Studies in Systems, Decision and Control 52

Muhammad Alam
Joaquim Ferreira
José Fonseca *Editors*

Intelligent Transportation Systems

Dependable Vehicular Communications for Improved Road Safety

Springer

Studies in Systems, Decision and Control

Volume 52

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Studies in Systems, Decision and Control" (SSDC) covers both new developments and advances, as well as the state of the art, in the various areas of broadly perceived systems, decision making and control- quickly, up to date and with a high quality. The intent is to cover the theory, applications, and perspectives on the state of the art and future developments relevant to systems, decision making, control, complex processes and related areas, as embedded in the fields of engineering, computer science, physics, economics, social and life sciences, as well as the paradigms and methodologies behind them. The series contains monographs, textbooks, lecture notes and edited volumes in systems, decision making and control spanning the areas of Cyber-Physical Systems, Autonomous Systems, Sensor Networks, Control Systems, Energy Systems, Automotive Systems, Biological Systems, Vehicular Networking and Connected Vehicles, Aerospace Systems, Automation, Manufacturing, Smart Grids, Nonlinear Systems, Power Systems, Robotics, Social Systems, Economic Systems and other. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution and exposure which enable both a wide and rapid dissemination of research output.

More information about this series at http://www.springer.com/series/13304

Muhammad Alam · Joaquim Ferreira
José Fonseca
Editors

Intelligent Transportation Systems

Dependable Vehicular Communications for Improved Road Safety

Springer

Editors

Muhammad Alam
Instituto de Telecomunicações
Campus Universitário de Santiago
Aveiro
Portugal

José Fonseca
Instituto de Telecomunicações
Campus Universitário de Santiago
Aveiro
Portugal

Joaquim Ferreira
Instituto de Telecomunicações
Campus Universitário de Santiago
Aveiro
Portugal

ISSN 2198-4182 ISSN 2198-4190 (electronic)
Studies in Systems, Decision and Control
ISBN 978-3-319-28181-0 ISBN 978-3-319-28183-4 (eBook)
DOI 10.1007/978-3-319-28183-4

Library of Congress Control Number: 2015958851

Printed on acid-free paper

This Springer imprint is published by SpringerNature
The registered company is Springer International Publishing AG Switzerland

To my family, my wife Amna and my doll Aiza for their support and love

Muhammad Alam

To Natália, Inês, Nuno and Miguel for their love and support

Joaquim Ferreira

To my wife Carla, my kids Ana, Miguel, to Filipe and Sara, to my parents, for being my light and my safe harbour

José Fonseca

Preface

Transportation systems are evolving towards Intelligent Transportation Systems (ITS) and the dependence on road transport in our daily lives has grown massively in recent years, in line with the problems arising from its use: permanent congestion on highways and urban centres, energy waste, CO_2 emissions with consequent impact on public health and high rates of accidents on the road networks. Recent research shows that the incorporation of information and communication technologies within vehicles and transportation infrastructure will revolutionize the way we travel today. The enabling technologies are intended to realize the frameworks that will spur an array of applications and use cases in the domain of road safety, traffic efficiency and driver's assistance. These applications will allow dissemination and gathering of useful information among vehicles and between transportation infrastructure and vehicles in pursuance of assisting drivers to travel safely and comfortably. However, dependable, reliable and real-time communication between vehicles and transport infrastructure are still critical challenges and need to be tackled for the success of these applications.

Understanding the importance of dependable and real-time communication in ITS domain, this book presents contents and significant results that provide the essential methodologies and algorithms for designing and implementing deterministic mechanisms for vehicular networks. The contents of the book are very consistent starting from the overview of basic concepts to the more technical aspects of dependable and real-time communication for vehicular networks along with the simulations, test beds and applications presentation. One of the distinctive aspects of this book is the presentation of work considering the real-time and dependable communication for vehicular networks. This book can contribute to enhance the knowledge of readers especially researchers, engineers and students working in this field. The gradual and interlinked organization of the chapters will enable readers to rapidly grasp the concepts related to dependable and real-time vehicular communication from physical and Medium Access Control layer to the application layers.

A concise overview of each chapter can be presented as follows. Chapter 1 presents the introduction, motivation and application of ITS. Basic architecture along with the communicating entities and functional elements constituting the European ITS Communications is presented. Furthermore, the two main protocol architectures for vehicular communication systems, one developed by the Institute of Electrical and Electronics Engineers (IEEE) and the other by the European Telecommunications Standard Institute (ETSI), are illustrated and compared. The chapter is concluded by providing an insight on the dependable and real-time communication in the scope of vehicular communications. IEEE has paid special consideration to the development of visible light communication (VLC) by introducing the IEEE 802.15.7 standard, which defines the PHY and MAC layer services for visible light personal area networks (VPANs). Although the implementation and use of VLC is still in early stages, there are research teams working in this area to find out solutions to achieve high data rates and reliable links using visible light communication. Therefore, Chap. 2 is fully devoted to visible light communications for cooperative ITS. The chapter presents the achievements of the experimental research in the scope of VLC prototyping for ITS. Special attention is devoted to the development of a VLC prototype based on IEEE 802.15. 7 standard, using low-cost embedded systems as the target platforms.

Strict real-time behaviour and safety guarantees are typically difficult to attain in vehicular ad hoc networks, but they are even harder to attain in high-speed mobility scenarios, where the response time of distributed algorithms may not be compatible with the dynamics of the system. In addition, in some operational scenarios, the IEEE 802.11p MAC may no longer be deterministic, possibly leading to unsafe situations. This calls for a reliable communication infrastructure with real-time, secure and safety properties, which is mandatory to support the detection of safety events and the dissemination of safety warnings. Therefore, Chap. 3 presents a proposal of a deterministic MAC protocol, the vehicular flexible time-triggered (V-FTT), which adopts a master multi-slave time division multiple access (TDMA), in which the road-side units act as masters to schedule the transmissions of the on-board units. The presented work analyses the proposed V-FTT protocol by quantifying an infrastructure deployment in motorways, particularly defining the usual coverage range for each RSU and the spacing between RSUs. A comprehensive survey on MAC protocols for vehicular networks, and especially targeting infrastructure TDMA-based deterministic protocols, has been presented in Chap. 4. In addition, the chapter presents a proposal for scheduling safety messages in the scope of wireless vehicular communications based on the V-FTT protocol.

Chapter 5 presents a comprehensive study on the efficiency of MAC protocols based on IEEE 802.11p/WAVE standard to timely deliver safety messages. Several aspects of an infrastructure-based MAC protocol, detail characteristics needed for safety-critical messages and bounded delay MAC protocols within specific scenarios, have been covered. Besides the V2I or I2V communication, there are situations where there is a need of relying exclusively on V2V-based communications to disseminate safety messages. Therefore, the chapter also presents an approach for cases where the infrastructure may not be accessible, or even not feasible to have total RSU coverage. Moving forward, Chap. 6 presents a direction-aware cluster-based multi-channel MAC protocol for vehicular ad hoc networks (VANETs) in which vehicles travelling in the opposite direction may result in a short communication period. How the cluster is made, and how the cluster head is elected based on the eligibility function that considers the number of connected neighbours, average speed deviation and the average distance between neighbours and itself, is elaborated. The chapter introduces direction-based clustering and multi-channel medium access control (DA-CMAC) protocol which aims to reduce access and merging collisions in the channel, by grouping the time slots into two sets based on the direction of movement.

Chapter 7 presents work on the predictable vehicular networks to provide reliability and predictability. The chapter shows how the MAC protocol for wireless mobile ad hoc networks can recover from timing failures and message collision and yet provide a predictable schedule in a time division fashion without the need for external reference. In addition, how mobile ad hoc networks and vehicular networks can organize themselves for emulating virtual nodes as well as emulating replicated state machines using group communication is presented. Vehicular networks are often facing the scalability problems due to high-speed mobility scenarios and under high dense vehicular environments. This results in high end-to-end delay and high packet drop rates; thus compromising the reliability of vehicular communications. Considering these challenging issues, Chap. 8 presents a fault-tolerant architecture to improve the dependability of infrastructure-based vehicular networks. The presence of road-side units (RSUs) and a backhauling network adds a degree of determinism that is useful to enforce real-time and dependability, both by providing global knowledge and supporting the operation of collision-free deterministic MAC protocols.

Chapter 9 explores and presents the development of the proactive handover mechanisms required to provide seamless connectivity and dependable communication in VANET environments. The chapter also presents classification of various handover mechanisms and proposes a new model of the handover process based on cumulative probability. In addition, results from simulation and analytical models have been presented, and a prototype is being deployed to further explore the issues associated with handover process. Chapter 10 presents work on the consideration of realistic road conditions for vehicular networks and elaborates a mathematical model that considers microscopic parameters. The model is able to capture the impact of road constraints such as traffic lights and road incidents on the traffic flow. It has been shown how the microscopic and macroscopic characteristics of vehicles moving on the roads are utilized for the improvement of vehicular connectivity dynamics. eCall is an initiative by EU with the purpose to bring rapid emergency assistance to an accident location. Hence, Chap. 11 shows how eCall is implemented via an Android phone using the cellular network and the IEEE 802.11p (ITS-G5) as communication medium. The main aim of the proposed system is to speed up the integration and implementation of eCall and accident detection mechanisms in legacy vehicles. In addition, this work provides a cost-effective and portable solution of eCall implementation. Experimental results related to accident and rollover detection are illustrated and discussed.

Finally, we express our sincere gratitude to Springer for giving us the opportunity and providing their valuable support throughout the completion of this book. We are also thankful to the contributors, reviewers and members of the Embedded Systems Group at Instituto de Telecomunicações, Aveiro Portugal for providing their valuable time and comments and helped us to review the contents of this book.

Aveiro, Portugal
October 2015

Muhammad Alam
Joaquim Ferreira
José Fonseca

Acknowledgements

This book covers topics related to the dependable vehicular communications for improved road safety, and its inspiration comes from the editors' contribution and experience at the forefront of the European research on intelligent transportation system (ITS). However, the editors would like to thank not only the collaborators that have contributed with chapters but also the Embedded Systems Group at the Instituto de Telecomunicações—Aveiro that have provided valuable comments and contributions toward the compilation of this book. We would also like to acknowledge the FP7 ICSI project that has contributed beyond state of the art on this important field of study.

Contents

1 **Introduction to Intelligent Transportation Systems** 1
Muhammad Alam, Joaquim Ferreira and José Fonseca

2 **Visible Light Communication for Cooperative ITS** 19
Mariano Falcitelli and Paolo Pagano

3 **Deterministic Vehicular Communications Supported by the Roadside Infrastructure: A Case Study** 49
Tiago Meireles, José Fonseca and Joaquim Ferreira

4 **STDMA-based Scheduling Algorithm for Infrastructured Vehicular Networks** 81
Luis Silva, Paulo Pedreiras, Muhammad Alam and Joaquim Ferreira

5 **Medium Access Control (MAC) Techniques for Safety Improvement** 107
Nuno Ferreira and José Fonseca

6 **Deterministic MAC Protocol Based on Clustering for VANETs** . . . 135
Unai Hernandez-Jayo, Aboobeker Sidhik Koyamparambil Mammu and Nekane Sainz

7 **Towards Predictable Vehicular Networks** 153
Elad Michael Schiller

8 **Fault Tolerant Architecture for Infrastructure based Vehicular Networks** 169
João Almeida, Joaquim Ferreira and Arnaldo S.R. Oliveira

9 **Exploring Seamless Connectivity and Proactive Handover Techniques in VANET Systems**. 195
Glenford Mapp, Arindam Gosh, Vishnu Vardhan Paranthaman, Victor Otite Iniovosa, Jonathan Loo and Alexey Vinel

10 Modeling Vehicles Mobility for Connectivity Analysis in VANET 221
Tariq Umer, Muhammad Amjad, Nadir Shah and Zhiguo Ding

11 HDy Copilot: A Mobile Application for Automatic Accident Detection and Multimodal Alert Dissemination 241
Bruno Fernandes, Muhammad Alam, Vitor Gomes, Joaquim Ferreira and Arnaldo Oliveira

Chapter 1
Introduction to Intelligent Transportation Systems

Muhammad Alam, Joaquim Ferreira and José Fonseca

Abstract Transportation systems are very important in modern life; therefore, massive research efforts has been devoted to this field of study in the recent past. Effective vehicular connectivity techniques can significantly enhance efficiency of travel, reduce traffic incidents and improve safety, alleviate the impact of congestion; devising the so-called Intelligent Transportation Systems (ITS) experience. This chapter aims to provide basic concepts and background that is useful for the understanding of this book. An overview of intelligent transportation systems and their applications is presented, followed by a brief discussion of vehicular communications. The chapter also overviews the concepts related to dependability on distributed real-time systems in the scope if ITS.

1.1 Cooperative Intelligent Transportation Systems

Transportation is fundamental for the human society; it allows the movement of people, animals and goods from one location to another. From the first domesticated animals and wheel carts to the modern cars and airplanes, transportation means and infrastructures have continuously evolved with an ever growing impact on our society, economy and environment.

During the past decades the volume and density of vehicles increased significantly, especially the road traffic; this lead to the increase of accidents and congestion, with negative impacts on the economy, environment and in the quality of people's lives [20]. In particular, according to the World Health Organization (WHO),

M. Alam (✉)
Instituto de Telecomunicações, Aveiro, Portugal
e-mail: alam@av.it.pt

J. Ferreira · J. Fonseca
Instituto de Telecomunicações, Universidade de Aveiro, Aveiro, Portugal
e-mail: jjcf@ua.pt

J. Fonseca
e-mail: jaf@ua.pt

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_1

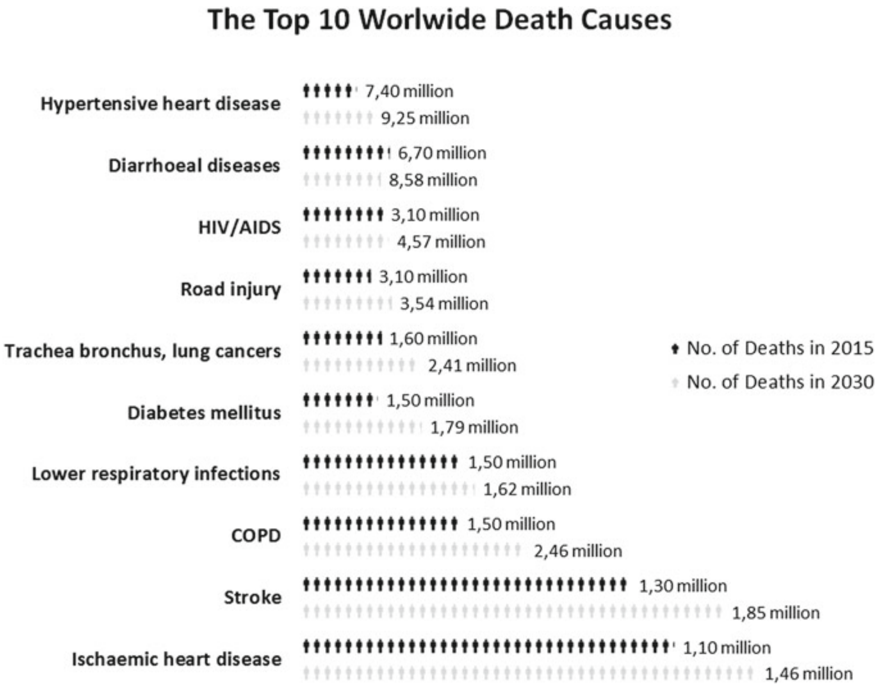


Fig. 1.1 Top 10 Worldwide death causes projections for 2015/2030

road traffic injuries are estimated to be the leading cause of death for young people (aged 15–29) and the ninth cause of death worldwide in 2015. Projections show that by the year of 2030 road injury related deaths will become the seventh main cause of death worldwide (Fig. 1.1) [21].

Although rules, regulations and methods to control the traffic (e.g. traffic signs, traffic lights) exist in most countries, these are becoming obsolete and unable to face the vehicle density growth. This problem is even more severe on developing countries such as Brazil and India [18, 20]. The construction and expansion of new roads and infrastructures could attenuate some of the existing problems, however, such solutions do not solve the fundamental issues and aren't sustainable since they require significant economic investments and land. Thus, new solutions are being sought out by the scientific and governmental organizations in order to improve traffic safety and efficiency.

1.1.1 Overview

Technological advancements, particularly in computer science and communication networks, paved the way for new methods and applications for traffic management;

Intelligent Transportation Systems (ITS) comprise a set of these new applications and are currently under heavy debate and work by governmental organizations and scientific communities. The initial ITS concept was proposed by the United States (US) in the 20th century; however, it is nowadays a subject of strong research and development worldwide, particularly in the US, Japan and the European Union (EU) [1]. Although ITS may refer to all modes of transport, the European Union (EU) defined its application in the field of road transport [11].

ITS integrate information and communication technologies (ICT) and apply them to the transport sector [10]. These systems gather data from sensors and equipment deployed within vehicles and infrastructures and provide services that aim to improve the current transportation system, making it more efficient, sustainable, safe, and environment friendly. ITS, such as Advanced Driver Assistance Systems (ADAS) for cars, electronic tolling systems, and traffic information systems, are currently deployed and found in the marketplace.

However, the first generation ITS, such as the aforementioned systems, are stand-alone, i.e. they are unable to share data and cooperate. A new subset of the overall ITS, in which participants communicate and share information to advise or facilitate actions, is currently under research focus. This subset, called Cooperative Intelligent Transportation Systems (C-ITS), aims to improve safety, sustainability, efficiency and comfort beyond the scope of stand-alone systems by taking advantage of the communication and cooperation between its participants. The type of information exchanged can be, for example, information regarding traffic jams, accidents and road hazards, amongst others [14].

To exchange information and increase the benefits of ITS services and applications, C-ITS rely on vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), infrastructure-to-vehicle (I2V) and infrastructure-to-infrastructure (I2I) communications. Special devices, deployed within vehicles and infrastructures, make use of technologies such as Dedicated Short-Range Communications (DSRC) to enable the aforementioned interactions. An overview of these technologies is presented in Sect. 1.2 of this chapter.

1.1.2 Applications

Recent research studies show that the incorporation of information and communication technologies with transportation infrastructure and vehicles will revolutionize the way we travel today. The enabling technologies are intended to realize the frameworks that will spur an array of applications and use cases in the domain of road safety, traffic efficiency, and driver's assistance. Although these applications will allow dissemination and gathering of useful information among vehicles and between transportation infrastructure and vehicles in pursuance of assisting drivers to travel safely and comfortably; still much efforts are required to implement these practices for the success of these applications.

