.5$ for *hello* and data messages, respectively. $\omega_I = 0.5$ assigns equal importance to both channel quality C and message priority P. Giving a higher weight to P is particularly useful for the evaluation of different data selection mechanisms, since differences in priority will be quickly reflected in the interval assigned.

In the following sections, we present the results of running each data selection method in both urban and highway scenarios. We consider the following behavior for each combination of applications proposed:

- ***Parking***: each vehicle begins with 5 messages containing information about fictitious parking places that they have passed by before the beginning of the simulation. The locations of these parking places are defined as the coordinates of 500 meters towards the opposite heading direction vector of the vehicle. We also define that half of the vehicles will eventually park in their final geographical coordinates of their mobility traces. Finally, the start age of messages is defined as a random number in the interval $[0, 300]$ seconds.
- ***Traffic***: each vehicle begins with zero messages. Instead, a new message is generated by each vehicle at every 5 seconds containing its current position, speed, and generation time.
- ***Both***: both applications are included in the simulation. Each vehicle begins with 5 messages containing parking information and generates traffic information messages at every 5 seconds.

4.3.5.3 Urban scenario

For urban scenario, we select a map fragment from Manhattan, New York City, USA (as previously shown in Figure 3.28). This segment has an area of 1.5 x 2 km^2 and was retrieved with OpenStreetMaps [94]. The average density at a random time instant is 50 vehicles/km^2.

Figure 4.22 shows the histogram of the connectivity time between every pair of vehicle in our urban scenario. In this urban setting, the connectivity time can vary from a few seconds to tens of seconds, depending on whether vehicles have a similar route. Notably, more than 4% fall in connectivity times that are lower than 3 seconds, which could be explained by the presence of buildings serving as obstacles in our scenario.

Figure 4.23(a) shows the results when applying the Jain's fairness index. As expected, FairAD and Max-min present the highest fairness index values,

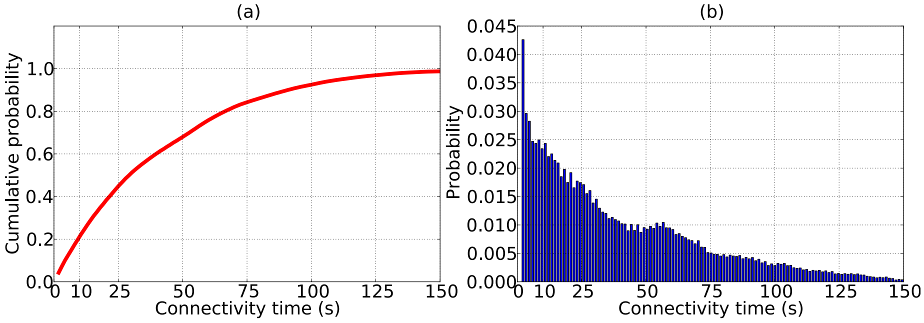


Figure 4.22: The connectivity time histograms for the urban scenario

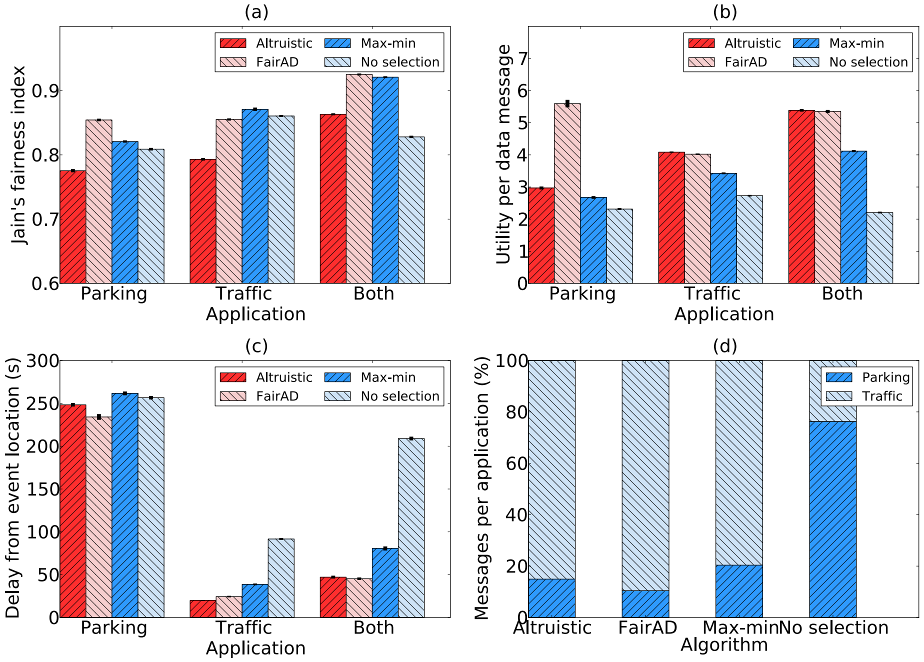


Figure 4.23: Results with 95% confidence intervals for the urban scenario

whereas Altruistic consistently presents lower values in all combinations. Although Max-min gives more priority to maximizing fairness, FairAD achieves higher fairness index in the parking application. This can be reasoned by the high gap in utility among vehicles depending on whether they will eventually park or not. For this reason, Max-min is not always able to compensate the low utility of all vehicles in the neighborhood. In contrast, FairAD manages to spread messages with higher utility more quickly and, in this particular scenario, is able to achieve a higher fairness index on average. The approach with no selection presents variable results, since it only considers the messages' IDs as criteria for selecting data to broadcast.

In terms of efficiency, Figure 4.23(b) presents the results for the utility per message received. In all cases, Altruistic and FairAD achieve higher efficiency compared with Max-min and No selection. Notably, FairAD outperforms Altruistic in the parking application. To explain this behavior, we have further analyzed the exchange of messages of both methods. The reason for such difference lies in the fact that Altruistic only prioritizes the total utility gain of all neighbors as a whole. Especially with such variability in the utility that each vehicle gains in this scenario, some vehicles simply do not receive any new message, which hinders the dissemination of certain messages that could be of higher utility for other vehicles encountered later in the city. Such behavior has been already previously observed in our results in Section 4.1.

The delay is generally lower for all methods that consider utility when exchanging messages, as shown in Figure 4.23(c). The delay values are higher with the parking application, since we assign random start age values in the beginning of the simulation taken from the interval $[0, 300]$.

Figure 4.23(d) shows the percentage of messages received by a vehicle for each application. We can observe that traffic related information is spread more quickly when employing data selection methods due to its higher relevance to most vehicles on the road. On the other hand, since parking information contain lower messages IDs in the simulation, more messages of this type are spread with the approach with no selection.