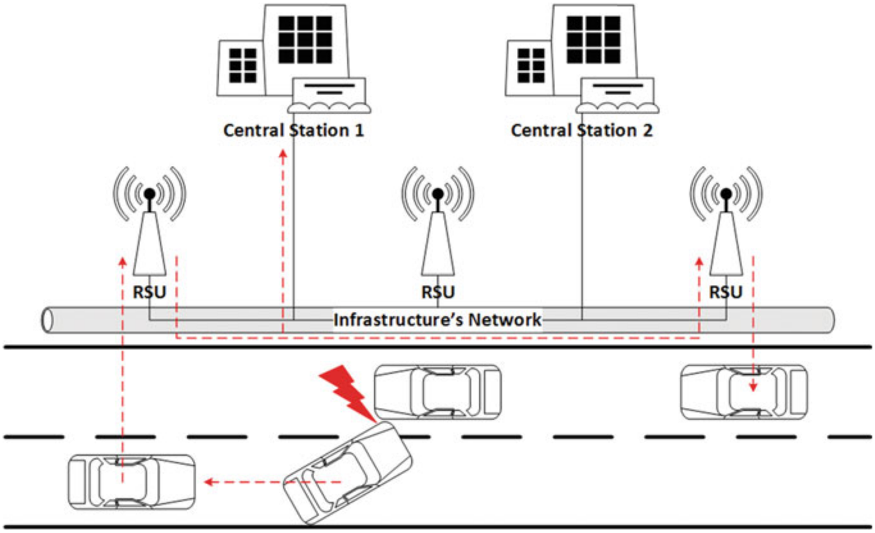


Fig. 1.2 Cooperation in C-ITS

By taking advantage of the communication capabilities of C-ITS participants, it is possible to develop new services and applications that, by gathering and sharing different types of information, can help in improving the level of safety, efficiency and comfort of modern transportation systems. For example (Fig. 1.2), a vehicle that is involved in a traffic collision, can immediately send an alert message to all vehicles in the vicinity through V2V communication; upon the reception of this message each vehicle can determine the most appropriate response, e.g. warn the driver or perform an evasive maneuver. If in range with a roadside unit (RSU), these vehicles are able to report the event and its location to a central system and alert the responsible authorities to take appropriate measures, e.g. provide the necessary emergency means.

Moreover, roadside units can monitor and broadcast information on the current traffic density and advise alternative routes to vehicles in the vicinity in order to avoid congestions and increase traffic efficiency.

Besides these basic use case scenarios, it is possible to imagine numerous challenging scenarios and applications that can take advantage of C-ITS communications. The European Telecommunications Standards Institute (ETSI) EN 302 665 standard has defined a "Basic Set of Applications" (BSA) which is composed of three main application classes [3, 9]:

- Road/Traffic Safety
- Traffic Efficiency
- Other Applications (Value-Added)

1.1.2.1 Road/Traffic Safety Applications

Traffic safety applications aim at reducing the risk of car accidents and at minimizing the resulting damage of unavoidable accidents. Due to its nature and importance, these applications impose the most demanding requirements, requiring dedicated reliable hardware as well as reliable and timely communications. These applications include cooperative awareness applications, e.g. headway management, lane departure warning and speed management, as well as hazard warning applications, e.g. hazard and adverse weather detection.

Traffic safety applications strongly rely on the exchange of two types of safety messages that have been standardized by ETSI [6, 7]:

- Cooperative Awareness Messages (CAMs)
- Decentralized Environmental Notification Messages (DENMs)

The Cooperative Awareness Messages (CAMs) are time-triggered position messages responsible to produce and maintain awareness between the ITS stations in the ITS network. A CAM message is packed with the status and attribute information of the vehicles or station that generates it. These information packed messages are periodically disseminated to the one hope neighbouring ITS stations and thus creates awareness in the system. The contents of CAM varies depends on the type of vehicle. The status information includes position, time, mobility status, etc. while the attribute information represents the attributes of the station such as vehicle type, dimension, role in the road traffic etc. Therefore, by receiving CAMs, each ITS station is aware of other stations in its neighbourhood area as well as their positions, movement, basic attributes and basic sensor information.

Another type of messages defined by ETSI standard are the event-driven based Decentralized Environmental Notification Messages (DENMs) mainly used by the Cooperative Road Hazard Warning (RHW) application in order to alert users of the road detected events. The DENMs carrying useful active road safety messages are broadcasted among vehicles ITS station and roadside ITS stations via V2V/I2V/V2I communication. For example, the roadside ITS stations may collect the broadcasted information from vehicle ITS stations, process the information and forward the information to a central ITS station thus enhance the traffic efficiency and result in better traffic management.

Both CAM and DENM messages are broadcasted to vehicles within a particular geographic region. For CAMs this region usually corresponds to the immediate neighborhood while for DENMs it is usually the area potentially affected by the notified event (which can span over several hundred meters). The main requirements of CAMs and DENMs are described in Table 1.1. Notice that equivalent types of messages were also standardized in the United States by IEEE and SAE.

Table 1.1 C-ITS traffic safety messages' requirements

	Frequency	Max. latency	Coverage	Length	Use case examples
CAM	1–10 Hz	100 ms	300 m up to 20 km	Up to 800 bytes	• Emergency vehicle warning • Collision risk warning
DENM	N/A	100 ms	300 m up to 20 km	Typically shorter than CAMs	• Signal violation warning • Hazardous location

1.1.2.2 Traffic Efficiency Applications

The main goal of traffic efficiency applications is the improvement of traffic fluidity by reducing travel time and traffic congestion; indirect economic and environmental benefits can also be obtained. These applications provide traffic information to the user, usually disseminated by roadside infrastructures.Traffic efficiency applications include inter-urban efficiency applications, e.g. adaptive electronic traffic signs and route guidance and navigation services, urban traffic efficiency applications, e.g. traffic flow optimization services, and freight/fleet applications, e.g. management of hazardous goods vehicles.

Although these applications don't present strict delay and reliability requirements, their quality gracefully degrades with the increase of delay and packet loss. The common requirements for this type of applications are described in Table 1.2.

1.1.2.3 Value-Added Applications

Value-added applications provide comfort and convenience applications for the users. These include infotainment, travel information, journey planning, Internet access, amongst others. Although the requirements for these applications are highly

Table 1.2 C-ITS traffic efficiency application requirements

	Frequency	Max. latency	Coverage	Use case examples
Traffic efficiency applications	1–10 Hz	200 ms	300 m up to 5 km	• Intersection manager • Optimal speed advisory

Table 1.3 C-ITS value-added infotainment requirements

	Min. frequency	Max. latency	Coverage	Use case examples
Traffic efficiency applications	1–10 Hz	200 ms	300 m up to 5 km	• Media download • E-mail

dependent on the application type, common applications can tolerate long delays (up to a certain degree) and may occasionally demand high data throughput. Typical requirements for infotainment applications are described in Table 1.3.

1.1.3 Architecture

1.1.3.1 Sub-systems

According to [8] there are four types of communicating entities (sub-systems) within C-ITS (Fig. 1.3):

- **Personal**: provides access to ITS applications through personal, user devices (e.g. Smartphone with a route guidance application);

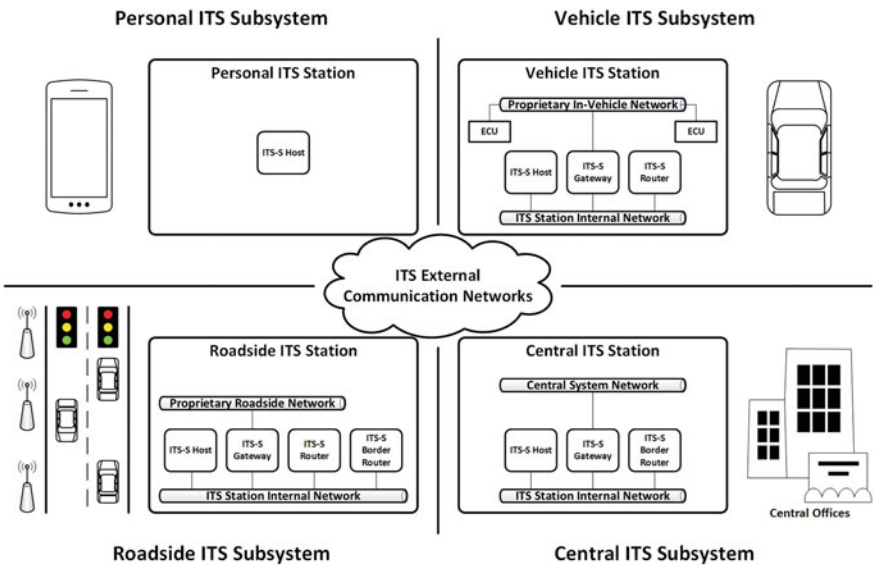


Fig. 1.3 European ITS communication sub-systems

- **Vehicle**: equipment on-board the vehicle (on-board unit or OBU) which hosts ITS applications. These applications can collect information about the vehicle and its environment, receive and/or provide information to the driver, partly or fully control the vehicle in critical situations;
- **Central**: equipment operated by the entities in charge of the different ITS applications. Used to maintain, monitor and provide functionality to ITS applications;
- **Roadside**: equipment installed along the roadside which hosts ITS applications (Roadside unit or RSU). These applications can collect information about the traffic flow and road environment (e.g. weather), control roadside equipment (e.g. traffic signals) and communicate with the vehicle's ITS sub-systems to provide/collect information.

All ITS subsystems are built upon the same reference architecture, i.e. every subsystem has an ITS station (ITS-S) as core component. An ITS-S hosts the different ITS applications and communicates with other components within the subsystem and other ITS-S. For example, vehicle's ITS subsystems may consist of a vehicle ITS-S and their in-vehicle network of sensors and Electronic Control Units (ECUs), while roadside's ITS subsystems may consist of a roadside ITS-S and roadside sensors, cameras, signs and signals. ITS stations can be inter-linked by a communication network, which is typically composed by a backbone network and a large number of edge and access networks. Communication between ITS stations should be seamless and independent of the type of subsystem [8].

The ITS station's architecture follows the principles of the OSI model [13] for layered communication protocols, and it was extended to include ITS applications. The ITS-S reference architecture is illustrated in Fig. 1.4.

In the ITS-S architecture, the functionality of the OSI layers 1 and 2 is represented by the "Access" block, layers 3 and 4 by the "Networking & Transport" block, and layers 5, 6 and 7 by the "Facilities" block.

The "Applications" block represents the ITS-S applications. These can use the services from other layers to connect to other ITS-S applications. An ITS application results from the complementary association of two or more ITS-S applications and it provides an ITS service to an ITS user.

The "Management" block is in charge of managing communications within the ITS station while the "Security" entity provides security services.

An ITS-S can be composed of the following functional elements:

- **ITS Station Host**: provides access to ITS applications through personal, user devices (e.g. Smartphone with a route guidance application);
- **ITS Station Gateway**: ITS-S gateways interconnect two different OSI protocol stacks at layers 5 to 7 and are capable of converting protocols. They provide connection to external, proprietary networks (e.g. in-vehicle networks)
- **ITS Station Router**: ITS-S routers interconnect two different ITS protocol stacks at layer 3, and are capable of converting protocols. They provide connection to other ITS-S (e.g. vehicle ITS-S and roadside ITS-S)

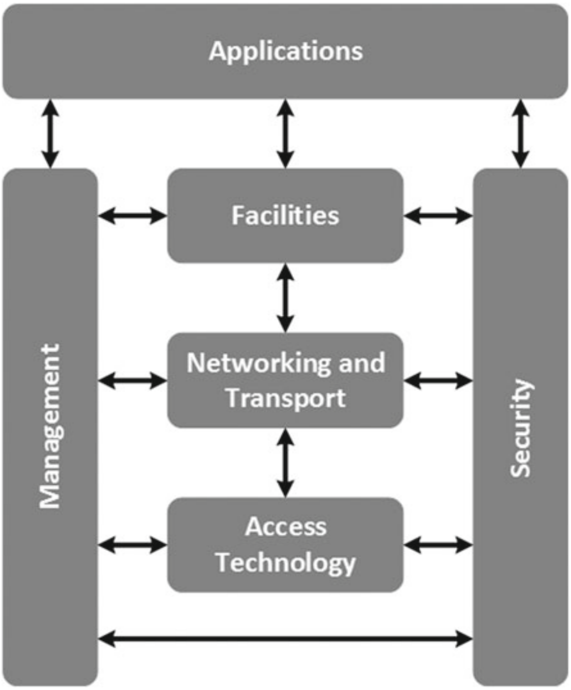


Fig. 1.4 ITS-S reference architecture

- **ITS Station Border Router**: ITS-S border routers nearly provide the same functionality as the ITS-S routers with the difference that the external network may not support the same management and security principles of ITS.

1.1.3.2 Networks

ITS stations rely on communication networks for communication and cooperation. The network architecture of C-ITS is composed by both external and internal networks. External networks interconnect ITS stations (e.g. a vehicle ITS-S to a roadside ITS-S) or connect ITS stations to other network entities (e.g. a server in the Internet). Internal networks interconnect ITS-S components (e.g. ITS-S host and ITS-S gateway). The most relevant types of external networks on C-ITS are illustrated in Fig. 1.5 [8].

ITS ad-hoc networks enable direct communication among vehicle, roadside and personal ITS stations through short-range wireless technologies. This type of network allows great mobility and flexibility without the need of a coordinating entity. A typical ITS ad-hoc network is a network of vehicle and roadside ITS stations interconnected by IEEE 802.11p wireless technology.

ITS access networks are dedicated networks, usually deployed by private road operators, that provide access to specific ITS services and applications and that may

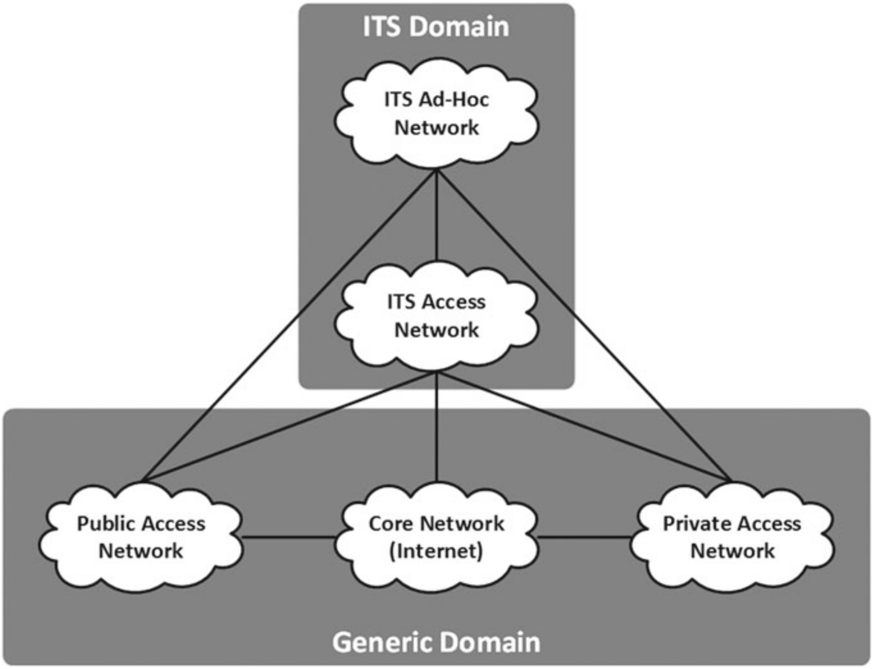


Fig. 1.5 ITS-S external networks

interconnect roadside ITS stations. Vehicle ITS stations can communicate between themselves via roadside ITS stations that are interconnected, instead of using an ad-hoc network. For example, an ITS access network can connect roadside ITS stations in a highway with a central ITS station (e.g. road traffic management center).

Public access networks provide access to general purpose networks that are publicly accessible. An example is a LTE network which provides Internet access to a vehicle ITS-S. Private access networks provide data services and secured access to another network for a restrict group of users. For example, a private access network can connect vehicle ITS stations to a company's intranet.

In addition to the above described networks, ITS stations can also be attached to proprietary local networks, for example, a vehicle ITS-S can connect to the in-car CAN network (Fig. 1.3) through a ITS-S Gateway. While great emphasis has been put into the communication between vehicles and infrastructures, communication with these proprietary networks has received far less attention [17].

Each different network provides support for at least one C-ITS use-case (e.g. road safety, traffic efficiency, infotainment, business applications, …), however, it is presumed that a single network isn't able to meet all the requirements for all applications and thus, combinations of multiple networks, comprising several ITS access and networking technologies, are expected.

Despite the possible communication combinations between ITS stations, the overall ITS environment puts great emphasis on direct communication between ITS-S [9]:

- from a vehicle ITS-S to another vehicle ITS-S (Vehicle-to-Vehicle or V2V)
- from a vehicle to a roadside ITS-S (Vehicle-to-Infrastructure or V2I)
- from a roadside to a vehicle ITS-S (Infrastructure-to-Vehicle or I2V)

1.2 Vehicular Communication Standards

Vehicular communications are an important field of research in the area of Intelligent Transportation Systems. ITS requires wireless communications among vehicles and between vehicles and the road side infrastructure. Vehicular communication systems can be more effective in preventing road accidents than the case where vehicles work individually to achieve the same goal. This is due to the cooperative techniques that can be exploited when vehicles and the roadside stations have available information about others parties situation (e.g. location, speed and heading). As an example of this class of safety applications, chain collisions could be avoided if the information about the first crash is disseminated by all the other nodes in the vicinity of the accident.

Dedicated Short-Range Communications (DSRC) is a wireless technology that has been designed to support a variety of applications based on vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications. Vehicular communications supported by DSRC systems operate in the 5.9 GHz reserved spectrum band and have an approximate maximum range of 1000 m. There are two main protocol architectures for vehicular communication systems, one developed by the Institute of Electrical and Electronics Engineers (IEEE) and the other one from the European Telecommunications Standard Institute (ETSI), as illustrated in Fig. 1.6.

The protocol stack in America is known as IEEE Wireless Access in Vehicular Environments (WAVE), while in Europe is referred as ETSI ITS-G5. Both of these standards rely on IEEE 802.11p, from the IEEE 802.11 family of Wi-Fi standards, for the implementation of physical (PHY) and medium access control (MAC) layers [12]. The physical layer is almost identical to IEEE 802.11a, using OFDM with BPSK, QPSK, 16-QAM and 64-QAM modulations, but with double timing parameters to attenuate interference resulting from the multi-path propagation and the Doppler shift effects. With double timing parameters, the channel bandwidth is 10 MHz instead of 20 MHz, and the data rate is half, i.e., 3 . . . 27 Mbit/s instead of 6 . . . 54 Mbit/s.

The medium access control (MAC) layer adopts a carrier sense multiple access with collision avoidance (CSMA/CA), as IEEE 802.11a, but it is adjusted for the vehicular communication environments, which differs significantly from the sparse and low-velocity characteristics of a traditional Wi-Fi deployment. In vehicular environments, nodes present high mobility, some areas are often densely populated and frequently there is non-line-of-sight. As consequence, some tweaks were introduced

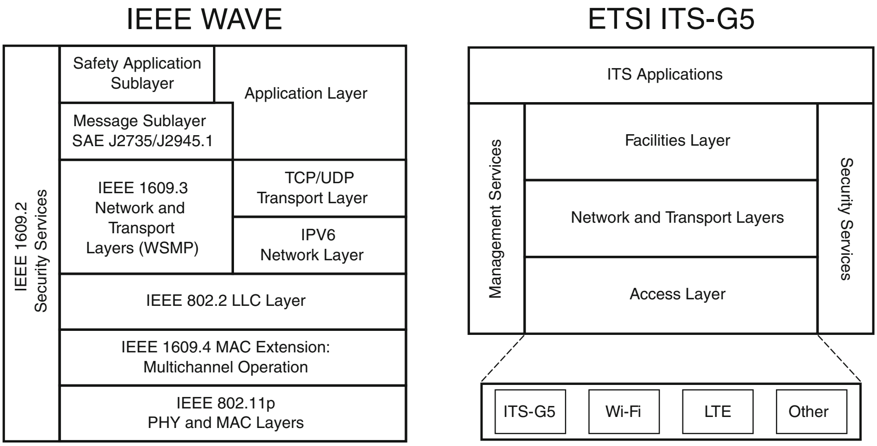


Fig. 1.6 IEEE WAVE and ETSI ITS-G5 protocol stacks

in the standard to allow low overhead operations, in order to guarantee fast and reliable exchange of safety messages. For example, non-IP messages that operate outside the context of a Basic Service Set (BSS) were defined, enabling a quick transmission of packets by avoiding the registration and authentication procedures, commonly present in typical wireless local area networks.