Finally, Figure 4.24 highlights the differences between each data selection method by showing the map of information received by a random vehicle when running both applications. Since messages with higher utility values are gathered with both Altruistic and FairAD, they present higher utility per message received compared with Max-min and No selection. Another point worth noting is that the information received with Altruistic and FairAD relates to coordinates that are closer to the vehicle's route, indicated with a solid line. In contrast, Max-min and No selection gather data related to farther lo-

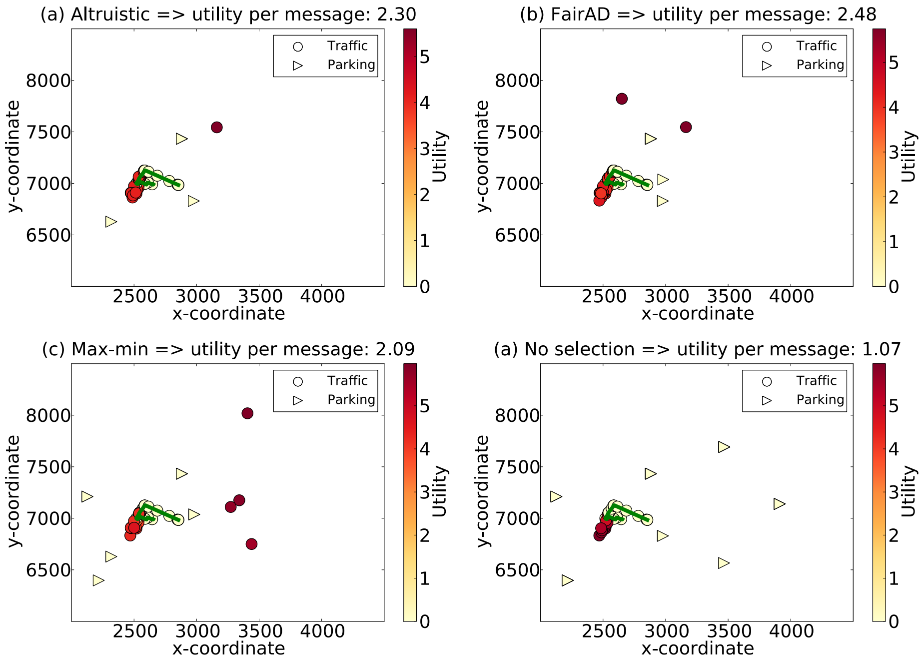


Figure 4.24: Geographical map of the information received by a random vehicle in the urban scenario

cations, thereby providing lower utility to the vehicle. Notably, as previously mentioned, the approach with no selection collects more parking information compared with other approaches.

In summary, the goal of each data selection method directly influences the behavior of the data exchange performed in the neighborhood. Overall, FairAD achieves both high fairness index and efficiency. Also, the delay is notably lower for methods employing data selection.

4.3.5.4 Two-directional highway scenario

The highway consists of a 1-kilometer straight road with two lanes in each road direction. We select a moderate density of 20 vehicles/km/lane that contains both vehicles moving at high speeds, i.e., 120 km/h, and low speed traffic due

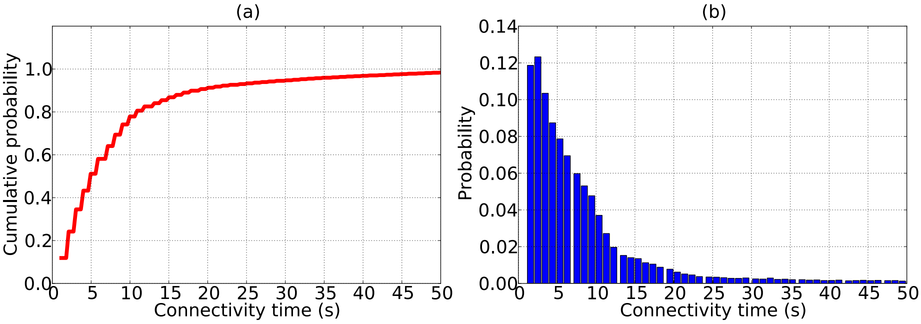


Figure 4.25: The connectivity time histograms for the highway scenario

to a small traffic jam in one of the road ends. For this scenario, in total 20 runs of 100 seconds are executed.

Figure 4.25 shows the histogram of the connectivity time between every pair of vehicle in this highway scenario. Compared with our urban scenario, the connectivity time between vehicles is generally lower due to quicker encounters in the highway, with 80% being concentrated up to only 10 seconds of connectivity.

Figure 4.26(a) shows the results when applying the Jain's fairness index. The results are similar to those presented in our urban scenario, where Max-min and FairAD achieve higher fairness index compared with Altruistic.

In terms of efficiency (Figure 4.26(b)), the higher utility per message received achieved by FairAD when compared with Altruistic is evident, in this case, for both applications. Especially because of the presence of quicker encounters between vehicles, only few vehicles are benefited from the data exchange in some occasions with Altruistic, which hinders the dissemination of other messages potentially important to other vehicles further ahead on the road.

Similarly to what we observe with the urban scenario, the delay is generally lower when employing data selection mechanisms, as shown in Figure 4.26(c). In particular, Max-min presents higher delay when running the parking application due to its inability to compensate differences in utility gain between vehicles in such quick encounters.

Figure 4.26(d) shows the percentage of messages received by a vehicle for each application. The limited connectivity time between vehicles accentuates

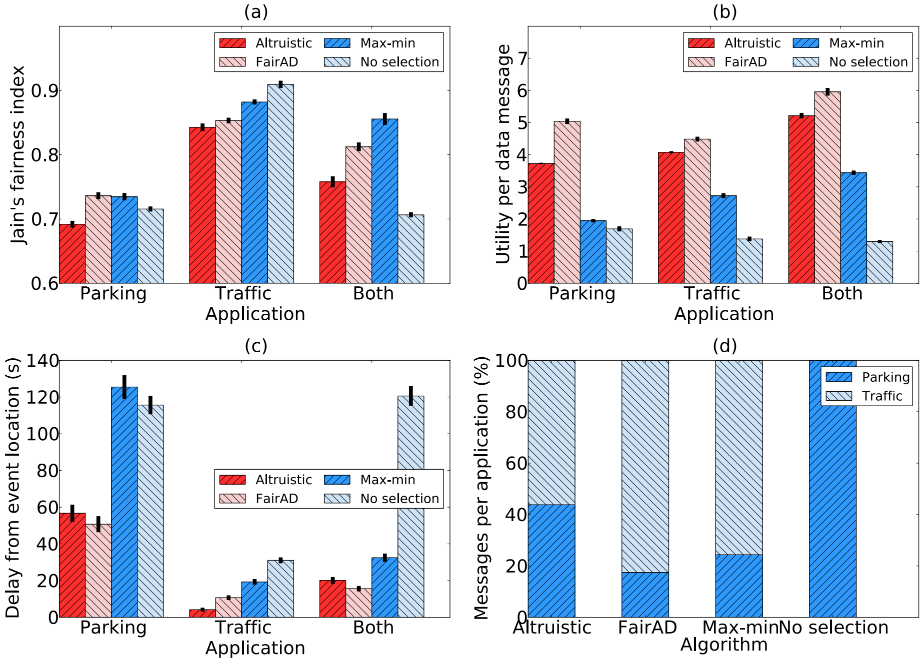


Figure 4.26: Results with 95% confidence intervals for the highway scenario

the priority given by the approach with no selection to disseminate parking information only.

Overall, the results follow a similar pattern to those presented for urban scenarios. However, because of the limited connectivity time present for data exchange, the differences between each method becomes more evident.

4.3.6 Conclusion

This section has presented FairAD, a dissemination protocol that utilizes the available bandwidth efficiently by maximizing the data utility gain of vehicles in the neighborhood and controlling the network load inserted into the network. It combines both a data selection algorithm to distribute application data utility fairly over vehicles and an adaptive transmission rate control to limit the number of messages broadcast.

Simulation results verify the benefits of employing data selection mechanisms in terms of efficiency and delay in delivering relevant data to interested vehicles. In comparison with other approaches, FairAD presents a higher fairness index and yet it maintains a high level of bandwidth utilization efficiency. In every scenario considered, the protocol shows to adaptively control the rate of transmissions as new information about the environment is collected.