The Federal Communications Commission (FCC) in the United States and the European Conference of Postal and Telecommunications Administrations (CEPT) allocated a dedicated spectrum band at 5.9 GHz (Fig. 1.7) for vehicular communications. In America, a bandwidth of 75 MHz was reserved, while in Europe only 50 MHz were assigned. This spectrum was divided into smaller 10 MHz wide channels and in the American case, a 5 MHz guard band at the low end was also included.

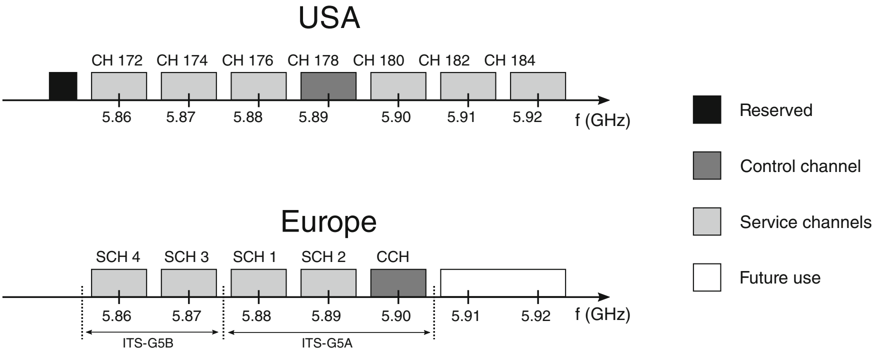


Fig. 1.7 Spectrum allocation for vehicular communications in USA and Europe

As a result, there are 7 different channels for IEEE WAVE operation and 5 for the case of ETSI ITS-G5.

In Europe, 30 MHz (3 channels) are reserved for road safety in the ITS-G5A band and 20 MHz are assigned for general purpose ITS services in the ITS-G5B band. As a general rule, a control channel (CCH 178 in the USA and CCH 180 in Europe) is exclusively used for cooperative road safety and control information. The remaining channels are designated as service channels (SCH). In the United States, concerns about the reduced capacity for road safety messages led to the decision to allocate SCH 172 specifically for applications regarding public safety of life and property.

1.3 Dependable Distributed Real-Time Systems and ITS

1.3.1 Distributed Systems and ITS

Distributed systems can be defined as systems composed by several processing units that communicate through a network to execute a set of activities in a distributed manner. Contrary to centralized systems, where all the computing is done in a single, central node without interaction with other computer systems, in distributed systems a set of nodes interconnected by network, cooperate and exchange information in order to achieve a common goal. Sensor networks, automated assembly lines, peer-to-peer networks and aircraft control systems are typical examples of distributed systems [19].

The network is one of the most, if not the most, important component of a distributed system. It enables nodes, which can be deployed in geographically separate locations, to synchronize and exchange information and thus cooperate. Such important element must be designed with utmost care. If the network is overloaded (i.e. system resources demand exceeds the maximum available) or isn't able to fulfill the specific requirements of each communication stream (e.g. packet delay, drop rate), the system may experience a performance degradation and in the worst case, a partial or global system failure (i.e. a part or the whole system may be unable to communicate effectively and thus, unable to properly operate) [5, 19].

Vehicular communication systems are inherently distributed, since the different nodes of the network are physically apart and exchange data and cooperate to achieve the common goal of guaranteeing traffic safety. The design of vehicular systems, pose even more challenges when compared to the design of traditional distributed systems, because vehicular environments present high mobility and unpredictable link conditions, characteristics that do not arise in typical static networks.

1.3.2 Real-Time Systems and ITS

In some occasions, distributed systems present timeliness requirements that are dictated by the environment in which they operate. Since the environment has inherent temporal dynamics, in order to properly interact with it, these systems not only have to produce logically correct solutions but also need to apply them within a specified time intervals. Systems, where the correctness of the system behaviour depends on both the logical computations and the physical time instant when they are produced and applied are called real-time systems [4]. These systems can be found, for example, in industrial automation systems, automotive applications, flight control systems and military applications.

Real-time systems are usually composed by computational activities, i.e. tasks, which implement specific functionalities and have stringent timing constraints that must be met in order to achieve proper behaviour. A typical constraint on a task is the deadline, i.e. the instant before which a task should complete its execution without impairing the system. Depending on the effects of a missed deadline, tasks can be categorized as [4]:

- **Non real-time**: task has no time constraints and always contributes to the system whenever it completes its execution;
- **Soft**: task's output still has some utility to the system after missing its deadline, however, the system's performance is degraded;
- **Firm**: task's output has no utility to the system after a deadline miss, however, it does not cause catastrophic consequences on the system behaviour;
- **Hard**: task only contributes to the system if it completes within its deadline. A deadline miss may cause catastrophic consequences, e.g. overall system failure with human and/or material losses.

Real-time systems can be categorized as soft or hard according to the supported task types and the consequences raised by deadline misses [4, 15]:

- **Soft Real-Time Systems**: Systems that only integrate soft and/or firm tasks are categorized as soft real-time. In these systems, deadline misses may induce overall performance degradation without catastrophic consequences. A typical example for this type of system is video and sound streaming in which a deadline miss typically results in minor image/sound glitches.
- **Hard Real-Time Systems**: Systems that contain at least one hard task are categorized as hard real-time. In these systems, a deadline miss may result in a system failure with catastrophic effects, e.g. material and/or human losses. A typical example for this type of system is a nuclear power plant control in which a deadline miss could result in the failure of the nuclear reactor.

As stated earlier, proper temporal behaviour is required for the correct operation of real-time distributed systems. The temporal behaviour of the whole system depends on several elements such as the node's software, e.g. running tasks, behaviour and the capacity of the underlying communication system to provide timely delivery of messages. Communication systems capable of delivering messages within specific temporal constraints are known as real-time communication systems [15].

The design of ITS in general and specifically vehicular communication systems should take into account the fact that strong real-time constraints are present in this type of scenarios, and therefore vehicular networks supporting safety-critical applications should be analysed as hard real-time distributed systems. For instance, in case of accident, the vehicles approaching the location of the hazard should receive a warning message with sufficient time in advance, in order for them to take appropriate measures, avoiding a possible chain collision. If these hard deadlines cannot be met, catastrophic consequences may occur, possibly causing human, economic and environmental losses. Beyond that, this type of safety-critical systems must exhibit a high probability to provide continuous correct service, in order to guarantee that real-time activities are performed within stringent bounds. This usually implies that several dependability aspects are taken into consideration during the design of the system, which will be further explained next.

1.3.3 Dependability and ITS

Dependability is a generic concept that describes the level of trust one can have in the operation of a system. According to Laprie et al. [2, 16], dependability can be divided in 3 different classes of notions—attributes, threats and means—as presented in Fig. 1.8.

The attributes of dependability denote different properties that can be expected from a dependable system, whose importance can vary between distinct applications:

- **Availability** is defined as the readiness to provide a correct service (even after failures).
- **Reliability** is the probability of a system to present continuous correct service.

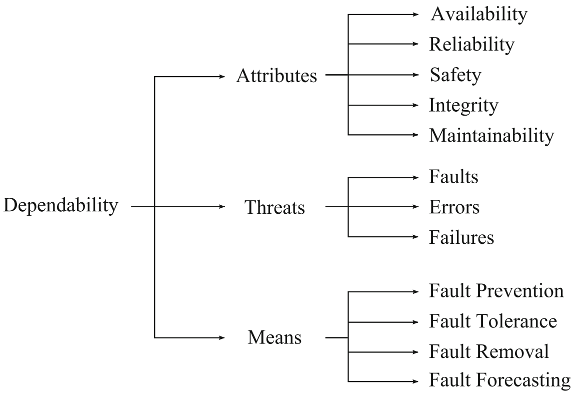


Fig. 1.8 The dependability tree, according to Avizienis et al. [2]

- **Safety** is defined as the absence of tragic consequences on the operating environment.
- **Integrity** is the absence of improper system alterations.
- **Maintainability** is defined as the capacity of the system to undergo modifications and repairs.

The impairments or threats are the undesired circumstances that can prevent a system from being dependable. There are typically three main terms associated with the threats to dependability: faults, errors and failures. A fault is a defect in the design or in the operation of the system that can lead to an error. An error is an incorrect value of the total system state that may cause an failure. Finally, a failure or service failure is an event that occurs when the delivered service of the system present some deviation relatively to the correct behaviour.

During the design of safety critical systems, several means or techniques can be used to attain the various attributes of dependability. The purpose of these techniques is to reduce the impact of faults in the overall system's operation:

- **Fault prevention** deals with preventing faults from occurring or being introduced in the system.
- **Fault tolerance** comprises methods to avoid system failures and the provision of service complying with its specifications even in the presence of faults.
- **Fault removal** techniques aim to reduce the number and severity of faults.
- **Fault forecasting** methods try to detect the present number of faults in the system, as well as their future occurrence and consequences.

In vehicular environment, dependability attributes are of uttermost importance, since a failure in system's operation can cause severe consequences. As a result, one should prevent failures to occur, and for that purpose, the techniques mentioned above must be considered in the design of vehicular communication systems. There are numerous situations where the various attributes of dependability can be effectively utilized in ITS domain. For instance, in high mobility scenarios the users (drivers or passengers) can experience the connections drop because of service provider or some components between the service provider and the users have failed. Therefore, the fault tolerance attribute can be used in the location based services by discovery protocol for vehicular networks to guarantee the requested service.

As stated earlier, traffic fatalities are one of the major causes of death in the world and vehicular communication has emerged as one of promising technology to improve the safety of drivers, passengers, and pedestrians on the road. These systems are considered subject to the unreliable characteristics of distributed systems and also linked by wireless communication. Vehicle communicate with each other important messages but are these messages reliable and trustworthy? Similarly, there are numerous other questions that need to be answered.

Therefore, new design aspects in this class of systems considering new architectures, applications, and communication mechanisms based on dependability attributes need to be proposed.

References

1. S. An, B.-H. Lee, D.-R. Shin, A survey of intelligent transportation systems, in *3rd International Conference on Computational Intelligence, Communication Systems and Networks*, July 2011, pp. 332–337
2. A. Avizienis et al., Basic concepts and taxonomy of dependable and secure computing. IEEE Trans. Dependable Secure Comput. **1**(1), 11–33 (2004)
3. R. Bossom et al., *Deliverable D31 European ITS Communication Architecture—Overall Framework* (2009)
4. G.C. Buttazzo, *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*, 3rd edn. (Springer Publishing Company, Incorporated, 2011). ISBN: 1461406757, 9781461406754
5. G. Coulouris et al., *Distributed Systems: Concepts and Design*, 5th edn. (Addison-Wesley Publishing Company, USA, 2011)
6. ETSI, ETSI EN 302 637-2 V1.3.2: Part2: Specification of Cooperative Awareness Basic Service (2014)
7. ETSI, ETSI EN 302 637-3 V1.3.2: Part3: Specification of Decentralized Environmental Notification Basic Service (2014)
8. ETSI, ETSI EN 302 665 V1.1.1: Intelligent Transport Systems (ITS)—Communications Architecture (2010)
9. ETSI, ETSI TR 102 638 V1.1.1: Basic Set of Applications—Definitions (2009)
10. European Commission, European Commission Mandate M/453 EN (2009)
11. European Parliament, Directive 2010/40/EU (2010)
12. IEEE Standard for Information Technology–Telecommunications and information exchange between systems local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007), Mar 2012, pp. 1–2793
13. International Standard Organization. ISO/IEC 7498-1 (1994)
14. P. Kompfner et al., *Deliverable D3.2 Multimodal Cooperative ITS Architecture: A First Concept* (2009)
15. H. Kopetz, *Real-Time Systems: Design Principles for Distributed Embedded Applications*, 2nd edn. (Springer Publishing Company, Incorporated, 2011). ISBN: 1441982361, 9781441982360
16. J.C.C. Laprie, A. Avizienis, H. Kopetz (eds.), *Dependability: Basic Concepts and Terminology* (Springer-Verlag New York Inc, Secaucus, 1992)
17. B. Oehry et al., *ITS Action Plan—Final Report Action 4.1* (2010)
18. G. Singh, D. Bansal, S. Sofat, Intelligent transportation system for developing countries—a survey. Int. J. Comput. Appl. **85**(3), 34–38 (2014)
19. A.S. Tanenbaum, M. Van Steen, *Distributed Systems* (2007)
20. K. Watkins, *Safe and Sustainable Roads: The Case for a Sustainable Development Goal* (2012). http://www.fiafoundation.org/media/44116/sustainable-transport-goal-report-2012.pdf
21. World Health Organization, Mortality 2015 and 2030—Baseline Scenario (2015). http://www.who.int/healthinfo/global_burden_disease/projections/en/. Accessed 28 July 2015

Chapter 2
Visible Light Communication for Cooperative ITS

Mariano Falcitelli and Paolo Pagano

Abstract Visible Light Communication (VLC) is the technique adopting electromagnetic frequencies in the visible spectrum for free space optical communications. Although its practical use is still at early stages, in the last few years research activities have been exploring different solutions to achieve high data rates and reliable links using common LEDs and light sensors. VLC can be used in a variety of applications or end user segments, exploiting already existing lighting infrastructures and thus making VLC a cheap communication system. Among these applications, a prominent case study is that of ITS (Intelligent Transportation Systems), where car headlamps and traffic lights can be used to communicate and fulfil the requirements of road safety applications. This option turns to be particularly effective in short range direct communications to exploit its line-of-sight feature and overcome the issues related to the isotropic nature of radio waves. Recently IEEE undertook standardization activities on VLC, resulting in the IEEE 802.15.7 standard, which disciplines PHY and MAC layer services for Visible-light Personal Area Networks (VPANs). This chapter shows the recent achievements of the experimental research in the scope of VLC prototyping for ITS. Special attention is devoted to the development of a VLC prototype based on IEEE 802.15.7 standard, using low cost embedded systems as the target platforms. The aim is to provide useful considerations for achieving devices suitable to be integrated in existing PANs, or to cooperate with other wireless networks to provide communication services in complex architectures like ITS.

2.1 Introduction

Visible light communication (VLC) is an emergent wireless communication technology, which uses white or coloured LEDs to provide information through visible light as the communication medium. VLC transmits data using all the frequencies

M. Falcitelli (✉) · P. Pagano
CNIT - National Laboratory of Photonic Networks, Pisa, Italy
e-mail: mariano.falcitelli@cnit.it

P. Pagano
e-mail: paolo.pagano@cnit.it

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_2

between 400 THz (750 nm) and 800 THz (375 nm) by intensity modulating the light sources faster than the persistence of the human eye. In the past few years, LEDs improvements in switching rates, brightness increase, and large scale diffusion drew the attention of research communities, which started looking at visible light as a new communication medium, complementary to radio frequencies, which are becoming more and more congested.

High power LED devices have benefits such as energy savings, long life, low maintenance cost, low temperature generation, better visibility and high brightness, all compared to those of the incandescent lights or fluorescent lights. For these reasons the traffic signals and the head lamps of the vehicles are gradually changing from electric light bulbs to LED lights.

The combined lighting and switching feature of LEDs has a strong innovative potential and it will produce important applications. For example LED-based traffic lights and vehicular VLC systems can become an integrated component of ITS and play a key role in road safety applications by broadcasting traffic information in advance to drivers running vehicles which will incorporate low cost VLC receivers.

Under certain aspects VLC can be considered to be relative to infrared (IR) optical wireless optical communication (OWC). The latter had a slow but constant evolution during the last century, bringing applications in the scope of short-range and low-data-rate communication. The infrared light can be found in our daily life, where the best known example is the remote control of electronic domestic devices. Nevertheless so far, IR wireless communication remains of secondary importance respect to short range RF based technologies, such as Bluetooth, and it have not evolved into broader scope, such as dependable alternative for broadband access networks (Fig. 2.1).

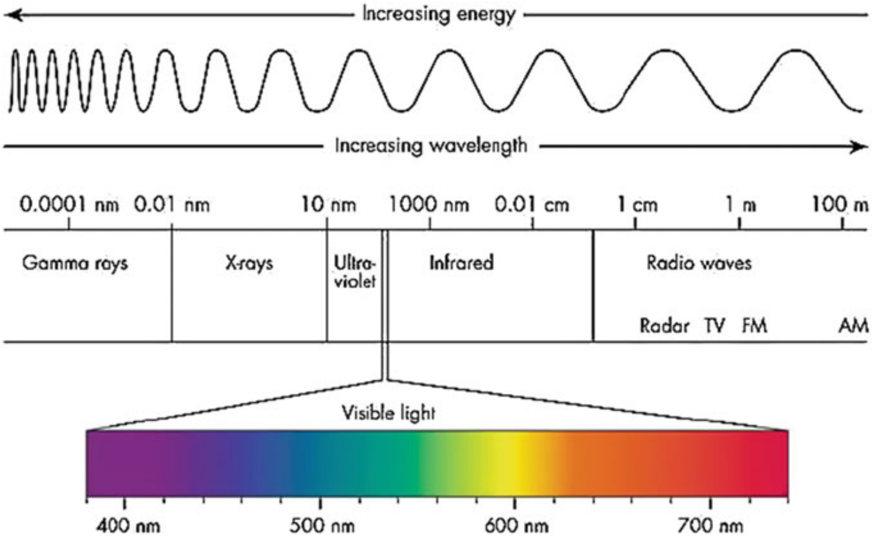


Fig. 2.1 Visible light spectrum

VLC is really a subset of OWC that has become a technology in itself because, in this case, the signal carrier can be seen by the human eye. As distinguishing feature, VLC provides illumination as well as communication, while traditionally OWC has been concerned just with communications. On the other hand VLC systems must operate through illuminating devices with eye safety constraints and they should be able to provide communication also when the illuminating light is dimmed or even turned off.

The potential use of the same device for simultaneous data transmission and illumination is tempting and fascinating. At first glance, the power and money already invested in providing illumination could be reused to facilitate high-data-rate communication between light sources and users. LED luminaries could act as network access points, turning VLC into a direct competitor to broadband radio technologies such as WiFi, fourth/fifth generation (4G/5G) systems and WiGig. The Visible Light (VL) spectrum is unlicensed and currently largely unused for communication, the availability of this free spectrum creates an opportunity for low-cost broadband communication that can help the more used bands. About this, indoor hybrid systems, comprised of RF technology and VLC links, in which directional broadcast VLC channels are exploited to supplement conventional RF channels start to be considered and investigated by means of simulation studies [7, 27]. Nevertheless, to date there are still no assessed solutions for the seamless integration of VLC with conventional wireless networks. The design of VLC systems is still challenging, because the specific properties of the medium offer both new problems and new possibilities. The most important VLC features are outlined below.