We further verified the correctness of FairAD by means of real-world experiments. With a typical experiment set-up, we show that the connectivity time between vehicles moving at high speeds in opposite directions can be limited to a few seconds and considerably compromises the amount of data exchanged. Finally, we have shown the applicability of FairAD with two specific utility functions proposed for parking and traffic information applications. The results obtained with these functions reassure the benefits of FairAD in terms of fairness index and bandwidth utilization efficiency.

Table 4.7: Overview of solutions presented in the chapter

Solution	Goal	Requirements	Target scenario
FSO	Basic protocol to achieve fairness between every two vehicles using unicast communication	Contextual info	Highway & urban
FairDD	Achieve fairness and basic network load control in the local neighborhood using periodic broadcast communication	Contextual info	Highway & urban
FairAD	Achieve fairness while adaptively controlling the network load in the local neighborhood using broadcast communication	Contextual info	Highway & urban

4.4 Concluding remarks

In this chapter, we have proposed algorithms and protocols to tackle the problem of selecting data while controlling the network load in non-safety applications. Table 4.7 shows an overview of these solutions and their evolution in terms of goals achieved throughout the chapter.

In Section 4.1, we presented a study of the existing trade-offs between relying on fairness or efficiency as primary goals for selecting data when the connectivity time or available bandwidth is not large enough for all data to be broadcast. By means of a basic protocol for the communication between pairs of vehicles, we verified that fairness brings advantages in terms of delivering relevant data to a higher number of interested users such as a higher Jain's fairness index and comparable or higher sum of utility gains.

In view of the advantages of disseminating data based on a fairness criteria, in Section 4.2 we proposed the FairDD protocol. FairDD distributes data utility fairly among vehicles by relying on periodic broadcasts synchronized in the neighborhood to prioritize messages according to the Nash Bargaining criteria. FairDD is also able to suppress the least relevant data, given a defined maximum network load allowed. We showed that employing data selection leads to a higher efficiency in terms of utility gain per data message received and to a quicker dissemination of relevant data to interested vehicles. Overall, FairDD presents a higher fairness index compared with other approaches while being efficiency in terms of utility per message received.

Finally, Section 4.3 presents the FairAD protocol. FairAD incorporates FairDD's fairness algorithm into the ATB protocol, thereby distributing data utility fairly over vehicles while adaptively controlling the network load. The protocol dynamically adjusts the intervals between consecutive broadcasts based on both data priority and network load. FairAD was validated with both small-scale real-world experiments and simulations of realistic large-scale networks. We showed that FairAD controls the network load, presents a higher fairness index, and it maintains a high level of efficiency in terms of utility per message received compared to other approaches.

Based on the results presented in this chapter, we can conclude that FairAD represents a suitable solution for achieving data utility fairness while adaptively controlling the network load in the neighborhood. FairAD incorporates concepts and evolves from FSO and FairDD and, therefore, represents the most complete solution of this chapter. We showed that using Nash Bargaining as means to achieve fairness guarantees a proper balance between fairness and efficiency in terms of utility gains. We additionally showed that approaches relying on Nash Bargaining can in some cases achieve a higher efficiency compared with approaches that focus solely on efficiency, as shown in Sections 4.1.5 and 4.3.5.2. This was explained by fact that approaches focusing only on efficiency prioritize the total utility gain of all neighbors as a whole. When a high variability in terms of utility gains is present, some vehicles simply do not receive any new message, which hinders the dissemination of certain messages that could be of higher utility for other vehicles encountered later in the city.

Throughout this chapter, we have tested different combinations of contextual parameters to be used as utility functions. We consistently verified for every combination that focusing on fairness leads to a better distribution of data utility among neighboring vehicles. These contextual parameters were assumed to be shared among neighbors via *hello* messages transmitted alongside with data messages. *Hello* messages have in this context a similar function to the *beacons* exploited in the solutions for safety applications in Chapter 3. Therefore, as an alternative solution, the data concerning such contextual parameters could be included in *beacons* instead of separate *hello* messages.

CHAPTER 5

Conclusion

In vehicular networks, vehicles are expected to continuously gather, process, and disseminate relevant data throughout the road. In particular, the process of disseminating data to interested vehicles is paramount to support the development of not only safety applications, but also information-rich applications. This thesis has presented data dissemination solutions that fulfill the requirements of both safety and non-safety applications. In Chapter 2, we reviewed state-of-the-art solutions with respect to both types of applications and outlined aspects not yet addressed in the literature. In Chapters 3 and 4, we described our contributions with respect to data dissemination for safety and non-safety applications, respectively.

In this chapter, we first elaborate on the main contributions and results of this thesis in Section 5.1. Next, Section 5.2 revisits and answers our research questions. Finally, Section 5.3 provides directions for future work.

5.1 Contributions

The contributions with respect to data dissemination for **safety** applications can be summarized as follows:

(Contribution 1) A directional data dissemination protocol for highway scenarios: in Section 3.1, we presented a data dissemination protocol to deal with both dense and sparse vehicular networks, namely, SRD. We focused on coping with disconnected highway scenarios while preventing the broadcast storm problem in dense networks. We achieved this goal by proposing a straightforward store-carry-forward communication model for sparse networks and an optimized delay-based suppression technique for dense networks. Our simulation results showed that SRD outperforms state-of-the-art protocols in terms of delivery ratio and introduces a lower load into the network. In addition, SRD presents higher delivery ratio in highly dynamic scenarios, thereby showing higher robustness when vehicles move to different roads frequently.

(Contribution 2) A scalable directional data dissemination protocol for dense highway scenarios: in Section 3.2, we proposed a time slot based suppression technique, namely DOT, to further tackle the broadcast storm problem in dense networks. By exploiting the presence of one-hop neighborhood information contained in periodic safety *beacons*, DOT is capable of controlling with high precision the density of vehicles within each time slot. By means of simulations, we showed that DOT is scalable, achieves near optimum delay results, and is robust to errors caused by possible inaccurate transmission range estimations. Furthermore, DOT outperforms other delay-based schemes in diverse network densities.

(Contribution 3) A scalable data dissemination protocol for both highway and urban scenarios: in Section 3.3, we adapted and extended concepts used in the two previous contributions for the case of multi-directional dissemination, thereby tackling scalability issues in both highway and urban scenarios. We presented AMD: an infrastructure-less protocol that combines a generalized delay-based suppression technique based on directional sectors and a store-carry-forward algorithm to support multi-directional data dissemination. By means of simulation, we showed that AMD scales properly in various network densities in both highway and urban scenarios. Compared with protocols especially designed for either highway or urban scenarios, AMD presents higher delivery ratio, lower end-to-end delay, and lower number of transmissions.

The contributions with respect to data dissemination for **non-safety** applications can be summarized as follows:

(Contribution 4) A comparative study between fairness and efficiency as goals for data selection: in Section 4.1, we studied the trade-offs between fairness and efficiency to tackle the problem of selecting data when the connectivity time or available bandwidth is not large enough for all data to be broadcast. Data selection approaches aim to maximize the utility (importance) gain of all vehicles. We relied on a basic protocol to exchange messages between pair of vehicles. Overall, using a fair data selection strategy based on Nash Bargaining showed to increase the delivery ratio of interested vehicles and fairness index while maintaining a high sum of utility gains.

(Contribution 5) A fair data dissemination protocol via synchronous broadcasting: given the advantages of disseminating data based on a fairness criteria, in Section 4.2 we proposed FairDD: a data dissemination protocol that distributes data utility fairly among vehicles in the neighborhood. The protocol relies on periodic broadcasts synchronized among neighbors in order to prioritize messages according to the Nash Bargaining criteria. This mechanism is also able to suppress the least relevant data, given a defined maximum network load allowed. With simulation, we showed that by employing data selection the network is utilized more efficiently in terms of utility gain per data message received and relevant data is more quickly spread to interested vehicles. More importantly, FairDD presents a higher fairness index compared with other approaches and yet it maintains a high level of efficiency in terms of utility per message received.