Line of Sight (LoS). In order to establish an optical link, LoS is requested between the transmitter and the receiver. This could be a major issue, as devices mobility or obstacles moving between transmitter and receiver can disrupt communications. Moreover, natural and artificial lights add noise and interference to the channel. When used outdoor, bad weather conditions like rain, snow, fog can further alter light signal.

Unlicensed spectrum. Visible light is an unrestricted very large (400 THz wide) spectrum available worldwide. This is in contrast with infrared light (IR) or radio frequency (R/F) technologies, which are limited by law and limited in band; many R/F frequencies are restricted for special applications (military, aircraft, etc.).

Healthy. Visible light is safe to human body, which makes it possible to transmit with high power, while radio waves are concerned to be dangerous to human body and infrared light may be harmful to human eyes.

No electromagnetic interference. VLC is resistant to electromagnetic noise and in turn does not cause electrosmog. It can be used in places where radio waves cannot be used, for examples, hospitals and areas around precision machines.

Security. Visible light communication requires LoS, and don’t penetrate through walls, while radio frequencies do. Communication is then limited to the area in which it originates. This property can be exploited to hide data communications from potential eavesdroppers.

High spatial reuse. Consider circumstances where many devices compete for wireless medium access, for instance classrooms, conference halls and other assembly spaces. Traditional wireless can hardly handle lots of users, which in turn experience degraded performances. Since VLC is high directional, a single optical link could for instance originate from a lamp in the ceiling pointing directly to the floor, so that only a few users share the link. Spacial reuse allows then to accommodate larger number of VLC devices without interference as in the wireless case.

Ubiquitous computing. VLC can be used as a communications medium for ubiquitous computing, because light producing devices, such as indoor lamps, commercial displays, traffic lights, outdoor lamps, etc. are used everywhere.

2.2 VLC Architecture and Expected Applications

A typical VLC architecture, as shown in Fig. 2.2 comprises transmitter entities and receiver entities, communicating by modulated visible light. Communicating entities can be end devices such as mobile personal devices, vehicles, and infrastructure lights. Each entity transmits and receives data by means of a VLC emitter and a VLC receiver, respectively. The VLC emitter is an optoelectronic transducer that transmits information using visible light as the physical transmission medium; high brightness LEDs are commonly used. LEDs are modulated at such high frequencies that human eye cannot perceive any difference in lighting compared to that when there is not modulation. As a result, VLC transmitters can be used for lighting and data communication simultaneously. The VLC receiver is an optoelectronic transducer (PIN photodiode or avalanche photodiode or CMOS sensor) that receives information, previously modulated in the visible light spectrum, and converts it into electrical signals than can be processed by a demodulator/decoder.

Three types of topologies are possible for the VLC link: directed Line-of-Sight (LOS), non-directed LOS, diffused non-LOS. As depicted in Fig. 2.3, the directed LOS allows the highest intensity for the received signal and thus it has the highest bitrate and the longest distance are achieved at the expense of severe demand of precise alignment; in the non-directed LOS the receiver has a wider field of view, the alignment is simpler, but the intensity of the signal is at medium level, so shorter

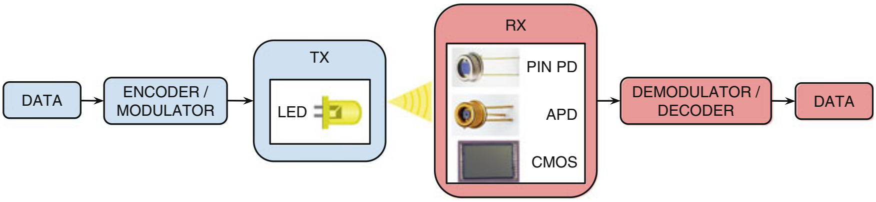


Fig. 2.2 Architecture of VLC system: transmission-reception chain

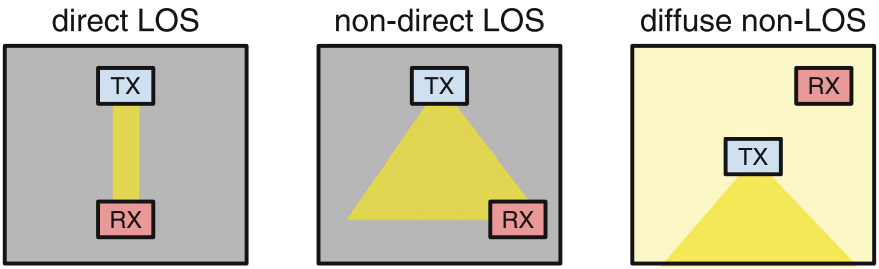


Fig. 2.3 Possible topologies of the VLC link

distances are achievable together with high/medium bitrate; the diffuse non-LOS is free form alignment issues, but it is suitable only in closed environments and it shows the lowest bitrate.

VLC technology is still in the introductory phase and substantial efforts are needed before it can be widely deployed for practical applications. Nevertheless a number of LED based applications are expected to be ready in few years in many sectors: from inner satellite to military purpose, from hospitals (where electromagnetic interference must be avoided) to aircraft, from lighting to automobiles. A short list follows.

Aviation: Radio waves cannot be used by passengers in aircraft. LED-based lights are already used in aircraft cabins and each of these lights could be potentials VLC transmitters to provide both illumination and media services for passengers. Furthermore, this will reduce the aircraft construction costs and its weight.

Smart Lighting: Smart buildings require aesthetic lighting. Smart lighting with VLC provides the infrastructure for both lighting and communication and reduces the circuitry and energy consumption within an edifice.

Hazardous Environments: In environments such as petrochemical plants, mines, etc., RF is potentially dangerous because there are explosion risks, so communication becomes difficult. VLC can be used in this area as it is a safe technology and provides illumination and communication at the same time.

Device Connectivity: By directing a visible light at a device one can have a very high speed data link and security because a beam of light is shined in a controlled way.

Defense and Security: VLC can enable secure and high data rate wireless communications within military vehicles and aircraft.

Hospitals: In hospitals, some equipment is prone to interference with radio waves, so using VLC has many advantages in this area.

Underwater Communications: VLC can support high data rates beneath the water, where other wireless technologies like RF do not work. Thus, communications between divers or remote operated vehicles are possible.

Vehicle and Transportation: Traffic lights and many cars use LED-based lights. Cars can communicate with each other to prevent accidents and also traffic lights can communicate with the car to ensure road safety. The role that VLC can play in this field will be deepened in the next section.

2.3 ITS Scenario

The most recent achievements of the activities in the area of intelligent transportation systems (ITS) promoted by academia, industrial stakeholders and Standard Development Organizations (SDO) are the so called Cooperative ITS (C-ITS) [15]. Their goal is to use and plan communication and sensor infrastructure to increase road safety. Communication cooperation on the road includes car-to-car, car-to-infrastructure, and vice versa. Data available from vehicles and road side units can be either consumed locally in the boundary of a geolocalized network or transmitted to a server for central fusion and processing. These data can be used to detect events such as road works, traffic jam, approaching emergency vehicle, etc. Such data are processed in order to produce driving recommendation dedicated to a single or a specific group of drivers and transmitted wirelessly to vehicles.

The development of the C-ITS has been driven by usage scenarios that see a great extent the use of radio wave technology, as depicted in Fig. 2.4. In this broad and more general framework we can identify a number of use cases where the unique characteristics of VLC devices can be exploited in more effective ways as compared to traditional radio wave technology. However, it is worth emphasizing that compared to the mature RF based technology, VLC is still in the introductory phase and substantial efforts are needed before it can be widely deployed for short-range ITSs applications.

It is worth to mentioning that before the onset of LED based VLC, a series of infrared (IR) optical devices have been used successfully in several ITS projects in Korea and Malaysia (electronic toll collection), Japan (Vehicle Information and Communication System [VICS]), and Germany (Truck Tolling Scheme), among others [11]. The lesson learned by those projects suggested that near IR can be used for broadcasting messages in line of sight from RSUs to vehicles and for receiving beacon frame from vehicles to RSUs, while far-infrared can be used for video surveillance [23]. Over 50 thousands IR VICS transceivers are already installed on the surface road in Japan and most of them are connected to Traffic Management Center. The maximum range of infrared VICS beacons is up to 10 m with the maximum data rate of 1 Mbps and the maximum packet data size of 59 bytes [18].

Both IR as well as VLC are used to be included in the so called ITS Infrastructureless Technologies. This category holds the technologies that do not require any traditional telecommunication infrastructure to operate [11]. Indeed, unlike cellular communications that rely on base stations and a large number of antennas deployed throughout a territory, the Infrastructureless Technologies are easy to install adapting the existing roadside settlement and they become ready to use within short time. About this, VLC is even cheaper than IR, because the light signal transmitters are

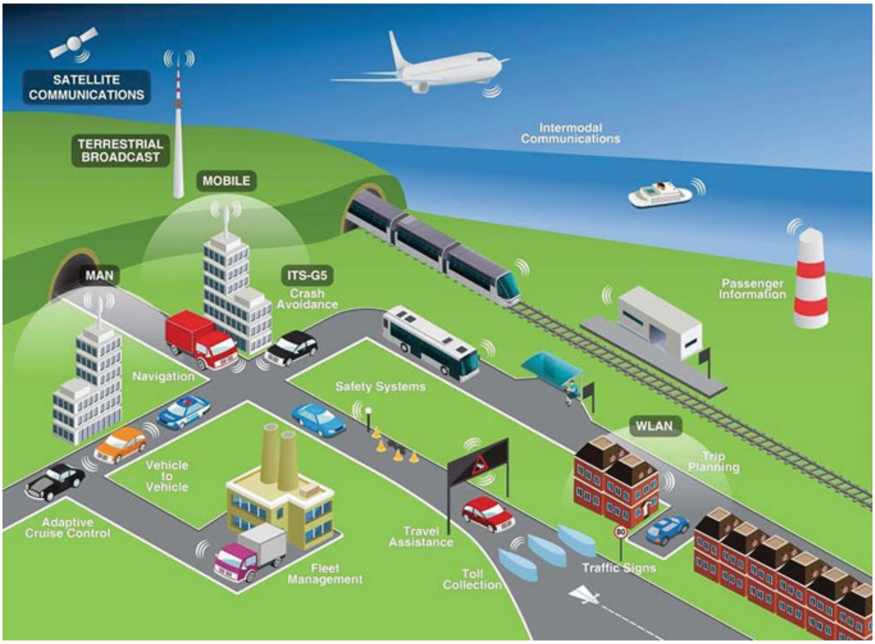


Fig. 2.4 ITS scenario [14]

the existing LED traffic lights and LED car headlamps, therefore if VLC is adopted in ITS applications instead of IR, there is no need to put ad hoc IR emitters in the environment.

2.3.1 VLC in ITS

Traffic signals and vehicles are gradually changing from electric light bulbs to LED lights because of their merits of energy saving, long life, low maintenance cost, better visibility and low temperature generation. These new lights have the potential to be used as transmitters of information, with signals transmitted by infrastructure lights and detected by receivers mounted in vehicles (I2V communications). RSUs such as LED-based traffic lights are well suited for information broadcast in vehicular communication systems in I2V mode. Traffic safety related information can be continuously broadcasted without extra power usage, enhancing smooth traffic flow as well as reducing accidents and fatalities. Since light goes straight on, high directional communication is possible, for instance different information can be transmitted for every lane of a road. It is also possible for cars to exchange data with adjacent vehicles (V2V communication), using head, tail and brake lights; in a V2V scenario example, a vehicle in front of a traffic light receives the information and relays it using the

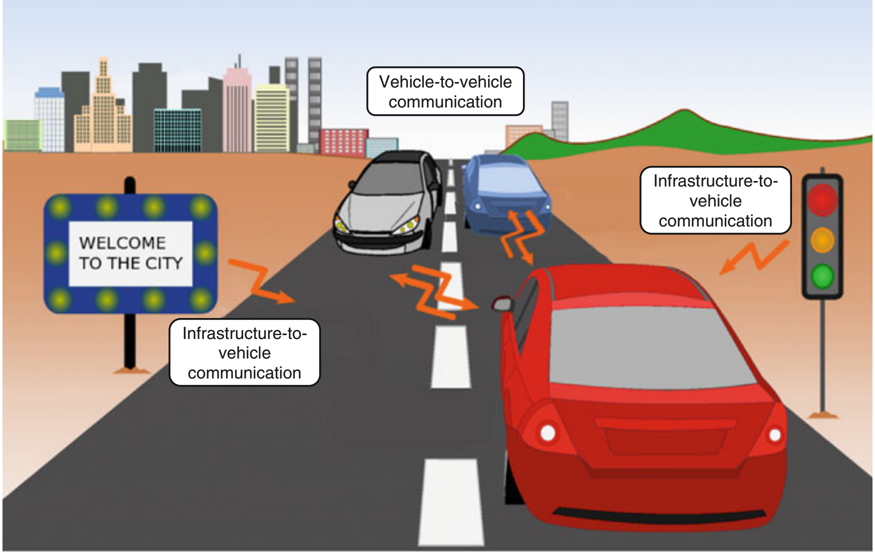

Vehicle-to-vehicle communication
WELCOME TO THE CITY
Infrastructure-to-vehicle communication
Infrastructure-to-vehicle communication

Fig. 2.5 Example of vehicular communication enabled by VLC

brake lights to the vehicle running behind. From the perspective of the vehicular ad-hoc network, VLC can be seen as a new access channel next to the RF existing ones. As shown in Fig. 2.5, potential applications of V2X systems are the same as those for RF channels, including active road safety, traffic efficiency, local services and Internet based services. Obviously the latency and reachability constraints for data exchanges are tighter for safety critical applications respect to the other kinds, but recent studies showed that they can be fitted by VLC also with off-the-shelf components [22].

Outdoor VLC links depend on the existence of an uninterrupted line of sight (LOS), so high density cars scenarios would see an increment in the number of links between vehicles, which could improve data delivery since multiple paths become available as more vehicles gets connected together with optical links. Conversely, in such a situation, R/F communications are likely to get into performance issues due to broadcast storms, disrupting real-time safety-critical applications and information dissemination [2].

Of course, outdoor mobile optical networks pose some technical issues and challenges with respect to indoor VLC: (i) relative mobility between vehicles or between infrastructure and vehicles is likely to disrupt LOS links; (ii) outdoor VLC is largely affected by natural and artificial lights, mainly sun light, which adds noise and interference to the received signal. The first problem could be addressed by optimizing fixed and mobile (on vehicle) lighting positioning, while interference may be minimized by using optical filters and optimized electronics. Anyhow, these problems pose an effective limit on the communication range; a number of experimental results and simulations showed that a reliable communication is possible when a VLC transmitter and a VLC receiver are no farther than 40–50 m [22].

A large number of analytical studies based on numerical simulations have been performed investigating the benefit of using VLC complementing RF DSCR messages to enable ITS related technologies, such as platooning [1], Cooperative Adaptive Cruise Control (CACC) [31] and Advanced Driver Assistance System (ADAS) [19], among others. Besides that strong efforts have been devoted for developing and testing experimental prototypes, as will be deepened in the next section.

2.4 Research and Prototypes for VLC in ITS

As soon as the high power LED technology was beginning to emerge, one of the first patented applications of VLC was in the scope on ITS. Indeed, utilization of LED traffic lights to transmit information have been patented very earlier in USA [16], even before the research was developed in that field. Afterwards, at the beginning of 2000s, many research group started to investigate this new technology and a number of prototypes was developed extending the field of application from I2V to V2V communication. At the same time, the interest in VLC grew and other field of applications have been explored. That motivated the set up of a special IEEE working group for the standardization of VLC for personal area networks, which at the end of 2011 released the first official version of the standard IEEE 802.15.7 [17].

Although none VLC commercial technology arrived in the domain of ITS so far, a series of prototypes exists that can be roughly classified in two classes: those resulting from research started before the publication of the IEEE 802.15.7 standard and those made after the IEEE 802.15.7 standard, trying to demonstrate how it can be exploited for ITS applications. This section contains the most significant results achieved by the researchers before the standard, while in the next sections the standard and its latest attempts of application in ITS will be introduced.

Starting from I2V communication, the basic performance of a LED based traffic light in terms of suitable modulation, required SNR, and the amount of receivable information was initially analyzed by Akanegawa et al. [3]. More recently a prototype of LED traffic lights was designed using discrete components for the opto-electronic parts and FPGA digital circuits for the signal processing parts for both the transmitter and the receiver [20, 21]. The modulation scheme used for this system was based on direct sequence spread spectrum techniques. The test trials on the prototype showed that the bottleneck for the data rate was the transmitter, where a data rate of 200 kbps was achieved. On the other hand the receiver was able to sample the signal 5-times higher at a maximum of 1 Mbps. The limitation of data rate occurs at the transmitter side because the traffic light needs a high amperage current load, but switching on and off such a current at high frequency is still challenging.

A system for transmitting messages from traffic light to cars was prototyped also by researchers of the University of Versailles [8]. The main application for the system that they patented [4] is the communication between the traffic lights and the car in

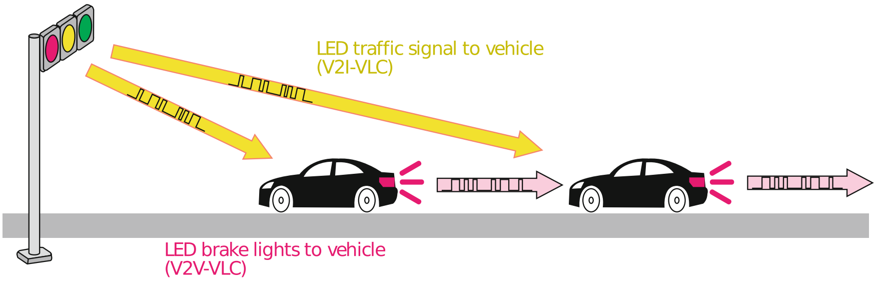


Fig. 2.6 VLC enhanced traffic light use case

order to transmit, for example, the countdown before the next traffic light signal change, as depicted in Fig. 2.6. The interest is also to alert the vehicles and to control the engine for a fast restart or for trigger the green wave.

The system consists of a broadcast station unit represented by a LED-based traffic light and a photodiode based receiver. Both emitter and receiver are interfaced with PCs. The emitter module was developed based on a commercial LED-based traffic light in order to investigate up until what point any traffic light can become a data broadcast unit with little modifications and at the lower cost. The same low cost constraints were used also to design the circuits and choising the electronic components for both the emitter and the receiver sides. Going more into details, the logic is implemented with 8 bit microcontrollers at both the transmitter and the receiver side. The receiver uses a PIN photodiode whose signal is amplified with an Automatic Gain Control (AGC), in order to receive data for both short and long distance. AGC is especially useful at short distance because of it prevents the saturation of the photodetector module. An optical component in front of the photodiode reduces the FOV angle of the receiver to $\pm 10°$. The system is robust, but non suitable to implement complex network stack. Indeed basic modulation coding schemes are used, such as the Manchester code and the Miller code. Both the codes use OOK (On-Off Keying) amplitude modulation which is simple and well suited for data transmissions at frequencies of tens of kilohertz.

The experimental tests have been conducted with a traffic light (red and green lights) installed in the corridor of the laboratory or outdoor. Basically, the message transmitted during the experiments is sent to the emitter and the frame indicates if Miller or Manchester code is selected. The receiver decodes the data in real-time and an algorithm allows post-processing or calculation of errors. For first experiments, a specific message made by 7 ACSII characters is sent continuously using a modulation frequency of 15 kHz. Error free trasmission (BER 10^{-7}) has been detected up to 50 m Outdoor with daylight and up to 20 m indoor with artificial light.

The use of VLC for V2V communication has been investigated by many research groups: it is believed that VLC could provide both accurate positioning and enforce vehicle safety. The architecture of such devices has been depicted with different degrees of detalis in many studies. Also, many numerical simulation have been performed in order to estimate the performance of that systems compared with RF DSCR. However a much less number of prototypes have been realized and tested, showing altought promising features of this technology. As we will see, some of them uses low cost components, while some others uses more sofisticated devices such as high speed cameras and dedicated CMOS imaging sensors.

An interesting example of low cost devices is the prototyping effort to implement a VLC system on scooters, using commercially available LED taillights and software defined radios (SDRs) [31].

Figure 2.7 shows the system block diagram of the prototype. On the transmitting end, the electronic control unit (ECU) connector periodically collects information

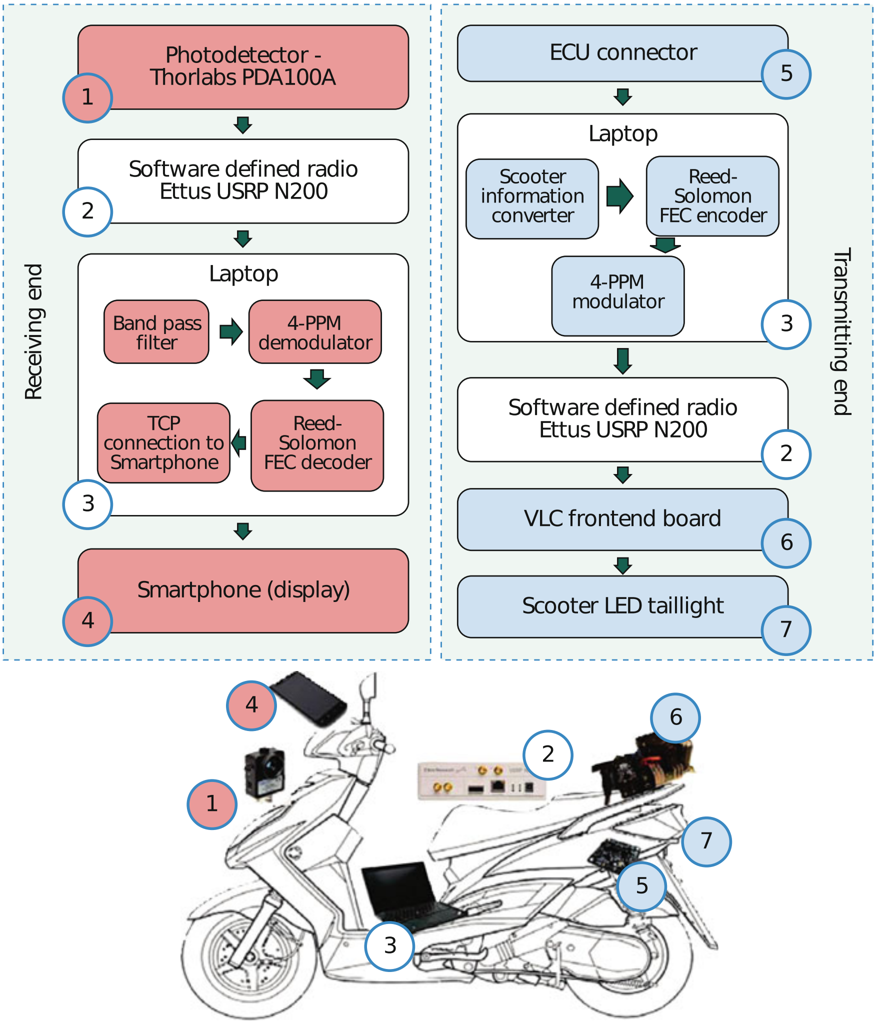


Fig. 2.7 Functional architecture of the VLC prototype implemented on scooters

such as current speed, engine revolution, brake status, and turning signal status from the scooter, and sends it to a laptop. In the laptop, digital packets with the information as the payload and a footer with forward error correction (FEC) code is created, and then modulated with 4-pulse position modulation (4-PPM). The packet goes through a digital-to-analog conversion in a SDR and is passed to the VLC frontend, which changes the light intensity of the LED taillight according to the input analog signal. On the receiving end, the photodetector converts the received optical signal to an electric signal, which goes through an analog-to-digital conversion in SDR. The laptop then performs the demodulation and decode processes to obtain the scooter information in the original packet. Finally, the information is sent to a smartphone mounted on the handlebar of the scooter. The smartphone presents a warning, and the information of a preceding scooter to the driver when a collision with this scooter is possible.

The choice to use the SDR is motivated because it allows the flexibility to easily change various physical layer designs and networking protocols during prototyping. Afterwards when the design of the devices will be focused on a commercial product, these designs can be transferred to a field programmable gate array (FPGA) or an application-specific integrated circuit (ASIC) to reduce costs.

The road trials have been performed with a one-way link, implemented between two scooters in such a way that the taillight of the scooter ahead transmitted data to a photodetector placed on the front of the chasing scooter. None lens was used in front of the photodetector, resulting in a wide 90° field-of-view (FOV) angle, while on the transmitter side the beam angle was more narrow (about 20°).

The experiments were carried out on a sunny day with no rain or fog. Both scooters operated in a real-world road at speeds between 10 and 40 km/h. A 20-byte packets that contain information about the current speed, brake status, and so on were continuously broadcasted from the taillight at a data rate of 10 kb/s. The measured packet loss on the reception side showed an acceptable level when the distance between the scooters was in the range of about 4–14 m.

Moving from low-cost systems to more sophisticated systems, the development of image-sensor-based VLC for automotive applications shows unique characteristics that deserve to be described. It is an alternate approach trying to integrate a special receiver function into a conventional image sensor. This V2V communication system investigated at Toyota Central R&D Laboratories uses an LED transmitter capable of sending various data by 10 Mb/s optical signals and a camera receiver, which employs a special CMOS image sensor, i.e., an optical communication image sensor (OCI) [29].

As shown in Fig. 2.8 the OCI has an array of non-conventional pixel, so called communication pixel (CPx), which is specialized for high-speed optical signal reception. Additionally, it has an output circuit producing a "1-bit flag image," which only reacts to high-intensity light sources such as LEDs and thus facilitates the LED detection in outdoor environments. By using the OCI, the camera receiver has obtained 10 Mb/s optical signal reception and accurate LED detection capabilities in various experiments under real world driving conditions. The experiments were performed with

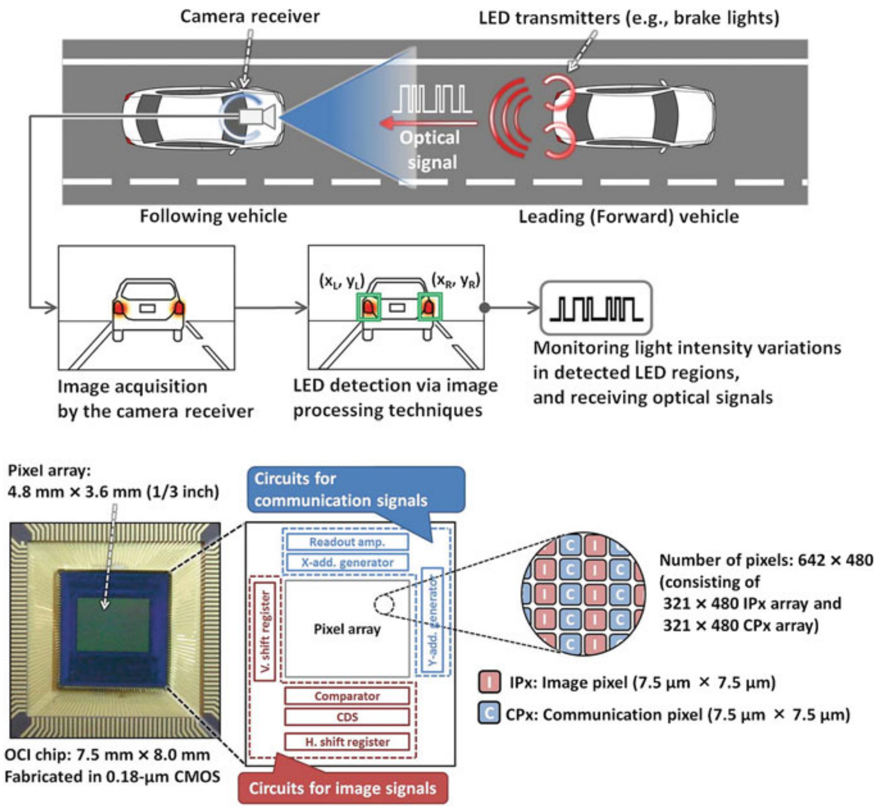

Camera receiver
LED transmitters (e.g., brake lights)
Optical signal
Following vehicle
Leading (Forward) vehicle
(x_L, y_L)
(x_R, y_R)
Image acquisition by the camera receiver
LED detection via image processing techniques
Monitoring light intensity variations in detected LED regions, and receiving optical signals
Pixel array: 4.8 mm × 3.6 mm (1/3 inch)
Circuits for communication signals
Readout amp.
X-add. generator
V. shift register
Pixel array
Y-add. generator
Comparator
CDS
H. shift register
Circuits for image signals
OCI chip: 7.5 mm × 8.0 mm
Fabricated in 0.18-μm CMOS
Number of pixels: 642 × 480 (consisting of 321 × 480 IPx array and 321 × 480 CPx array)
I IPx: Image pixel (7.5 μm × 7.5 μm)
C CPx: Communication pixel (7.5 μm × 7.5 μm)

Fig. 2.8 OCI based V2V communication system (up) and structure of the OCI (down)

the optical V2V communication system consisting of two LED transmitters mounted on a leading vehicle and one camera receiver mounted on a following vehicle.

The transmitter system consisted of an LED array unit controlled by a PC. The controller aggregates data caming from the vehicle and from the front camera, generates packets of 2464 bits, encodes the packets with Manchester, block interleave and BCH code for up to 3 error correction and send the data to the LED array unit. The latter has a driver circuit and 10 × 10 LEDs emitting up to 4 W of optical power. It is important to stress that the system did not use current automotive taillight LEDs, but 870-nm nearinfrared (NIR) LEDs capable of being modulated at high speed. Therefore the system is not yet a proper VLC device, because the comunication carrier was in the IR spectrum, although the results of this attempt will facilitate the extension of this technology in the visible.

For confirming the performances and potential of this system, many experiments have been conducted under real driving and outdoor lighting conditions. The test trials showed that the LED detection method using the flag image effectively eliminates most unnecessary objects in images and achieves correct and real-time LED detection even in challenging outdoor environments. In data transmission experiments, the leading veichle simultaneously sent both a set of data about the driving conditions of the veichle (such as vehicle ID and speed) and a stream of color image data (320 × 240 pixels) up to 20 fps. The measurements proved that the following vehicle received these data with acceptable packet loss, i.e. the front-view image stream is received with an efficiency rate of 87 % in the daytime and 89 % at nighttime.

2.5 Standardization

The achievement of VLC devices for ITS applications may be accelerated by standardization initiatives. There are currently two entities involved in VLC standardization: the Visible Light Communication Consortium (VLCC) in Japan and the IEEE 802.15 WPAN Task Group 7. The former proposed two standards at JEITA in 2007: one is Visible Light Communication System Standard (JEITA CP-1221), mainly focused on position detection applications, and the other is Visible Light ID System Standard (JEITA CP-1222), but they have not been commercially exploited.

The most important contribution came from the IEEE 802.15 WPAN Task Group 7, who released the first official VLC standard in the second half of 2011 [17]. This standard covers both the physical layer (PHY) air interface and the medium-access control (MAC). The IEEE 802.15.7 standard is significant for VLC community, because it represents the basis for developing products with guaranteed functionalities. It also provides a minimum benchmark for future developments. The standard intends to support a variety of expected applications, relating to VLC Personal Area Networks (VPAN).

As we can see in Table 2.1, three classes of devices are considered for VLC: infrastructure, mobile and vehicle. According to their physical properties and capabilities—limitations like physical mobility, power supply and of course their applications, their specifications such as range and data rates are defined. For instance infrastructure has "unlimited" power supply, while vehicle moderate and mobile terminals very limited. These yield higher power light sources for infrastructures and vehicles and furthermore potentially higher range. Regarding mobility, only

Table 2.1 Classification of IEEE 802.15.7 devices

	Infrastructure	Mobile	Vehicle
Fixed coordinator	Yes	No	No
Power supply	Ample	Limited	Moderate
Form factor	Uncostrained	Constrained	Uncostrained
Light source	Intense	Weak	Intense
Physical mobility	No	Yes	Yes
Range	Short/long	Short	Long
Data rates	High/low	High	Low

the infrastructure type has no physical mobility. Based in their applications vehicle devices need low data rates/long range for exchanging information about traffic for example, while mobile and infrastructure devices can reach much higher rates within shorter distance for exchanging multimedia like high definition videos, online gaming etc.

This standard defines a PHY and MAC layer for short-range optical wireless communications using visible light in optically transparent media. It is capable of delivering data rates sufficient to support audio and video multimedia services and also considers mobility of the visible link, compatibility with visible-light infrastructures, impairments due to noise and interference from sources like ambient light and a MAC layer that accommodates visible links. Furthermore, the standard adheres to applicable eye safety regulations.

The IEEE 802.15.7 standard supports three multiple access topologies: peer-to-peer, star configuration and broadcast mode; with data rates ranging from 11.67 kb/s to 96 Mb/s for indoor and outdoor applications. The architecture of a generic VLC standard conform device, as shown in Fig. 2.9 besides the ISO OSI stack, is defined in terms of a number of layers and sublayers; each layer offers services to the higher layers.

A VPAN device comprises a physical layer (PHY), which contains the light emitter/receiver along with its low-level control mechanism, and a medium access control (MAC) sublayer that provides access to the physical channel for all types of transfers. A logical link control (LLC) layer can access the MAC sublayer through the service-specific convergence sublayer (SSCS). A device management entity (DME) is also supported in the architecture. The DME can talk to the PLME (Physical Layer Management Entity) and MLME (MAC Link Management Entity) for the purposes of

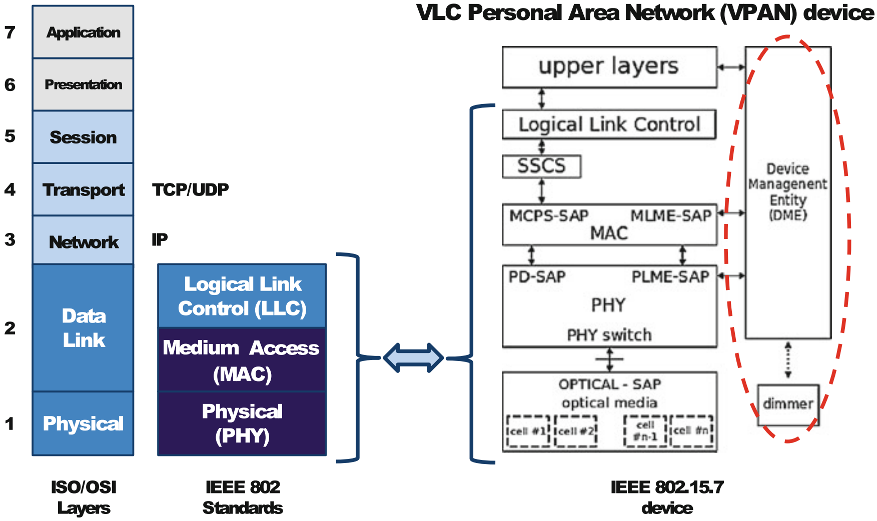


Fig. 2.9 IEEE 802.15.7 VPAN device architecture

interfacing the MAC and PHY with a dimmer. The DME can access certain dimmer related attributes from the MLME and PLME in order to provide dimming information to the MAC and PHY. The DME can also control the PHY switch using the PLME for selection of the optical sources and photodetectors.

The MAC layer handles physical layer management issues such as addressing, collision avoidance and data acknowledgment protocol. Many features of the MAC sublayer are shared with the IEEE 802.15.4 specifications, such as beacon management, channel access, guaranteed time slot (GTS) management, frame validation, acknowledged frame delivery, association, and disassociation. However some functions are peculiar of the visible light medium such as visibility, flicker-mitigation and dimming support.

Visibility support is provided across all topologies to maintain the illumination function in the absence of communication or in the idle or receive modes of operation. The purpose of this function is to maintain illumination and mitigate flickering.

The physical layer is divided into three types; PHY I, II & III, and these employ a combination of different modulation schemes.

PHY I operates from 11.67 to 266.6 kb/s, PHY II operates from 1.25 to 96 Mb/s and PHY III operates between 12 and 96 Mb/s. PHY I and PHY II are defined for a single light source, and they support on-off keying (OOK) and variable pulse-position modulation (VPPM). PHY III uses multiple optical sources with different frequencies (colors) and uses a particular modulation format called color shift keying (CSK).

Each PHY mode contains mechanisms for modulating the light source, run length limited (RLL) line coding, and channel coding for forward error correction (FEC).

RLL line codes are used to avoid long runs of 1 and 0 s that could potentially cause flicker and clock and data recovery (CDR) detection problems. RLL line codes take in random data symbols at input and guarantee DC balance with equal 1 and 0 s at the output for every symbol. Various RLL line codes such as Manchester, 4B6B, and 8B10B are defined in the standard, and provide tradeoffs between coding overhead and ease of implementation.

For ITS application the PHY I type is the most convenient, since it is designed specifically for outdoor applications. Although it provide the slowest data rates, robust convolutional and Reed-Salomon codes are used for forward error correction to overcome the additional path loss due to longer distance and potential interference introduced by optical noise sources such as daylight and fluorescent lighting.

PHY I modulation mode are two: on-off keying (OOK) and variable pulse-position modulation (VPPM). Each one has an associated optical clock rate which is "divided down" by the various coding schemes to obtain the final resulting data rates, as shown in Table 2.2.

The optical clock rate for PHY I is chosen to be no higher than 400 kHz to account for the fact that LEDs used in applications such as traffic lights require high currents to drive the LEDs and therefore switch slowly.

With OOK, as the name suggests, the data is conveyed by turning the LED off and on. In its simplest form a digital '1' is represented by the light 'on' state and a digital '0' is represented by the light 'off' state. At the slowest optical clock, the

Table 2.2 PHY I operating modes

Modulation	RLL code	Optical clock rate (kHz)	FEC		Data rate (kb/s)
			Outer code (RS)	Inner code (CC)	
OOK	Manchester	200	(15,7)	1/4	11.67
			(15,11)	1/3	24.44
			(15,11)	2/3	48.89
			(15,11)	None	73.3
			None	None	100
VPPM	4B6B	400	(15,2)	None	35.56
			(15,4)	None	71.11
			(15,7)	None	124.4
			None	None	266.6

802.15.7 standard uses Manchester Coding to ensure the period of positive pulses is the same as the negative ones but this also doubles the bandwidth required for OOK transmission.