(Contribution 6) A fair and adaptive data dissemination protocol: in Section 4.3, we described the FairAD protocol. FairAD incorporates FairDD's fairness algorithm into the ATB protocol, thereby distributing data utility fairly over vehicles while adaptively controlling the network load. The protocol dynamically adjusts the intervals between consecutive broadcasts based on both data priority and network load. We showed the applicability of the protocol with two examples of utility functions for two Traffic Information Systems (TIS) applications: parking-related and traffic information applications. FairAD was validated with both small-scale real-world experiments and simulations of realistic large-scale networks. We showed that FairAD controls the network load, presents a higher fairness index, and it maintains a high level of efficiency in terms of utility per message received compared to other approaches.

5.2 Research questions revisited

The main focus of this thesis was to study data dissemination solutions for vehicular environments that fulfill the requirements of both safety and non-safety applications. In particular, we concentrated our efforts on *scalable* data dissemination solutions for infrastructure-less vehicular networks. The main research question of this thesis was:

> *How to achieve scalable data dissemination in infrastructure-less vehicular environments while fulfilling specific requirements of both safety and non-safety applications?*

In view of the distinct requirements between safety and non-safety applications, we approached our main research question by answering two sub research questions. In the following, we answer each question by revisiting our hypotheses and contributions provided by this thesis.

> ***(RQ.1) Safety****: how to disseminate data in a timely manner to all vehicles in the affected region while minimizing the number of transmissions?*
>
> To answer research question ***(RQ.1)***, we started from the hypothesis that in sparse networks we can cope with intermittent connectivity by exploiting the mobility of vehicles to *store, carry, and forward* messages to further vehicles on the road. We validated this hypothesis by presenting a data dissemination protocol that relies on a straightforward store-carry-forward model capable of achieving high delivery ratio in disconnected networks **(Contribution 1)**.
>
> In addition, we argued that the presence of *beacons* could be *exploited* to achieve efficient selection of neighboring vehicles to forward messages, especially in dense networks. This hypothesis was validated by showing that the positioning information contained in *beacons* allows vehicles to be efficiently assigned to different time slots according to their priority in rebroadcasting messages. In this way, we can effectively achieve higher delivery ratio, lower end-to-end delay, and lower number of transmissions compared to approaches that do not exploit *beacons* **(Contribution 2)**.
>
> Finally, these two strategies were combined and extended to the case of multi-directional dissemination, thereby fulfilling the requirements of safety applications in both highway and urban scenarios **(Contribution 3)**.
>
> ***(RQ.2) Non-safety****: how to select and disseminate the most relevant data to interested vehicles while controlling the network load?*
>
> We addressed question ***(RQ.2)*** with the hypothesis that when considering vehicles with conflicting data interests, a data dissemination solution should rely on concepts of *fairness*. To validate this hypothesis, we first motivated in Chapter 2 that a fair distribution of utility is needed to maximize individual interest gains and, at the same time, prevent situations where only a subset of vehicles receive relevant information. Later, we presented a comparative study between fairness

and efficiency as goals for data selection. We showed that an approach which relies on the Nash Bargaining concept from game theory is able to increase the delivery ratio of interested vehicles and fairness index while maintaining a high sum of utility gains **(Contribution 4)**.

Our second hypothesis was that in order to cope with both sparse and dense networks, a mechanism to control the *network load* should *adaptively* adjust its parameters according to the current network conditions. This hypothesis is supported by the fact that data dissemination solutions should cope with sudden density variations in view of the dynamic nature of vehicular networks. We first presented a data dissemination protocol that distributes data utility fairly among vehicles via synchronous periodic dissemination **(Contribution 5)**. However, defining the periodicity of dissemination a priori is clearly not suitable when the number of vehicles in the neighborhood varies continually. Instead, the order of transmission for different messages should change adaptively according to the most up-to-date state of the neighborhood. To this end, we further elaborated on distributing data utility fairly over vehicles while adaptively controlling the network load in **(Contribution 6)**.

Overall, we fulfill the requirements of non-safety applications by maximizing data utility gains of interested vehicles while controlling the network load. From our results, we can conclude that using Nash Bargaining as criteria for data selection leads to a proper balance between utility fairness and efficiency.

In conclusion, this thesis has contributed to the literature by providing new insights into the process of disseminating data in vehicular networks. The solutions presented throughout the chapters have the potential to deliver data related to a wide range of events such as accidents, traffic jams, and points of interest, thereby increasing safety, efficiency and comfort to road users. While safety and non-safety applications have distinct requirements, they both share the crucial requirement of scalability due to the continual network density variations. For this reason, special attention to scalability has been paid, which allowed us to design flexible data dissemination mechanisms that are able to work in both sparse and dense networks. Finally, the proposed solutions require no modification in underlying protocol standards nor assume infrastructure support. Infrastructure-less solutions may prove to be particularly useful at an early stage of deployment, since they are by design robust against intermittent connectivity.

5.3 Future research directions

This thesis has addressed problems related to data dissemination in vehicular networks. However, there are still open issues to be addressed in future work.

- **Infrastructure**: this thesis has focused solely on the case of infrastructure-less networks, assuming that all vehicles are equipped with wireless devices. However, roadside units and cellular networks will certainly play a role in improving the network's connectivity, especially at an early stage of market penetration of the technology. Both technologies have the potential to improve the solutions described in this work. One straightforward example is to assign roadside units the task of storing and rebroadcasting data later on to other vehicles passing by the road, as proposed in [34]. This would bring clear benefits in terms of delay and delivery ratio. At the moment of writing, early results indicate that cellular network technologies, e.g., 4G Long-Term Evolution (LTE), cannot always meet the delay requirements of safety applications [104, 105]. However, non-safety applications can certainly benefit from a more pervasive and centralized knowledge, for example, by sending requests for certain data to be redirected to specific regions where the current overall utility gain is low. In our view, due to the large amount of data that vehicles can generate, distributed vehicular networks can alleviate the data traffic in cellular networks, thereby making a hybrid approach more attractive as suggested in [18].

- **Data processing**: our efforts were solely directed to the process of distributing data. However, further study is needed to evaluate the impact of our solutions when also considering the processing of data, e.g., data aggregation, performed by each vehicle in the network. This is particularly important to reduce the amount of data generated by non-safety applications.

- **Towards a single data dissemination module**: in this thesis, we considered separate data dissemination modules to individually address safety and non-safety applications. This separation was reasoned by the different requirements and presence of separate protocols and radio channels for each type of application, as determined for standardized underlying layers. However, such separate radio channels may lead to sub-optimal usage of the available bandwidth. On the one hand, broadcasting safety-related messages in highly dense networks can quickly exceed the control channel capacity and compromise safety requirements. On the other hand, sparse networks can lead to an underutilization of the channel. To this end, it would be interesting to evaluate the use of dynamic channel allocation where the amount of bandwidth could be adjusted according to the current context and application requirements [106, 107]. This would bring several opportunities with regard to creating a single data dissemination module capable of adaptively choosing the best combination of bandwidth size, priority of messages, and radio channel in order to meet requirements of various applications.