Dimming is supported by adding an OOK extension which adjusts the aggregate output to the correct level. Light dimming is defined as controlling the perceived brightness of the light source according to the user's requirement and is a cross layer function between the PHY and the MAC. An idle pattern can be transmitted during MAC idle or RX states from infrastructure light sources for dimming support. This is important since it is desired to maintain visibility during idle or RX periods at the infrastructure. The idle pattern has the same duty cycle that is used during the active data communication so that there is no flicker seen during idle periods. The standard also supports dimming during data transmission: for instance, dimmed OOK modulation breaks the frame into subframes, inserting compensation symbols before each subframe in order to increase or reduce perceived brightness.

At bit rates higher than 100 kbit/s IEEE 802.15.7 prescribes for the PHY I layer the VPPM modulation scheme. Pulse position modulation (PPM) encodes the data using the position of the pulse within a set time period. The duration of the period containing the pulse must be long enough to allow different positions to be identified, e.g. when the position are two (2-PPM) a '0' is represented by a positive pulse at the beginning of the period followed by a negative pulse, and a '1' is represented by a negative pulse at the beginning of the period followed by a positive pulse. VPPM is similar to 2-PPM but it was tailored in such a way as the pulse width is controlled from the light dimming support: pulse amplitude is kept constant, while pulse width varies according to the desired dimming level. Figure 2.10 shows examples of two "0" and one "1" with different dimming levels (T is the symbol period).

The transmitter chain of the VPPM mode sees the input data sent through an RS FEC encoder for error protection, followed by a 4B6B RLL code for DC balance and flicker mitigation. The 4B6B coding takes a 4-bit symbol and changes it into a DC

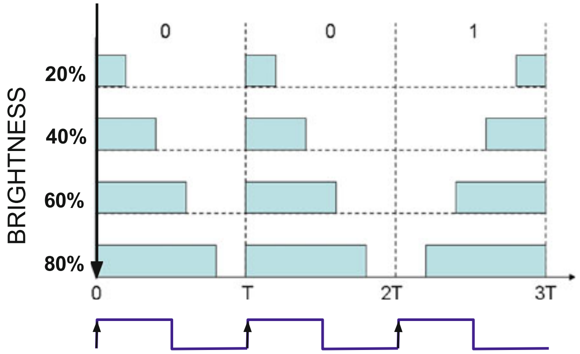


Fig. 2.10 Variable pulse position modulation

balanced 6-bit code, according to a tabular scheme. The counts of 1 and 0 in every VPPM encoded symbol are always equal to 3. Since the bit rate is constant regardless of the requested dimming level, as the light is dimmed, the range decreases with the dimming level.

So far, none of the proposed solutions for using VLC in ITS are actually compliant with the IEEE 802.15.7 standard. The main reason lies in the fact that the standard is subsequent to the majority of the studies carried out about VLC for ITS.

In the next section, the analysis of a IEEE 802.15.7 conform VLC prototype for V2X message delivery is presented.

2.6 VLC IEEE 802.15.7 System for ITS Applications

This section shows the features of the IEEE 802.15.7 conform VLC prototype developed at CNIT as enhancing extension of an embedded ITS station. The aim of this section is to provide some guidelines to develop a low-cost VLC system using commercial off-the-shelf (COTS) devices suitable for ITS applications.

2.6.1 VLC Prototype Design

The SEED-EYE board was used as developing board: it is a Wireless Sensor Network (WSN) node in house developed for multimedia ITS services [28]. It hosts the high performance Microchip PIC32, and has a full set of communication interfaces such as Ethernet, IEEE 802.15.4/ZigBee, and USB. The computational resources of the micro-controller are also devoted to process the images from a low cost CMOS Camera, making it an unique WSN node with high efficient image processing capabilities, tailored for ITS applications such as parking slot detection, traffic flow monitoring, etc. [26]. The SEED-EYE Board comes with full software support, including an

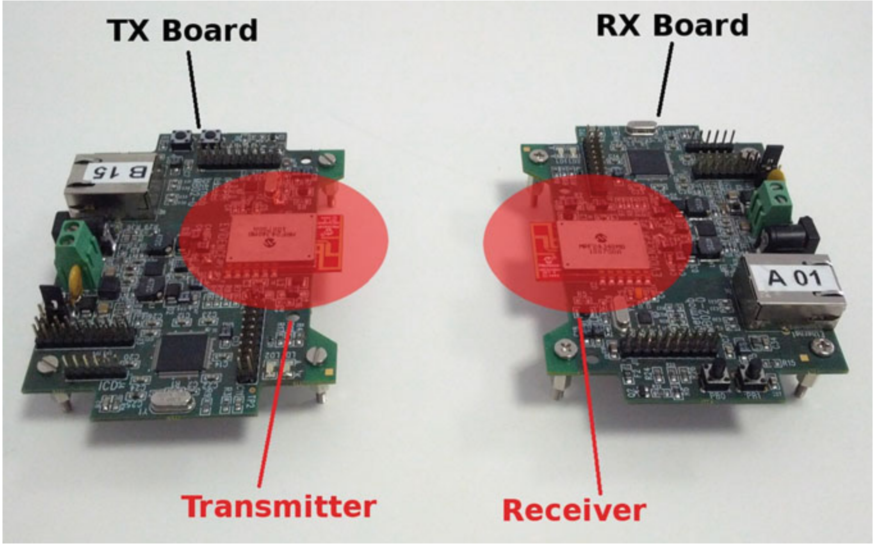


Fig. 2.11 The R/F IEEE 802.15.4 unidirectional system

open source OSEK/VDX Real Time OS for small micro-controllers for automotive applications, (ERIKA Enterprise Real Time OS [13]). A picture of that reference system is showed in Fig. 2.11.

This board was chosen for three main reasons: (1) it is made by low-cost off-the-shelf components, as the target is to develop scalable and pervasive systems capable to cover a large ratio of the sensitive environment, with large market penetration rates, eligible to be integrated in more complex systems like ITS; (2) a fully-customized firmware for the IEEE 802.15.4 transceiver was available [25] and this was used as reference guide for implementing the IEEE 802.15.7 protocols, especially for what concerns the MAC layer; (3) although the IEEE 802.15.4 transceiver was not used in the present work, its functionality is kept on the board in order to enable, as next developments, a vertical handover between IEEE 802.15.7 VLC and IEEE 802.15.4 R/F technologies.

2.6.1.1 The Reference Model

Because of the strong novelty of the design, ASICs with VLC transceivers suitable for the SEED-EYE board were not available on the market at that time. To overcome this limitation and starting a process having the final goal of producing a VLC dedicated component to be included in next releases of the board, the VLC transceiver functions were implemented on software blocks running on an extended architecture which even uses twin boards.

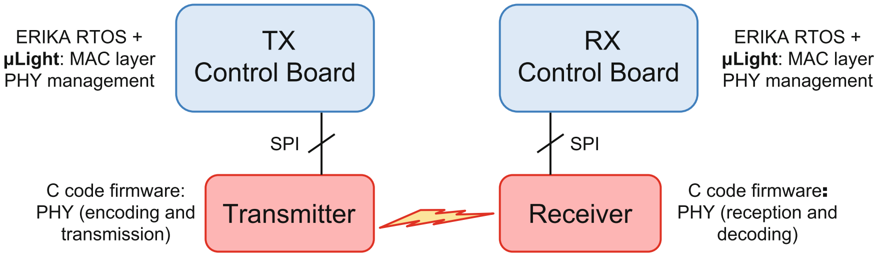


Fig. 2.12 The VLC IEEE 802.15.7 functions, as implemented in the system

This extended architecture is shown in Fig. 2.12: the MAC layer with the management PHY functions have been assigned to the control board, while PHY encoding/decoding and transmission tasks have been assigned to transmitter/receivers boards. In order to distinguish between boards with different tasks, the following terminology is used:

- TX/RX Control Board: a SEED-EYE implementing application-level tasks, MAC and PHY services;
- Transmitter/Receiver Board: a SEED-EYE implementing the optical devices; its tasks are data encoding/decoding and data transmission/reception over the visible light medium.

Control Board and Transmitter/Receiver Board share communication tasks via SPI interface. Figure 2.12 shows the functional blocks of the IEEE 802.15.7 half-duplex system, highlighting the implementation level of the various MAC and PHY services, either done as new library of the OS (called μLight), or as raw code. A picture of the complete system is reported in Fig. 2.13. In the next sections some details of the hardware end software design are deepened.

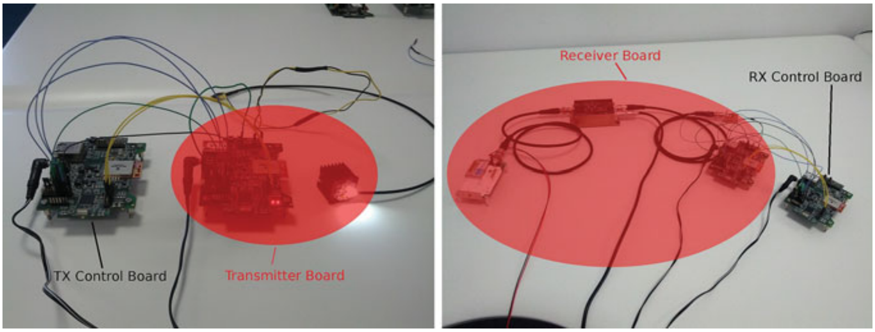


Fig. 2.13 Complete system: TX Control Board and Transmitter Board (*left*), RX Control Board and Receiver Board (*right*)

2.6.1.2 The Hardware Prototypes

The board used as basic brick of the system was designed within the IPERMOB project [30], targeted to a large-scale prototype deployed and tested on the land-side of the Pisa International Airport. The SEED-EYE board [28] is an advanced Wireless Sensor Network (WSN) node specifically thought for ITS applications [5]. It comes with full software support, including porting for Contiki OS [12] and ERIKA Enterprise RTOS [13], the latter of which was used in the present work. This device is equipped with an 80 MHz PIC32 micro-controller with built-in 128 KB of RAM and 512 KB of Flash ROM. It implements in hardware IrDA, SPI, I2C, UART, USB, and CAN communication protocols easing the connection with external units; the operative voltage of the chip ranges from 2.3 to 3.6 V and some power sleeping modes (RUN, IDLE, and SLEEP modes) are allowed, along with multiple switchable clock modes useful for the development of power saving policies. Moreover a CMOS Camera is embedded on the board, what makes this device suitable to implement next generation imaging WSN [10]. From the point of view of the network layer and radio communications, SEED-EYE embeds a Microchip MRF24J40B transceiver. This transceiver is IEEE 802.15.4 compliant and operates in the 2.4 GHz ISM unlicensed band. It has an extremely high coverage (up to 100 m in open space at max power) and it is highly configurable. The R/F communication interface was not used in this work, but its functionality is kept on the board in order to enable, as next developments, a vertical handover between IEEE 802.15.7 VLC and IEEE 802.15.4 R/F technologies.

2.6.1.3 Optical Components

On the transmitter side, only two components are needed: LED and optical lens. The LED was a commercially available phosphor-white OSTAR LED, commonly used as luminous source, generating a radiation flux with a divergence angle of around 120°; the optical lens right after the LED has the purpose to reduce the beam divergence at 18°. On the receiver side, many components were needed:

- a custom Avalanche Photo-Diode (APD) (Hamamatsu C 5331-11 [6]), with a tiny (1 mm^2) active area (the surface which can receive the light signal) and a frequency bandwidth ranging from 4 kHz to 100 MHz;
- an amplifier (FEMTO HVA-200M-40-B [6]), which receives the electrical signal from the APD and amplifies it by a factor of 10 or 100 (switchable gain 20 dB/40 dB);
- an adaptation circuitry composed by two standard avalanche fiberglass diodes BYW54 [6]; it is necessary because the output of the amplifier is a voltage falling in the range [−5 V, +5 V], but the inputs of the SEED-EYE generally require a voltage between −0.3 and 3.6 V (some pins are 5 V-tolerant). The adaptation circuitry cuts the negative part of the signal and reduces the maximum positive voltage;

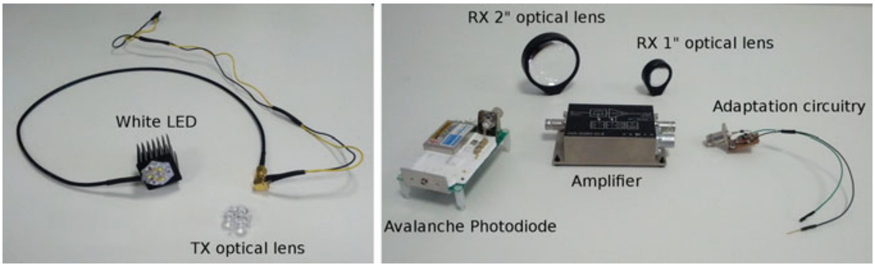


Fig. 2.14 Optical equipment: Transmitter (*left*), Receiver (*right*)

- two optical lenses: a Thorlabs LMR1/M, with a focal length of 1″ used for low-range tests, and a Thorlabs LMR2/M, with a focal length of 2″, used for medium-range tests ($\geq 10\,\text{m}$). They are used to increase communication distance by placing them in front of the APD and focusing the light on the active area of the detector.

In Fig. 2.14, all the components of the transmitter/receiver are illustrated.

2.6.1.4 The Software Prototypes

The software stack was developed using either the API (Application Programming Interface) of the ERIKA open source Real-Time Operative System (RTOS) for TX/RX Control Boards, or the MPLAB® Official Microchip Integrated Development Environment (IDE) for the VLC transmitter/receiver boards.

2.6.1.5 μLight Stack

Using the highly modular ERIKA API framework, and inspired by the experience in the IEEE 802.15.4 MAC and PHY implementation acquired during previous works [24], an IEEE 802.15.7-compliant network stack was programmed for the TX/RX Control Boards. The resulting software library, μLight, follows a layered approach, as shown in Fig. 2.15, conform to a VPAN device. To describe the details of software library is out of the scope of the present paper, below is presented only a brief outline. The Hardware Abstraction Layer is basically a wrapper for the functions of the transmitter/receiver driver (Optical TX/RX driver) that is beneath. With some extra functions added, it is responsible for keeping track of the transmitter/receiver state. The MAC and PHY layers accommodate a partial implementation of the IEEE 802.15.7 stack, on the one hand almost all the PHY I Service Access Point (SAP) primitives were implemented, on the other hand a minimal set of MAC SAP was implemented to allow easier and meaningful testing at this developing step. Over the MAC layer, μLight has a small high level library used for simple applications where IEEE 802.15.7 is required: to initialize the board as a VPAN coordinator and

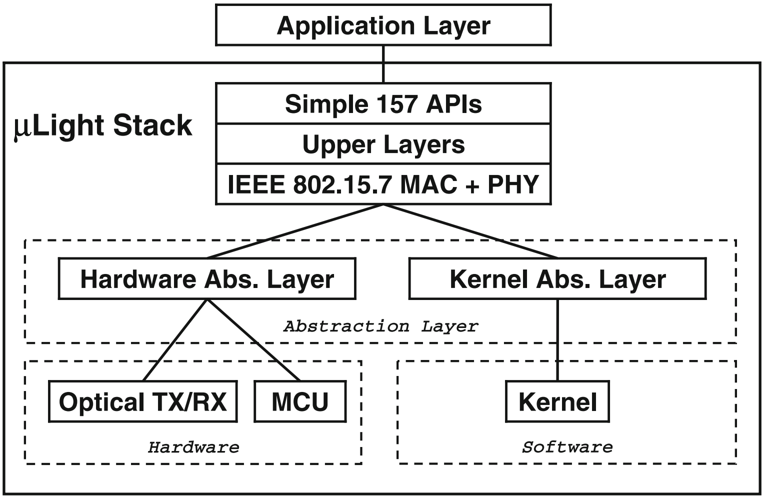


Fig. 2.15 μLight Architecture

starts a new VPAN, to initialize the board as a VPAN device and seeking for a coordinator, to set some function to be called when a frame is received from MAC layer, etc. Eventually a very simple Device Management Entity has been implemented, with the purpose of enabling and disabling the idle pattern dimming and setting a dimming level. The μLight software library required a driver to control the optical transmitter/receiver, so a new driver has also been added to ERIKA OS. This driver implements the Control Board side of the SPI protocol described in the next paragraph.

2.6.1.6 Optical Transmitter/Receiver

The transmitter/receiver was developed to perform three main tasks:

1. provide an interface for the Control Board to transfer data and configuring the transmitter/receiver itself (enable/disable transmission/reception, set data rate, etc.),
2. encode/decode data,
3. transmit/receive data within constrained timing.

The SPI peripheral is used for transmission and reception; both transmitter and receiver units are configured as SPI slaves. The optical transmitter/receiver is seen by the Control Board as a stack of addressable control registers and a stack of TX/RX data buffers. All control registers are 8 bit wide, though some of them are significant in pairs. They are addressable by a 6 bit address (short address). TX and RX buffers are 1025 bytes wide: the first two bytes hold the length of the data, while the other

1023 bytes hold the actual data. They are addressable by a 13 bit address (long address). The implemented communication protocol allows only the PHY I level of the standard to perform. This PHY type is intended for outdoor usage with low data rate applications, therefore it is suitable e.g. for devices deployed to roadside ITS Stations. For the current prototype only the OOK modulation format was used. Data transmission is done at a maximum data rate of 100 kbps, while the header is always sent at 11.67 kbps. The IEEE 802.15.7 standard prescribes some error correction techniques to maximize fail-safe communications in noisy environments. Thus, Reed Solomon encoding, Convolutional Codes, Manchester encoding and CRC-16 have been implemented on the Transmitter Board; similarly, Reed Solomon decoding, Viterbi decoder and Manchester decoding have been implemented on the Receiver Board. The codes are based on publicly available sources, and has been adapted and optimized for the specific needs.

2.6.2 VLC Prototype Performances

The proposed solution was experimentally investigated performing two kind of measurements: on the first the devices were characterized on the laboratory test bench in terms of processing times of the signal and measurement of the physical layer throughput. Then the system was arranged in a free-access and bright corridor of the laboratory building, in order to represent typical noisy outdoor conditions (e.g. V2I communication of a roadside ITS Station). In that condition Bit Error Rate (BER) measurements were performed.

2.6.2.1 Test Bench Measurements

Each task of the communication chain committed to the Transmitter/Receiver Boards has been characterized in terms of processing times. The source code of Transceiver Boards firmware has been compiled with MPLAB®XC32 Compiler v1.20, with the lowest level of optimization, because the freeware version was used. Time measurement are done by sampling the low-to-high and high-to-low transitions of a debug pin on the Transmitter/Receiver Boards; this debug pin switches whenever a task starts and finishes; results, shown in Table 2.3 are averaged over multiple repetitions of the same task. On the transmitter side, the signal delays due to the standard processing protocols are in the expected range, that is very close to the physical limit allowed by the prescribed clock frequency (200 kHz). On the receiver side, the data clearly shows that Viterbi algorithm is really slow. Also in the case of RS(15,7) blocks with correction of all errors, the convolutional decoding time is ten times slower than the others. For that cases the performance of the current system are so far from a satisfactory degree, as to indicate clearly the need of radically changing the architecture of some electronic component. For e.g. remaining on low cost readily available

Table 2.3 VLC transceiver processing times

Board	Task	Processing time (μs)
Transmitter	SPI oprical transmission ISR	2.6
	RS(15,7) block encoding	20
	RS(15,11) block encoding	16
Receiver	Viterbi single iteration	15
	Viterbi complete decoding[a]	0.27×10^6
	RS(15,7) block decoding without errors	32
	RS(15,7) block decoding with errors	72
	RS(15,7) block decoding with errors[b]	0.021×10^6
	RS(15,11) block decoding without errors	18
	RS(15,11) block decoding with errors	40

[a] 1023 B PSDU + RS(15,7)
[b] 1023 B PSDU

devices, all the VLC transceiver functions can be implemented on FPGA, where thanks to its strong parallel processing capabilities, multiple Reed Solomon blocks can be computed at the same time and Viterbi algorithm can be heavily parallelized.

Some kind of measurement of the PHY layer throughput was performed, to further assess the efficiency of the VLC device. Throughput tests were performed by sending and receiving multiple packets and measuring the time between the start of a packet transmission and the end of a packet reception. Time measurements account for the encoding, transmission, reception and decoding of a complete Presentation Protocol Data Unit (PPDU), which according to IEEE 802.15.7 is formed in turn by the series of Synchronization HeadeR (SHR), Physical layer HeadeR (PHR), and PHY Service Data Unit (PSDU). In these tests the two variables are the PSDU length and the data transmission rate. It is worth to note that the PHY I variable data rate applies to the PSDU only, since the SHR (which is 8 byte long in these tests) is transmitted at 200 kHz, and the PHR is always sent at 11.67 kbps. The results are shown in Fig. 2.16 where the throughput efficiency is plotted for different PPDU and data rate. The throughput efficiency is computed as the ratio between real throughput and reference throughput. Real throughput is computed with the formula: $\frac{PSDU\ length}{total\ transmission\ time}$, where "total transmission time" refers to the PPDU transmission time (SHR and PHR are thus considered overhead). The reference throughput is the maximum theoretical value achievable with an ideal ultra fast microchip, not introducing any delay respect to the nominal data rate of SHR, PHR and PSDU. Also in this case the worsening of the performances is evident in conditions where the Red Solomon and Viterbi algorithms are requested by the standard (from 11.67 to 48.89 kbps). On other hand,

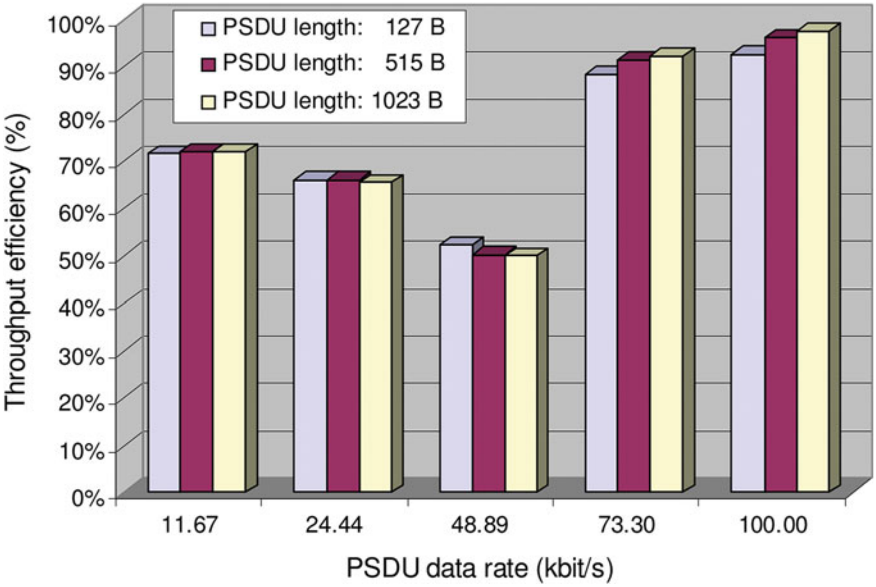


Fig. 2.16 Throughput efficiency of the VLC system

when convolutional codes are not requested by the standard as in the 73.3 kbps (where only RS(15.11) is working) and in the 100 kbps (none noise correction) cases, the real throughput is close to the ideal state.

2.6.2.2 BER Measurements

The VLC system was arranged in a free-access and bright corridor of the laboratory building, in order to test its communication performance in a noisy environment (see Fig. 2.17). The optical alignment between transmitter and receiver was made by hand, without pursuing high precision, because the goal of the test was to reproduce real life conditions.

Several tests were performed in order to find out on the first the maximum achievable distance between the TX and the RX, and then to measure the BER. The latter is the number of bit errors divided by the total number of transferred bits during a given time interval of transmission. Ten different system configuration were considered at each tested distance (0.5, 1.8, 2.8, 5.1 and 10.2 m), where the variables were the 5 implemented data rates and 2 packet sizes: small packet (127 B PSDU length) and large packet (950 B PSDU length). Figure 2.18 shows the measured BER for both the small and the long packet transmission scenario; an error-free communication was achieved up to 5.1 m, while communications at 10.2 m showed some errors. Different symbols refer to the different packet sizes. Comparing the two plots at 10.2 m a common BER degree can be found only at 100 kbps, when none error correction

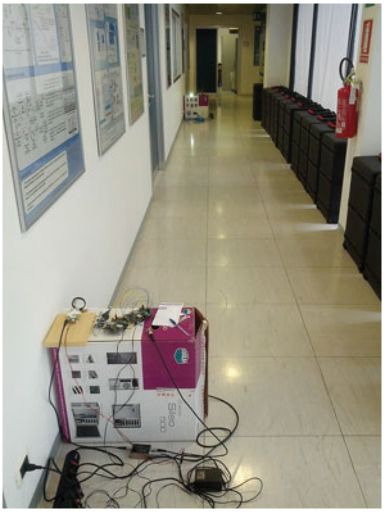


Fig. 2.17 Experimental setup for BER measurement

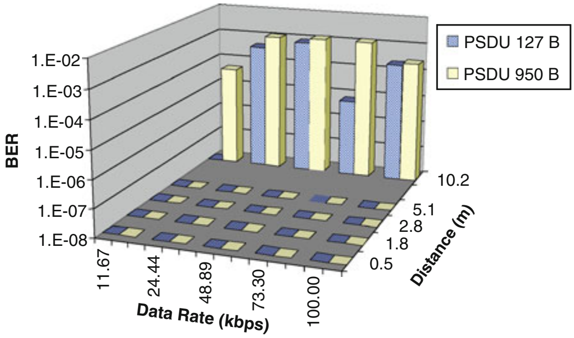


Fig. 2.18 BER at many data rate and distance values and two payloads: 127 B PSDU, 950 B PSDU

protocol are working; while at the other data rate, the BER scatters above and below 100 kbps value without a coherent behaviour respect to payload and data rate. This could be due to the accidental nature of noise generation combined to the ability of the protocols to recover some errors instead of others.

2.7 Conclusion

A simplex VLC prototype for ITS applications has been realized. The device implements PHY I and MAC layers such as conform to the IEEE 802.15.7 standard. The experimental characterization has shown that successful message delivery is very close to the reference case at highest bit rates, when convolutional codes are not used. Faster electronic devices are needed to handle in a suitable way the error correction protocols prescribed by IEEE 802.15.7 when the communication occurs at slow rates.

The quality of signal transmission is found to be acceptable within 10 m. It is influenced mainly by the optical alignment system, which was not particularly accurate during the trials. Photo-diodes with a larger active area or telescopic systems on the receiver can improve these performances. At now, referring to ITS domain, only I2I communications services are feasible with the current prototype. Improved equipment will allow to implement ITS V2I and V2V communication services via VLC.

Future work will be addressed to improve and increase the implementation of IEEE 802.15.7 functionalities in the system. It is expected to achieve great performance improvements, transferring the Optical Transmitter/Receiver functions on a FPGA. In fact dedicated HW architectures could overcome the signal delay limitations due to the processing time of the general purpose CPU used at present [9]. Moreover the design of an adaptation layer between IPv6 and IEEE 802.15.7 would be effective to allow the access of VLC technology to the Internet of Things infrastructure. After the functional verification of IEEE 802.15.7-conform VLC technologies, it could be interesting to evaluate the possibility to realize the vertical handover between R/F and VLC communication systems, in order to extend the range of application in providing cooperative ITS services. Beside that, many others specific applications could be designed in the broad field of cooperative intelligent transport systems, together with the promotion of standardization initiatives at ISO and ETSI working groups.

References

1. M.Y. Abualhoul et al., Enhancing the field of view limitation of Visible Light Communication-based platoon, in *2014 IEEE 6th International Symposium on Wireless Vehicular Communications (WiVeC)*, Sep 2014, pp. 1–5. doi:10.1109/WIVEC.2014.6953221
2. A. Agarwal, T.D.C. Little, Role of directional wireless communication in vehicular networks, in *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pp. 688–693. doi:10.1109/IVS.2010.5547954
3. M. Akanegawa, Y. Tanaka, M. Nakagawa, Basic study on traffic information system using LED traffic lights. IEEE Trans. Intell. Transp. Syst. **2**(4), 197–203. ISSN:1524-9050. doi:10.1109/6979.969365
4. Y. Alayli et al., Patent n° 09 58694. Communications par phares (2009)
5. D. Alessandrelli et al., ScanTraffic: smart camera network for traffic information collection, in *Proceedings of European Conference on Wireless Sensor Networks* 2012, pp. 196–211
6. A. Bell, Master's thesis. http://noes.sssup.it/images/theses/doc/thesisbe/lle.pdf.2013
7. D.A. Basnayaka, H. Haas, Hybrid RF and VLC systems: improving user data rate performance of VLC systems, in *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, May 2015, pp. 1–5. doi:10.1109/VTCSpring.2015.7145863
8. A. Cailean et al., A robust system for visible light communication, in *2013 IEEE 5th International Symposium on Wireless Vehicular Communications (WiVeC)*, June 2013, pp. 1–5. doi:10.1109/wivec.2013.6698223
9. F. Che et al., Design and implementation of IEEE 802.15.7 VLC PHY I transceiver, in *2014 12th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT)*, Oct 2014, pp. 1–4. doi:10.1109/ICSICT.2014.7021249
10. M. Chitnis et al., Distributed visual surveillance with resource constrained embedded systems, in *Visual Information Processing in Wireless Sensor Networks: Technology, Trends and Applications*, ed. by L. Ang, K. Seng (IGI Global, Pennsylvania, 2012), pp. 272–292

11. K. Dar et al., Wireless communication technologies for ITS applications [Topics in Automotive Networking]. IEEE Commun. Mag. **48**(5), 156–162 (2010). ISSN:0163-6804. doi:10.1109/MCOM.2010.5458377
12. A. Dunkels, B. Grnvall, T. Voigt, Contiki—a lightweight and flexible operating system for tiny networked sensors, in *Proceedings of the First IEEE Workshop on Embedded Networked Sensors (Emnets-I)*. Tampa, Florida, USA, Nov 2004
13. Erika Enterprise RTOS. http://erika.tuxfamily.org
14. European Telecommunications Standards Institute, Intelligent Transport Systems. http://www.etsi.org/ITS.2012
15. A. Festag, Cooperative intelligent transport systems standards in Europe. IEEE Commun. Mag. **52**(12), 166–172 (2014). ISSN:0163-6804. doi:10.1109/MCOM.2014.6979970
16. P.A. Hochstein, Traffic information system using light emitting diodes. US Patent 5,633,629, May 1997. http://www.google.com/patents/US5633629
17. IEEE Standard for Local and Metropolitan Area Networks-Part 15.7: Short-Range Wireless Optical Communication Using Visible Light, IEEE Std 802.15.7-2011, Sep 2011, pp. 1–309. doi:10.1109/IEEESTD.2011.6016195
18. M. Kobayashi, K. Suzuki, S. Nishimura, Utilization of probe data for traffic flow control, in *18th ITS World Congress* (2011)
19. N. Kumar, L.N. Alves, R.L. Aguiar, Visible light communication for advanced driver assistant systems, in *7th Conference on Telecommunications, Conftele 2009, Sta Maria da Feira—Portugal*, May 2009
20. N. Kumar, L. Nero Alves, R.L. Aguiar, Employing traffic lights as road side units for road safety information broadcast, in *Roadside Networks for Vehicular Communications: Architectures, Applications, and Test Fields*, ed. by R. Daher, A. Vinel (IGI Global, Hershey, 2013), pp. 118–135
21. N. Kumar et al., Visible light communication for intelligent transportation in road safety applications, in *2011 7th International Wireless Communications and Mobile Computing Conference (IWCMC)*, July 2011, pp. 1513–1518. doi:10.1109/IWCMC.2011.5982762
22. C.B. Liu, B. Sadeghi, E.W. Knightly, Enabling Vehicular Visible Light Communication (V2LC) Networks, in *Proceedings of the Eighth ACM International Workshop on Vehicular Inter-networking, VANET'11* (ACM, Las Vegas, Nevada, USA, 2011), pp. 41–50. ISBN:978-1-4503-0869-4. doi:10.1145/2030698.2030705. http://doi.acm.org/10.1145/2030698.2030705
23. M.M. Mahmod et al., Wireless strategies for future and emerging ITS applications, in *Proceedings of T5th World Congress ITS* (2008)
24. P. Pagano, R. Pelliccia, M. Petracca, M. Ghibaudi, On Board Unit hardware and software design for Vehicular Ad-hoc NETworks (VANET), in *Demo session at "The Fully Networked Car @ Geneva International Motor Show (FNC 2011)"* (2011)
25. P. Pagano et al., ERIKA and open-ZB: an implementation for real-time wireless networking, in *Proceedings of the 2009 ACM Symposium on Applied Computing. SAC'09* (ACM, Honolulu, Hawaii, 2009), pp. 1687–1688. ISBN:978-1-60558-166-8. doi:10.1145/1529282.1529661. http://doi.acm.org/10.1145/1529282.1529661
26. G. Pellerano et al., 6LoWPAN conform ITS-Station for non safetycritical services and applications, in *2013 The 13th International Conference on ITS Telecommunications (ITST 2013)*. Tampere, Finland, Oct 2013
27. M.B. Rahaim, A.M. Vegni, T.D.C. Little, A hybrid radio frequency and broadcast visible light communication system, in *2011 IEEE GLOBECOM Workshops (GC Wkshps)*, Dec 2011, pp. 792–796. doi:10.1109/GLOCOMW.2011.6162563
28. SEED-EYE: An Advanced Multimedia Wireless Sensor Network Node for ITS Applications. http://noes.sssup.it/index.php/hardware/seed-eye
29. I. Takai et al., Optical vehicle-to-vehicle communication system using LED transmitter and camera receiver. IEEE Photon. J. **6**(5), 1–14 (2014). ISSN:1943-0655. doi:10.1109/JPHOT.2014.2352620
30. The IPERMOB Project. http://www.ipermob.org
31. S.-H. Yu et al., Smart automotive lighting for vehicle safety. IEEE Commun. Mag. **51**(12), 50–59 (2013). ISSN: 0163-6804. doi:10.1109/MCOM.2013.6685757

Chapter 3
Deterministic Vehicular Communications Supported by the Roadside Infrastructure: A Case Study

Tiago Meireles, José Fonseca and Joaquim Ferreira

Abstract The development of wireless vehicular networks for cooperative Intelligent Transport Systems (ITS) opened the possibility of launching cooperative applications that can improve vehicle and road safety, passenger's comfort and efficiency of traffic management. These applications exhibit tight latency and throughput requirements, for example safety critical services such as the Emergency Electronic Brake Light require guaranteed maximum latencies lower than 100 ms, while most infotainment applications require QoS support and data rates higher than 1 Mbit/s. Current wireless communication standards such as IEEE 802.11-2012 amendment 6 and ETSI-G5 have some drawbacks in what concerns their medium access control technique, which is based in CSMA/CA, particularly for high speed and high density environments. To deal with such environments, an infrastructured based TDMA protocol, the Vehicular Flexible Time Triggered protocol was proposed. In this chapter several protocol parameters are quantified, taking in account realistic scenarios and current wireless communication standards, that are applicable in these environments. To deploy such an infrastructured based network in an entire motorway might be expensive, therefore a concept of safety zone is used whenever there is a part of the motorway that is covered by road side units that implement the protocol. A worst case approach is used in order to prove that the V-FTT protocol has a bounded delay in what concerns the time occurred between a safety event detection and the instant of time of its reception by all vehicles travelling in the safety zone. With the exception of the lowest bit rate (3 Mbps), the V-FTT protocol has a guaranteed bounded delay under the maximum allowed latency of the most common safety vehicle applications.

T. Meireles (✉)
FCEE, Universidade da Madeira (UMa), 9000-390 Edifício da UMa, Funchal, Portugal
e-mail: hipkin@uma.pt

J. Fonseca
Instituto de Telecomunicações, ESTGA - Universidade de Aveiro, Aveiro, Portugal
e-mail: jaf@ua.pt

J. Ferreira
Campus Universitário de Santiago, DETI - Universidade de Aveiro, Aveiro, Portugal
e-mail: jjcf@ua.pt

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_3

3.1 Introduction

Road safety has always been a concern in most developed countries, particularly with the growing number of vehicles and the deployment of several kilometres of high speed motorways. Adding to the traditional passive and active safety devices, the last few years have made possible the development of wireless vehicular networks for cooperative Intelligent Transport Systems (ITS) and opened the possibility of launching cooperative applications that can improve vehicle and road safety, passenger's comfort and efficiency of traffic management. In order to support such visionary scenarios, applications running in the vehicles are required to communicate with other applications in other vehicles or with applications deployed in the back office of the emergency services, road operators or public services. The mobile units of a vehicular network are the equivalent to nodes in a traditional wireless network, and can act as the source, destination or router of information. Besides the ad-hoc implementation of a network consisting of neighbouring vehicles joining up and establishing Vehicle-to-Vehicle (V2V) communication, there is also the possibility of a more traditional wireless network setup, with base stations along the roads in Vehicle-to-Infrastructure (V2I) communication that work as access points and manage the flow of information, as well as portals to external WANs. Devices operating inside vehicles are called On Board Units (OBUs), while devices operating on the side of the road are Road Side Units (RSUs), and have different requirements and modes of operation.

Safety, efficiency and comfort ITS applications exhibit tight latency and throughput requirements, for example safety critical services such as the Emergency Electronic Brake Light require guaranteed maximum latencies lower than 100 ms while most infotainment applications require QoS support and data rates higher than 1 Mbit/s. Besides latency and throughput, safety applications also require deterministic communications (real-time). For example, a vehicle involved in an accident should be granted timely access to the wireless medium to transmit warning messages, even in congested road scenarios.