- **Fairness across applications**: for non-safety applications, we advocated a fair data utility dissemination among vehicles. However, another aspect to be investigated is to how the concept of fairness can be extended to the application level. Here, we would be interested in methods that prevent messages from a certain application to fall into starvation. Depending on the business model adopted, more time and utility gain could be reserved to particular applications.
- **Internet and multimedia applications**: our solution for non-safety applications focused on the questions of what and when to broadcast data in the neighborhood. However, more tailored solutions should be incorporated to address robustness issues with respect to multimedia streaming or Internet access, e.g., with network coding [108].
- **Security and privacy**: security and privacy are a crucial aspects that have not been addressed in this thesis. Compromising the data disseminated can bring serious consequences such as accidents if no measure is taken to prevent the insertion of malicious information. One challenge is to achieve a balance between privacy and security. Although context-adaptive dissemination, such as our solution for non-safety applications, has shown to limit the impact of attacks [109], further analysis is needed to evaluate the consequences of vehicles not willing to cooperate in the dissemination process.

Bibliography

[1] Wards Auto, “World Vehicle Population Tops 1 Billion Units,” 2011. [Online]. Available: http://wardsauto.com/ar/world_vehicle_population_110815

[2] World Health Organization, “Global Status Report on Road Safety,” 2013.

[3] Texas A&M Transportation Institute, “As traffic jams worsen, commuters allowing extra time for urgent trips.” [Online]. Available: http://mobility.tamu.edu/ums/media-information/press-release/

[4] Research and Innovative Technology Administration (RITA) - U.S. Department of Transportation (US DOT), “About ITS,” 2013. [Online]. Available: http://www.its.dot.gov/faqs.htm

[5] H. Hartenstein and K. P. Laberteaux, *VANET: Vehicular Applications and Inter-Networking Technologies*, H. Hartenstein and K. P. Laberteaux, Eds. Chichester, UK: John Wiley & Sons, Ltd, Jan. 2010.

[6] B. van Arem, C. J. G. van Driel, and R. Visser, “The Impact of Cooperative Adaptive Cruise Control on Traffic-Flow Characteristics,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 4, pp. 429–436, Dec. 2006.

[7] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil, “Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions,” *IEEE Communications Surveys & Tutorials*, vol. 13, no. 4, pp. 584–616, 2011.

[8] M. van Eenennaam, W. Klein Wolterink, G. Karagiannis, and G. Heijenk, “Exploring the solution space of beaconing in VANETs,” in *IEEE Vehicular Networking Conference (VNC)*. IEEE, Oct. 2009, pp. 1–8.

[9] R. S. Schwartz, H. W. van Dijk, and H. Scholten, “Towards opportunistic sensed data dissemination in vehicular environments,” in *IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. IEEE, Mar. 2011, pp. 336–339.

[10] U. Lee and M. Gerla, “A survey of urban vehicular sensing platforms,” *Computer Networks*, vol. 54, no. 4, pp. 527–544, Mar. 2010.

[11] R. A. Uzcátegui and G. Acosta-Marum, “Wave: A tutorial,” *IEEE Communications Magazine*, vol. 47, no. 5, pp. 126–133, May 2009.

[12] ETSI Standard, "Final draft ETSI ES 202 663 V1.1.0 - Intelligent Transport Systems (ITS); European profile standard for the physical and medium access control layer of Intelligent Transport Systems operating in the 5 GHz frequency band," Tech. Rep., 2009.

[13] IEEE Computer Society, "IEEE 802.11p standard - Amendment 6: Wireless Access in Vehicular Environments," Tech. Rep., 2010.

[14] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments," in *IEEE Vehicular Technology Conference (VTC Spring)*. IEEE, May 2008, pp. 2036–2040.

[15] M. Torrent-Moreno, P. Santi, and H. Hartenstein, "Fair sharing of bandwidth in VANETs," in *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks (VANET)*. Cologne, Germany: ACM, Sep. 2005, pp. 49–58.

[16] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein, "Vehicle-to-Vehicle Communication: Fair Transmit Power Control for Safety-Critical Information," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 7, pp. 3684–3703, Sep. 2009.

[17] C. Sommer, O. K. Tonguz, and F. Dressler, "Traffic information systems: efficient message dissemination via adaptive beaconing," *IEEE Communications Magazine*, vol. 49, no. 5, pp. 173–179, May 2011.

[18] F. Bai and B. Krishnamachari, "Exploiting the wisdom of the crowd: localized, distributed information-centric VANETs [Topics in Automotive Networking," *IEEE Communications Magazine*, vol. 48, no. 5, pp. 138–146, May 2010.

[19] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "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)*, ACM New York, NY, USA. Seattle, WA, USA: ACM, 1999, pp. 151–162.

[20] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks (SASN)*. Alexandria, VA, USA: ACM, Nov. 2005, pp. 11–21.

[21] R. S. Schwartz, R. R. R. Barbosa, N. Meratnia, G. Heijenk, and H. Scholten, "A Simple and Robust Dissemination protocol for VANETs," in *European Wireless Conference*. Lucca, Italy: IEEE, Apr. 2010, pp. 214–222.

[22] R. S. Schwartz, R. R. R. Barbosa, N. Meratnia, G. Heijenk, and H. Scholten, "A directional data dissemination protocol for vehicular environments," *Computer Communications*, vol. 34, no. 17, pp. 2057–2071, Nov. 2011.

[23] R. S. Schwartz, K. Das, H. Scholten, and P. Havinga, "Exploiting beacons for scalable broadcast data dissemination in VANETs," in *Proceedings of the ninth ACM international workshop on Vehicular inter-networking, systems, and applications (VANET)*. Low Wood Bay, Lake District, United Kingdom: ACM, Jun. 2012, p. 53.

[24] R. S. Schwartz, A. E. Ohazulike, H. W. van Dijk, and H. Scholten, "Analysis of Utility-Based Data Dissemination Approaches in VANETs," in *4th International Symposium on Wireless Vehicular Communications (WIVEC) - VTC Fall*. San Francisco, CA, USA: IEEE, Sep. 2011, pp. 1–5.

[25] R. S. Schwartz, A. E. Ohazulike, and H. Scholten, "Achieving Data Utility Fairness in Periodic Dissemination for VANETs," in *IEEE 75th Vehicular Technology Conference (VTC Spring)*. Yokohama, Japan: IEEE, May 2012, pp. 1–5.

[26] R. S. Schwartz, A. E. Ohazulike, C. Sommer, H. Scholten, F. Dressler, and P. Havinga, "Fair and adaptive data dissemination for Traffic Information Systems," in *IEEE Vehicular Networking Conference (VNC)*. Seoul, South Korea: IEEE, Nov. 2012, pp. 1–8.

[27] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing (MobiHoc)*. Lausanne, Switzerland: ACM, 2002, p. 194.

[28] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc networks," in *Proceedings of the 3rd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems (MSWIM)*. Boston, MA, USA: ACM, 2000, pp. 61–68.

[29] W. Peng and X. Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks," in *First Annual Workshop on Mobile and Ad Hoc Networking and Computing (MobiHOC)*. Boston, MA, USA: IEEE, 2000, pp. 129–130.

[30] W. Peng, "Efficient broadcast in mobile ad hoc networks using connected dominating sets," *Journal of software*, 1999.

[31] S. Panichpapiboon and W. Pattara-atikom, "A Review of Information Dissemination Protocols for Vehicular Ad Hoc Networks," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 3, pp. 784 – 798, 2011.

[32] N. Wisitpongphan, O. K. Tonguz, J. S. Parikh, P. Mudalige, F. Bai, and V. Sadekar, "Broadcast storm mitigation techniques in vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 84–94, Dec. 2007.

[33] H. Füß ler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad hoc networks," *Ad Hoc Networks*, vol. 1, no. 4, pp. 351–369, Nov. 2003.

[34] G. Korkmaz, E. Ekici, F. Özgüner, and U. Özgüner, "Urban multi-hop broadcast protocol for inter-vehicle communication systems," in *Proceedings of the first ACM workshop on Vehicular ad hoc networks (VANET)*. Philadelphia, PA, USA: ACM, Oct. 2004, pp. 76–85.

[35] N. Cenerario, T. Delot, and S. Ilarri, "A Content-Based Dissemination Protocol for VANETs: Exploiting the Encounter Probability," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 771–782, Sep. 2011.

[36] J. J. Blum and A. Eskandarian, "Avoiding Timeslot Boundary Synchronization for Multihop Message Broadcast in Vehicular Networks," *IEEE 69th Vehicular Technology Conference (VTC Spring)*, pp. 1–5, Apr. 2009.

[37] M. van Eenennaam, G. Heijenk, and G. Karagiannis, "Over-the-Horizon Awareness for Advanced Driver Assistance Systems: the TrafficFilter and microSlotted 1-Persistence Flooding," Tech. Rep., 2011.

[38] Y.-t. Tseng, R.-h. Jan, C. Chen, C.-f. Wang, and H.-H. Li, "A vehicle-density-based forwarding scheme for emergency message broadcasts in VANETs," in *The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*. San Francisco, CA, USA: IEEE, Nov. 2010, pp. 703–708.

[39] L.-C. Tung and M. Gerla, "An efficient road-based directional broadcast protocol for urban VANETs," in *IEEE Vehicular Networking Conference (VNC)*. Jersey City, NJ, USA: IEEE, Dec. 2010, pp. 9–16.

[40] K. Suriyapaibonwattana and C. Pomavalai, "An Effective Safety Alert Broadcast Algorithm for VANET," in *International Symposium on Communications and Information Technologies*. Laos: IEEE, Oct. 2008, pp. 247–250.

[41] T. Tielert, D. Jiang, Q. Chen, L. Delgrossi, and H. Hartenstein, "Design methodology and evaluation of rate adaptation based congestion control for Vehicle Safety Communications," in *IEEE Vehicular Networking Conference (VNC)*. Amsterdam, The Netherlands: IEEE, Nov. 2011, pp. 116–123.

[42] R. Meireles, P. Steenkiste, and J. Barros, "DAZL: Density-Aware Zone-based packet forwarding in vehicular networks," in *IEEE Vehicular Networking Conference (VNC)*. IEEE, Nov. 2012, pp. 234–241.

[43] M. Koubek, S. Rea, and D. Pesch, "Reliable Delay Constrained Multihop Broadcasting in VANETs," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, no. 753256, pp. 1–13, 2010.

[44] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Technical Report CS-200006, Duke University, Tech. Rep., 2000.

[45] T. Spyropoulos and C. S. Psounis K. asd Raghavendra, "Efficient routing in intermittently connected mobile networks: The multiple-copy case," *IEEE/ACM Transactions on Networking (TON)*, vol. 16, no. 1, pp. 77–90, 2008.

[46] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: social-based forwarding in delay tolerant networks," in *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*. ACM, 2008, pp. 241–250.

[47] O. Tonguz, N. Wisitpongphan, and F. Bai, "DV-CAST: A distributed vehicular broadcast protocol for vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 17, no. 2, pp. 47–57, Apr. 2010.

[48] F. J. Ros, P. M. Ruiz, and I. Stojmenovic, "Reliable and Efficient Broadcasting in Vehicular Ad Hoc Networks," in *IEEE 69th Vehicular Technology Conference (VTC Spring)*. Barcelona, Spain: IEEE, Apr. 2009, pp. 1–5.

[49] ——, "Acknowledgment-Based Broadcast Protocol for Reliable and Efficient Data Dissemination in Vehicular Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 1, pp. 33–46, Jan. 2012.

[50] M. Fogue, P. Garrido, F. J. Martinez, J.-C. Cano, C. T. Calafate, and P. Manzoni, "Evaluating the impact of a novel message dissemination scheme for vehicular networks using real maps," *Transportation Research Part C: Emerging Technologies*, vol. 25, pp. 61–80, Dec. 2012.

[51] W. Viriyasitavat, F. Bai, and O. K. Tonguz, "UV-CAST: An urban vehicular broadcast protocol," in *IEEE Vehicular Networking Conference (VNC)*. Jersey City, NJ, USA: IEEE, Dec. 2010, pp. 25–32.

[52] C. Chen, L. Liu, X. Du, X. Wei, and C. Pei, "Available connectivity analysis under free flow state in VANETs," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, p. 270, 2012.

[53] W. Viriyasitavat, F. Bai, and O. K. Tonguz, "Dynamics of Network Connectivity in Urban Vehicular Networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 3, pp. 515–533, Mar. 2011.

[54] M. van Eenennaam, A. Remke, and G. Heijenk, "An analytical model for beaconing in VANETs," in *IEEE Vehicular Networking Conference (VNC)*. Seoul, South Korea: IEEE, Nov. 2012, pp. 9–16.

[55] Z. Wang and M. Hassan, "How much of DSRC is available for non-safety use?" in *Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking (VANET)*. San Francisco, CA, USA: ACM, Sep. 2008, pp. 23–29.

[56] M. Amadeo, C. Campolo, and A. Molinaro, "Enhancing IEEE 802.11p/WAVE to provide infotainment applications in VANETs," *Ad Hoc Networks*, vol. 10, no. 2, pp. 253–269, Mar. 2012.

[57] N. N. Qadri, M. Fleury, B. R. Rofoee, M. Altaf, and M. Ghanbari, "Robust P2P Multimedia Exchange within a VANET," *Wireless Personal Communications*, vol. 63, no. 3, pp. 561–577, Sep. 2010.

[58] Z. Yang, M. Li, and W. Lou, "CodePlay: Live multimedia streaming in VANETs using symbol-level network coding," in *The 18th IEEE International Conference on Network Protocols*. Kyoto, Japan: IEEE, Oct. 2010, pp. 223–232.

[59] J.-S. Park, U. Lee, S. Y. Oh, M. Gerla, and D. S. Lun, "Emergency related video streaming in VANET using network coding," in *Proceedings of the 3rd international workshop on Vehicular ad hoc networks (VANET)*. Los Angeles, CA, USA: ACM, Sep. 2006, pp. 102–103.

[60] T. Kosch, C. Adler, S. Eichler, C. Schroth, and M. Strassberger, "The scalability problem of vehicular ad hoc networks and how to solve it," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 22–28, Oct. 2006.

[61] I. Leontiadis, P. Costa, and C. Mascolo, "A hybrid approach for content-based publish/subscribe in vehicular networks," *Pervasive and Mobile Computing*, vol. 5, no. 6, pp. 697–713, Dec. 2009.

[62] M. Röckl and P. Robertson, "Data Dissemination in Cooperative ITS from an Information-Centric Perspective," in *IEEE International Conference on Communications*. Cape Town, South Africa: IEEE, May 2010, pp. 1–6.

[63] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, Salt Lake City, UT, USA, Jun. 2011, pp. 350–358.

[64] T. Wang, L. Song, and Z. Han, "Coalitional Graph Games for Popular Content Distribution in Cognitive Radio VANETs," *IEEE Transactions on Vehicular Technology*, vol. PP, no. 99, pp. 1–24, 2013.

[65] K. C.-j. Lin and C.-w. Chen, "Preference-aware content dissemination in opportunistic mobile social networks," in *IEEE International Conference on Computer Communications (INFOCOM)*. Orlando, FL, USA: IEEE, Mar. 2012, pp. 1960–1968.

[66] B. Yu, F. Bai, and B. Krishnamachari, "RISA: Distributed Road Information Sharing Architecture," in *IEEE International Conference on Computer Communications (INFOCOM)*. Orlando, FL, USA: IEEE, Mar. 2012, pp. 1494–1502.

[67] J. Liu, D. Greene, M. Mosko, and J. Reich, "Using utility and microutility for information dissemination in Vehicle Ad Hoc Networks," in *IEEE Intelligent Vehicles Symposium (IV)*, Eindhoven, The Netherlands, Jun. 2008, pp. 755–762.

[68] J. Pu and M. Hamdi, "Utility-based fair bandwidth sharing in vehicular networks (extended)," *Wireless Communications and Mobile Computing*, vol. 10, no. 12, pp. 1648–1655, 2010.

[69] B. Shrestha, D. Niyato, Z. Han, and E. Hossain, "Wireless Access in Vehicular Environments Using BitTorrent and Bargaining," in *IEEE Global Telecommunications Conference (GLOBECOM)*. New Orleans, LO, USA: IEEE, Dec. 2008, pp. 1–5.

[70] S. Bastani, B. Landfeldt, C. Rohner, and P. Gunningberg, "A social node model for realising information dissemination strategies in delay tolerant networks," in *Proceedings of the 15th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems (MSWiM)*. Paphos, Cyprus: ACM, Oct. 2012, pp. 79–88.

[71] J. Mittag, F. Schmidt-Eisenlohr, M. Killat, J. Härri, and H. Hartenstein, "Analysis and design of effective and low-overhead transmission power control for VANETs," in *Proceedings of the fifth ACM international workshop on VehiculAr InterNETworking (VANET)*. San Francisco, CA, USA: ACM, Sep. 2008, pp. 39–48.

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

[73] L. Wischhof and H. Rohling, "Congestion control in vehicular ad hoc networks," in *IEEE International Conference on Vehicular Electronics and Safety*. Xi'an, China: IEEE, Oct. 2005, pp. 58–63.

[74] M. S. Bouassida and M. Shawky, "A Cooperative Congestion Control Approach within VANETs: Formal Verification and Performance Evaluation," *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, pp. 1–13, 2010.

[75] L. Wischhof, A. Ebner, and H. Rohling, "Information Dissemination in Self-Organizing Intervehicle Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 90–101, Mar. 2005.

[76] J. Rybicki, B. Scheuermann, M. Koegel, and M. Mauve, "PeerTIS: a peer-to-peer traffic information system," in *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking (VANET)*. Beijing, China: ACM, Sep. 2009, pp. 23–32.

[77] C. Sommer, O. K. Tonguz, and F. Dressler, "Adaptive beaconing for delay-sensitive and congestion-aware traffic information systems," in *IEEE Vehicular Networking Conference (VNC)*. Jersey City, NJ, USA: IEEE, Dec. 2010, pp. 1–8.

[78] S. Dietzel, F. Kargl, G. Heijenk, and F. Schaub, "On the potential of generic modeling for VANET data aggregation protocols," in *IEEE Vehicular Networking Conference (VNC)*. Jersey City, NJ, USA: IEEE, Dec. 2010, pp. 78–85.

[79] R. S. Schwartz, M. van Eenennaam, G. Karagiannis, G. Heijenk, W. Klein Wolterink, and H. Scholten, "Using V2V communication to create Over-the-horizon Awareness in multiple-lane highway scenarios," in *IEEE Intelligent Vehicles Symposium (IV)*. San Diego, CA, USA: IEEE, Jun. 2010, pp. 998–1005.

[80] A. Varga, "The OMNeT++ discrete event simulation system," in *Proceedings of the European Simulation Multiconference (ESM)*, Prague, Czech Republic, Jun. 2001, pp. 319–324.

[81] A. Köpke, M. Swigulski, K. Wessel, D. Willkomm, P. T. K. Haneveld, T. E. V. Parker, O. W. Visser, H. S. Lichte, and S. Valentin, "Simulating Wireless and Mobile Networks in OMNeT++ The MiXiM Vision," in *Proceedings of the First International ICST Conference on Simulation Tools and Techniques for Communications, Networks and Systems*. Marseille, France: ICST, Mar. 2008, pp. 1–8.

[82] L. Quadstone, "Quadstone Paramics v5.0 Modeller User Guide," *Scotland, UK*, 2004.

[83] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO - Simulation of Urban MObility - an Overview," in *The Third International Conference on Advances in System Simulation (SIMUL)*. Barcelona, Spain: IARIA, Oct. 2011, pp. 55–60.

[84] Vehicular Technology Society, “IEEE Standard for Wireless Access in Vehicular Environments (WAVE) – Networking Services,” in *IEEE 1609.3*, 2010.

[85] ——, “IEEE Standard for Wireless Access in Vehicular Environments – Security Services for Applications and Management Messages,” in *IEEE 1609.2*, 2010.

[86] Department of Defense USA, “Global Positioning System Standard,” Tech. Rep. September, 2008.

[87] A. Goldsmith, *Wireless communications*. Cambridge University Press, 2005.

[88] M. Kaynia, F. Fabbri, R. Verdone, and G. E. Ø ien, “Analytical study of the outage probability of ALOHA and CSMA in bounded ad hoc networks,” in *European Wireless Conference*, Lucca, Italy, Apr. 2010, pp. 544–550.

[89] T. Rappaport, *Wireless communications: principles and practice*. Prentice Hall PTR New Jersey, 2001, vol. 2.

[90] C. Sommer, R. German, and F. Dressler, “Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis,” *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, Jan. 2011.

[91] K. Sjöberg, J. Karedal, M. Moe, Ø. Kristiansen, R. Sø råsen, E. Uhlemann, F. Tufvesson, K. Evensen, and E. G. Ström, “Measuring and using the RSSI of IEEE 802.11p,” in *17th World Congress on Intelligent Transport Systems*, vol. 2010, no. 1, Busan, South Korea, 2010, pp. 1–9.

[92] C. Sommer, D. Eckhoff, R. German, and F. Dressler, “A computationally inexpensive empirical model of IEEE 802.11p radio shadowing in urban environments,” in *Eighth International Conference on Wireless On-Demand Network Systems and Services*. Bardonecchia, Italy: IEEE, Jan. 2011, pp. 84–90.

[93] P. Fuxjäger, A. Costantini, D. Valerio, P. Castiglione, G. Zacheo, T. Zemen, and F. Ricciato, “IEEE 802.11p transmission using GNURadio,” in *6th Karlsruhe WSR*, Karlsruhe, Germany, 2010, pp. 1–4.

[94] M. M. Haklay and P. Weber, “OpenStreetMap: User-Generated Street Maps,” *IEEE Pervasive Computing*, vol. 7, no. 4, pp. 12–18, Oct. 2008.

[95] ETSI Standard, “ETSI TR 102 638 V1.1.1 (2009-06) Technical Report Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions,” vol. 1, pp. 1–81, 2009.

[96] J. Nash Jr, “The bargaining problem,” *Econometrica: Journal of the Econometric Society*, vol. 18, no. 2, pp. 155–162, 1950.

[97] E. Kalai and M. Smorodinsky, “Other solutions to Nash’s bargaining problem,” *Econometrica: Journal of the Econometric Society*, vol. 43, no. 3, pp. 513–518, 1975.

[98] E. Kalai, “Proportional solutions to bargaining situations: Intertemporal utility comparisons,” *Econometrica: Journal of the Econometric Society*, vol. 45, no. 7, pp. 1623–1630, 1977.

[99] D. Bertsekas and R. Gallager, *Data Networks*. Prentice Hall, 1992.

[100] M. Bussieck and A. Meeraus, “General algebraic modeling system (GAMS),” *Applied Optimization*, vol. 88, pp. 137–158, 2004.

[101] W. Jain, R. and Chiu, D.M. and Hawe, *A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Systems*. Digital Equipment Corp., 1984.

[102] C. Varschen and P. Wagner, “Mikroskopische Modellierung der Personenverkehrsnachfrage auf Basis von Zeitverwendungstagebüchern,” *Integrierte Mikro-Simulation von Raum-und Verkehrsentwicklung. Theorie, Konzepte, Modelle, Praxis*, vol. 81, pp. 63–69, 2006.

[103] Vehicular Technology Society, “IEEE Standard for Wireless Access in Vehicular Environments (WAVE) – Multi-channel Operation IEEE Vehicular Technology Society,” in *IEEE 1609.4*, 2010.

[104] T. Mangel, T. Kosch, and H. Hartenstein, “A comparison of UMTS and LTE for vehicular safety communication at intersections,” in *IEEE Vehicular Networking Conference (VNC)*. Jersey City, NJ, USA: IEEE, Dec. 2010, pp. 293–300.

[105] A. Vinel, “3GPP LTE Versus IEEE 802.11p/WAVE: Which Technology is Able to Support Cooperative Vehicular Safety Applications?” *IEEE Wireless Communications Letters*, vol. 1, no. 2, pp. 125–128, Apr. 2012.

[106] M. Di Felice, L. Bedogni, and L. Bononi, “DySCO: a dynamic spectrum and contention controlframework for enhanced broadcast communication invehicular networks,” in *Proceedings of the 10th ACM international symposium on Mobility management and wireless access (MobiWac)*. Paphos, Cyprus: ACM, Oct. 2012, pp. 97–106.

[107] S. Ozturk, J. Mišić, and V. B. Mišić, “Capacity limits in a variable duty cycle IEEE 802.11p-based VANET,” *Wireless Communications and Mobile Computing*, vol. 12, no. 18, pp. 1672–1684, Dec. 2012.

[108] U. Lee, J.-s. Park, J. Yeh, G. Pau, and M. Gerla, “Code torrent: content distribution using network coding in VANET,” in *Proceedings of the 1st international workshop on Decentralized resource sharing in mobile computing and networking (MobiShare)*. Los Angeles, CA, USA: ACM, Sep. 2006, pp. 1–5.

[109] E. Schoch, M. Keppler, F. Kargl, and M. Weber, “On the security of context adaptive information dissemination,” *Security and Communication Networks*, vol. 1, no. 3, pp. 205–218, May 2008.

About the author

Ramon de Souza Schwartz obtained his B.Sc. degree in Computer Science from the Federal University of Espírito Santo (UFES), Brazil, in 2007. Later, he obtained his M.Sc. degree in Telematics from the University of Twente (UT), The Netherlands, in 2009. In the same year, he joined the Pervasive Systems group at the University of Twente to pursue his Ph.D. degree in the field of data dissemination in vehicular environments. During this time, he worked on the iLAND and SI4MS research projects. In June of 2013, he joined the Hitachi Europe SAS, Information and Communication Laboratory, Sophia Antipolis, France. He currently works as a R&D engineer in the context of vehicular networking and applications for Intelligent Transportation Systems.

List of publications in which he participated in reverse chronological order:

1) R.S. Schwartz, A.E. Ohazulike, C. Sommer, H. Scholten, F. Dressler, and P. Havinga. *On the applicability of fair and adaptive data dissemination in traffic information systems*. In: Elsevier Ad Hoc Networks, accepted for publication, submitted in April 2013.

2) R.S. Schwartz, H. Scholten, and P. Havinga. *A Scalable Data Dissemination Protocol for Both Highway and Urban Vehicular Environments*. In: Springer EURASIP Journal on Wireless Communications and Networking, accepted for publication, submitted in February 2013.

3) R.S. Schwartz, A.E. Ohazulike, C. Sommer, H. Scholten, F. Dressler, and P. Havinga. *Fair and adaptive data dissemination for traffic information systems*. In: 4th IEEE Vehicular Networking Conference (VNC), 14-16 Nov 2012, Seoul, South Korea. pp. 1-8.

4) A. Tüysüz Erman, R.S. Schwartz, A. Dilo, H. Scholten, and P. Havinga, *Infrastructure Assisted Data Dissemination for Vehicular Sensor Networks in Metropolitan Areas*, in Roadside Networks for Vehicular Communications: Architectures, Applications, and Test Fields, IGI Global, 2012, pp. 264–287.

5) R.S. Schwartz, K. Das, H. Scholten, and P. Havinga. *Exploiting beacons for scalable broadcast data dissemination in VANETs*. In: Proceedings of the 9th ACM international workshop on Vehicular inter-networking, systems, and applications (VANET), 25 June 2012, Low Wood Bay, Lake District, United Kingdom. pp. 53-62.

6) R.S. Schwartz, A.E. Ohazulike, and H. Scholten. *Achieving Data Utility Fairness in Periodic Dissemination for VANETs*. In: IEEE 75th Vehicular Technology Conference (VTC Spring), 6-9 May 2012, Yokohama, Japan. pp. 1-5.

7) R.S. Schwartz, R.R.R. Barbosa, N. Meratnia, G. Heijenk, and H. Scholten. *A directional data dissemination protocol for vehicular environments*. Elsevier Computer communications, 34 (17), 2011. pp. 2057-2071.

8) R.S. Schwartz, A.E. Ohazulike, H.W. van Dijk, and H. Scholten. *Analysis of Utility-Based Data Dissemination Approaches in VANETs*. In: 4th International Symposium on Wireless Vehicular Communications (WIVEC) - VTC Fall, 5-6 September 2011, San Francisco, CA, USA. pp. 1-5.

9) R.S. Schwartz, H.W. van Dijk, and H. Scholten, *Towards opportunistic sensed data dissemination in vehicular environments*, in IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), 21-25 March 2011, Seattle, WA, USA. pp. 336–339.

10) R.S. Schwartz, M. van Eenennaam, G. Karagiannis, G. Heijenk, W. Klein Wolterink, and H. Scholten, *Using V2V communication to create Over-the-horizon Awareness in multiple-lane highway scenarios*, in IEEE Intelligent Vehicles Symposium (IV), 21-24 June 2010, San Diego, CA, USA. pp. 998–1005.

11) R.S. Schwartz, R.R.R. Barbosa, N. Meratnia, G. Heijenk, and H. Scholten. *A Simple and Robust Dissemination Protocol for VANETs*. In: 16th European Wireless Conference, 12-15 April 2010, Lucca, Italy. pp. 214-222.

12) C. Marcondes, M.Y. Sanadidi, M. Gerla, R.S. Schwartz, R.O. Santos, and M. Martinello, *PathCrawler: Automatic harvesting web infra-structure*, in IEEE/IFIP Network Operations and Management Symposium (NOMS), 7-11 April 2008, Salvador, Brazil. pp. 339–346.

13) C. Marcondes, M.Y. Sanadidi, M. Gerla, M. Martinello, and R.S. Schwartz, *Exploring Embedded Path Capacity Estimation in TCP Receiver*, in Workshop on End-to-End Monitoring Techniques and Services, 21 May 2007, Munich, Germany. pp. 1-8.