Wireless communication standards were developed for that purpose, for example the IEEE 1609 family of standards for Wireless Access in Vehicular Environments (WAVE) which rely on the IEEE 802.11-2012 Amendment 6 [1], also known as 802.11p, and the equivalent European standard ETSI ITS G5 [7]. Their medium access control (MAC) layer adopts a carrier sense multiple access with collision avoidance (CSMA/CA), same as IEEE 802.11a, but with a new additional, non-IP, communication protocol, essentially low overhead, port mapping protocols, designed to be small, efficient and tailored to the simple, single-hop broadcast over capacity constrained radio frequency channels. Since the IEEE 802.11p medium access control is based on CSMA, collisions may occur indefinitely due to the non-determinism of the back-off mechanism. So, native IEEE 802.11p MAC alone does not support real-time communications. Nevertheless, the probability of collisions occurring may be reduced if the load of the network is kept low, which is difficult to guarantee in vehicular communications, or if some MAC protocol restricts and controls the medium access to provide a deterministic behaviour. Strict real-time behaviour and safety guarantees are typically difficult to attain in ad-hoc networks, but they are even harder to attain in high speed mobility scenarios, where the response time of distributed consensus algorithms, e.g. for cluster formation and leader election, may not be compatible with the dynamics of the system.

In some operational scenarios the IEEE 802.11p MAC may no longer be deterministic, possibly leading to unsafe situations. This calls for a reliable communication infrastructure with real-time, secure and safety properties, which is mandatory to support the detection of safety events and the dissemination of safety warnings. The design choices to implement a deterministic MAC protocol for wireless vehicular communications are to rely on the road side infrastructure (V2I) or to be based on ad-hoc networks (V2V), without road-side units support. Hybrid approaches are also possible. There are two complementary ways to secure determinism (real-time), of the roadside network necessary for RSUs' coordination: use a real-time network technology, usually at Layer 2, and employ resource reservation protocols to extend the guarantees to multiple networks and to higher layers. It is likely that users place more trust in a vehicle safety network that is managed by the roadside infrastructure. Adding to that, the motorway infrastructure can have a global vision of the motorway or at least parts of it.

A deterministic MAC protocol was presented in [15], the vehicular flexible time-triggered (V-FTT), which adopts a master multi-slave time division multiple access (TDMA), in which the road-side units act as masters to schedule the transmissions of the on-board units. In this work, that proposal is analysed by quantifying a possible infrastructure deployment in motorways, particularly defining the usual coverage range for each RSU and the spacing between RSUs. According to the coverage range it is determined the maximum number of vehicles that can travel simultaneously inside the coverage range of a RSU, in order to quantify a maximum size for the Synchronous OBU Window (SOW), which is where OBUs can send information to the roadside infrastructure without contending for the medium. The duration of the SOW influences the duration of the other contention free window of the V-FTT protocol, which is the Infrastructure Window. Several calculations are done, having in mind the worst case scenario of trying to serve a full populated motorway in the same Elementary Cycle, and therefore the maximum duration of the contention free windows of the V-FTT protocol are determined.

That worst case scenario is then studied by determining what can be the worst possible delay between a OBU detecting some safety event until all the other OBUs are effectively warned. The idea is to prove that the maximum delay between event detection and event dissemination to all vehicles is bounded. This maximum worst case delay is then compared to the maximum latency that the most common safety applications require, such as Electronic Emergency Brake Light (EEBL-maximum latency of 100 ms) and Post Crash Warning (maximum latency of 500 ms).

The chapter ends by repeating the worst delay analysis for a real scenario: the A5 motorway from Lisbon to Cascais, one of the busiest motorways in Portugal. A possible coverage of the most dangerous motorway spots is suggested, and calculations are made in order to determine which will be the worst case delay for this scenario.

3.1.1 V-FTT Protocol Overview

The V-FTT Protocol was presented in [15]. It is an infrastructure based protocol, which adopts a master multi-slave time division multiple access where the communication medium is divided into time windows where road side units and protocol compliant OBUs have specific time windows where they can transmit without medium contention. The V-FTT protocol timeline is cyclic and divided into elementary cycle (EC). Each EC has three windows:

- **Infrastructure Window (IW)**—based on the information received from the vehicle on board units (OBUs) and some cross-validation with its own sources, the Road Side Units (RSUs) build the schedule for OBU transmissions. For that purpose each RSU periodically broadcasts a Trigger Message (TM) containing all identifiers (t_{ID}) of the OBUs allowed to transmit safety messages in the next period of OBU transmission, named Synchronous OBU Window. Based on OBU information and cross- validation, RSUs identify safety events and send warnings to OBUs belonging to vehicles affected by those specific safety events (protocol enabled and others). The warning messages (WM) have variable duration, depending on the number of occurred events. Each RSU will therefore transmit its TM and WM in its respective RSU transmission slot. Since each RSU slot will have a fixed size, care must be taken in order to fairly distribute slot time to TM and WM. There is no medium contention during the IW.
- **Synchronous OBU Window (SOW)**—this is where OBUs have the opportunity to transmit information to RSUs (V2I) without medium contention. Each OBU will have a fixed size slot (SM) to transmit vehicle information (speed, acceleration, etc.) and any safety event (e.g. Electronic Emergency Brake Light—EEBL). The SOW duration is variable. Each OBU will have a maximum of one slot per SOW, in order to ensure a fair access to the medium by all OBUs.
- **Free Period (FP)**—In the free period a contention period is ensured, where non-enabled OBUs are able to transmit safety messages and RSUs and OBUs are able to transmit non-safety short messages. Enabled OBUs may also transmit safety messages but without any guarantees since they have to contend for the medium. A minimum size for the FP must be guaranteed in order to reserve a contention period in the Elementary Cycle.

Figure 3.1 presents the Elementary Cycle and the transmission windows it contains.

The IEEE802.11-2012 standard defines a synchronization interval for all vehicles to tune the control Channel (CCH) in order to receive safety messages. This is shown in Fig. 3.2. There are several modes of operation, the normal mode simply

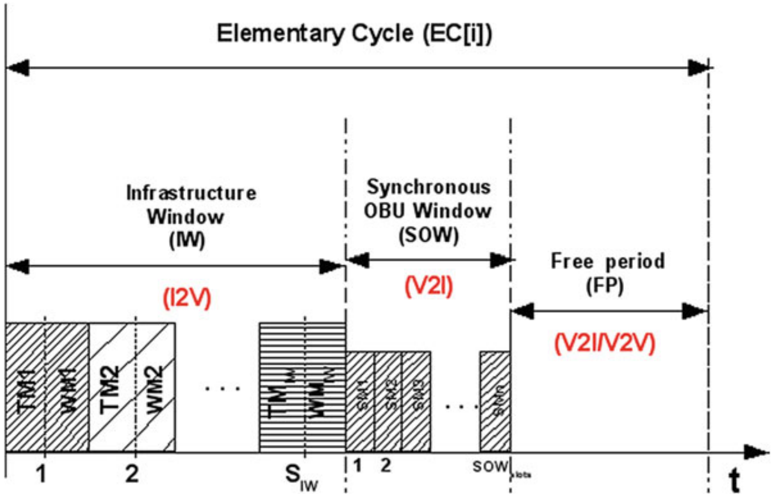


Fig. 3.1 Vehicular FTT (V-FTT) protocol

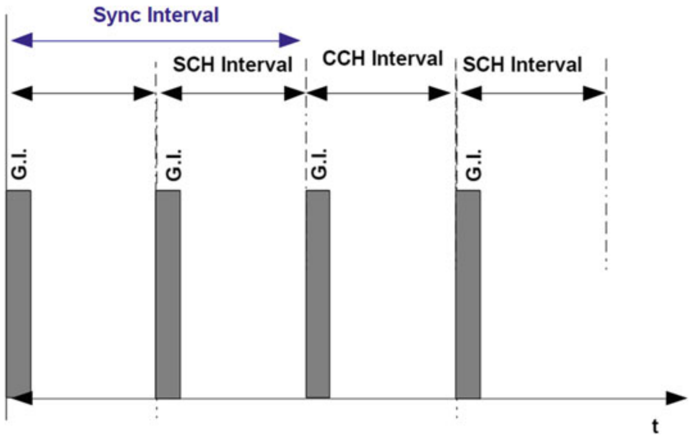


Fig. 3.2 IEEE 802.11p/WAVE synchronization interval (Adapted from [13])

switches from CCH to SCH every 50 ms, providing a guard interval (G.I.) of 4 ms to allow transceivers to do the switching, whereas the continuous mode a vehicle continuously monitors the CCH. This later situation occurs in the ETSI-G5 standard where all vehicles must have two radio devices, where one of them is always tuned to the Control channel in order not to miss any safety warnings.

In [15] a model for RSU deployment was proposed: RSUs are first installed in dense traffic areas (e.g. motorways near urban areas) and accident-prone zones such as dangerous curves or specific road sections such as tunnels or bridges. The road locations that have a record of a large number of crashes are also known as blackspots

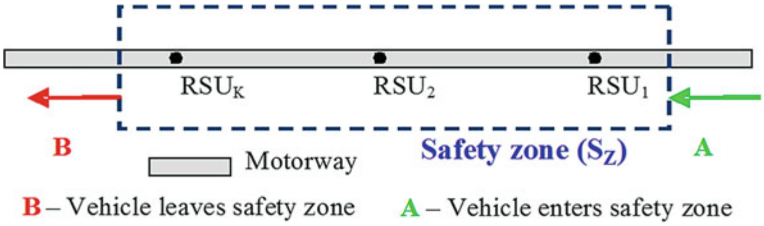

RSU_K
RSU_2
RSU_1
B
A
Motorway
Safety zone (S_Z)
B – Vehicle leaves safety zone
A – Vehicle enters safety zone

Fig. 3.3 Definition of Safety Zone (Sz)

and this term will be used from this point on. In order to be effective, each blackspot zone must have total RSU coverage. These specific and limited areas covered by RSUs are entitled Safety Zones (S_z), shown in Fig. 3.3.

The idea is that RSUs know exactly how many vehicles exist in the Safety Zone and will determine the instants where OBUs can communicate using the V-FTT protocol described earlier. For that purpose, OBUs register themselves in the motorway infrastructure or in each Safety Zone, receiving a temporary identifier (t_{ID}). Each time a vehicle exits the Safety Zone, its t_{ID} can be reused.

3.2 V-FTT Protocol Analysis

In this sub-section the several V-FTT protocol parameters that were presented in [15], will be quantified, for use with the IEEE802.11p standard or another similar wireless communication standard.

3.2.1 *Road Side Unit Coverage Area*

It is important to quantify the V-FTT protocol using real scenario characteristics. The quantification of the coverage radius (C_r) of a Road site Unit (RSU) influences the maximum number of vehicles served by an RSU (N_{VRSU}), which in turn influences the maximum sizes of the Infrastructure Window and the Synchronous OBU window. In order to define each RSU coverage area a compromise must be made between coverage area and terminal (vehicle) capacity. An IEEE802.11-2012/WAVE device was designed for maximum coverage range of 1000 m [8, 18], but tests proved that 750 m is a more realistic range [17], therefore C_r will be assumed to have a value of 750 m:

$$C_r = 750 \text{ m} \tag{3.1}$$

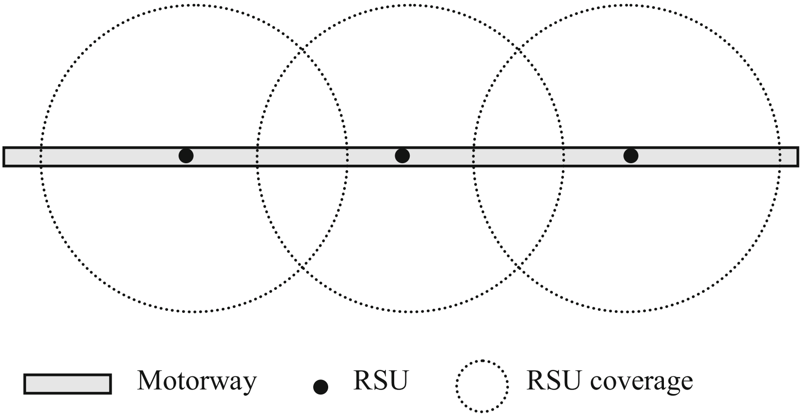


Fig. 3.4 RSU coverage

Several studies [2, 3, 14] defend that in WLANs the overlap of coverage area between Access Points should be between 15 and 25 % in order to ease the handover process (refer to Fig. 3.4). Since vehicular networks deal with high speed travelling mobile stations (vehicles) RSU coverage will be assumed to have 25 % of overlapping. This means that the overlapping range O_r is:

$$O_r = C_r \times 0.25 = 187.5\ \text{m} \tag{3.2}$$

The spacing between RSUs (S_r) will then be equal to (3.3):

$$\begin{aligned} S_r &= (2 \times C_r) - O_r \\ S_r &= (2 \times 750) - 187.5 = 1312.5\ \text{m} \end{aligned} \tag{3.3}$$

Takeing in account that motorways usually do not have curves with angles larger than 90°, considering an overlapping range of RSU of 25 % and assuming a linear distribution of RSUs, it is straightforward to conclude that an OBU can only hear a maximum of 2 RSU transmissions simultaneously. This means that $S_{IW} = 2$ (refer to Fig. 3.5).

RSUs assign each vehicle a temporary identifier, named t_{ID}. A size of 16 bit allows the identification of 65,536 distinct vehicles. Equation (3.4) defines the number of vehicles that can fit in the length of the Safety Zone (l_{S_z}).

$$n_{S_z} = \frac{l_{S_z}}{\left(V_{length} + v_{spacing}\right)} \times n_{lanes} \tag{3.4}$$

Using the maximum value of 65,536 vehicles and solving the equation above in order of the l_{S_z}, considering that vehicle average length (V_{length}) is 4.58 m [6],

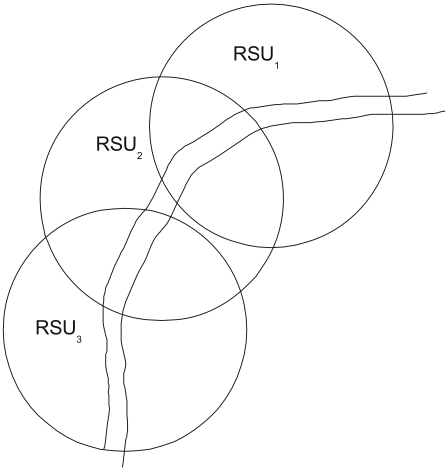


Fig. 3.5 Sketch of a motorway curve and RSUs coverage areas (25 % overlap)

Table 3.1 N_{VRSU}—Maximum number of vehicles covered by each RSU ($C_r = 750$ m)

N_{VRSU}	Normal traffic	Traffic jam
1 lane	44	103
2 lanes	87	206
3 lanes	130	309
4 lanes	174	412
5 lanes	217	507

vehicle spacing ($v_{spacing}$) is 10 m and that t_{ID} can be reused whenever a vehicle exits the Safety Zone, it means that this t_{ID} size allows to define a Safety Zone such as:

- a motorway with a maximum of 95 km with 5 lanes per travel path.
- a motorway with a maximum of 119 km with 4 lanes per travel path.

It is useful to find out what is the maximum number of vehicles that can be covered by a RSU. This will be named N_{VRSU} which can be determined by Eq. (3.5):

$$N_{VRSU} = \frac{2 \times C_r}{\left(V_{length} + v_{spacing}\right)} \times n_{lanes} \tag{3.5}$$

Considering an average vehicle length of 4.58 m [6], Table 3.1 shows several values for N_{VRSU}, where the spacing between vehicles ($v_{spacing}$) is 10 m for traffic jam and 30 m for normal traffic [19].

In the following sub-sections the maximum sizes for SOW and IW will be determined.

Table 3.2 Duration of a BSM in an OFDM 10 MHz channel

Bit rate (Mbps)	BSM (μs)	BSM + SIFS (μs)
3	288	320
6	164	196
12	106	138

3.2.2 *Synchronous OBU Window Length*

It is important to determine the maximum length of the Synchronous OBU Window (SOW) for use with the IEEE802.11p/WAVE or other similar standard.

Assuming a worst case scenario of attributing slots for all OBUs travelling in the zone which is simultaneously covered by more than one RSU, the length of SOW is presented in Eq. (3.6):

$$l_{SOW} = SOW_{slots} \times (IFS + BSM) \tag{3.6}$$

Considering that the safety message that is transmitted by each OBU, named as Basic Safety Message (BSM) has a size of 390 bit [15] and that the inter-frame space is smaller than the used in the IEEE 802.11 standard, which is 32 μs for the Shortest Inter Frame Space (SIFS) in a 10 MHz channel [11], then the time needed to transmit a BSM of 390 bit is shown in Table 3.2 according to the bit rate used.

It is important to find out the number of slots available for OBU transmission in the Synchronous OBU Window (SOW_{slots}). The number of SOW_{slots} is presented in Eq. (3.7) and varies from 0 to:

$$SOW_{slots} = (S_{IW} \times N_{VRSU}) - (S_{IW} - 1) \times N_{V_{int}} \tag{3.7}$$

N_{VRSU} and S_{IW} were already determined earlier, another important parameter is the value of N_{Vint} which is presented in Eq. (3.8).

$$N_{V_{int}} = \bigcup_{i=1}^{S_{IW}-1} S_{RSU_i} \cap S_{RSU_{i+1}} \tag{3.8}$$

In other words $N_{V_{int}}$ is the number of vehicles that can fit in the overlapping range O_r. In Table 3.3 the maximum values of SOW_{slots} are shown:

The time needed to transmit a maximum size Synchronous OBU Window (SOW) is obtained by multiplying the values in Tables 3.1 and 3.2. The results are shown in Fig. 3.6.

Table 3.3 Maximum size of SOW_{slots} for a RSU coverage of 750 m with 25 % of overlapping range

SOW_{slots}	Normal traffic	Traffic jam
1 lane	76	180
2 lanes	152	360
3 lanes	228	540
4 lanes	304	720
5 lanes	380	900

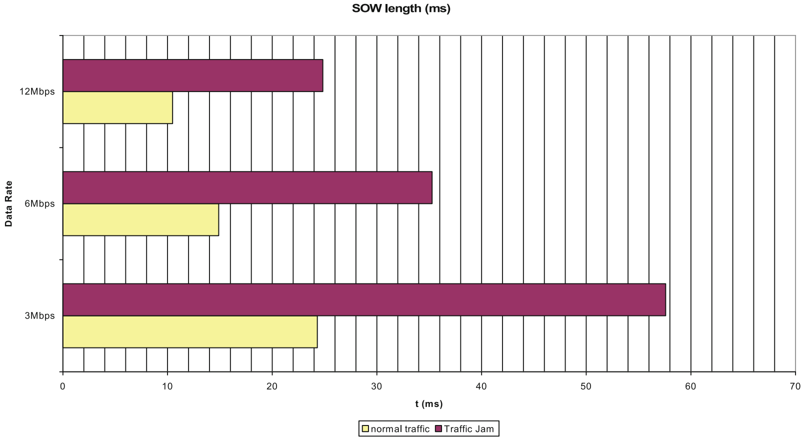


Fig. 3.6 SOW length per lane (ms)

In order to better clarify the meaning of these values, as an example, consider a motorway with 4 lanes per travel path and that the IEEE 802.11-2012 standard is used. Since the size of a CCH interval for the WAVE protocol varies from 50 to 100 ms, for the worst case scenario of having the motorway fully populated with vehicles, it is not possible to allow all OBUs to update their status in every EC, particularly for the case of a large motorway and a traffic jam scenario. This is shown in Fig. 3.7.

The CCH interval has a size that will not be larger than 100 ms (it is 50 ms by default) so it is easy to roughly determine the maximum number of vehicles served per CCH interval. The maximum available transmission time for the SOW window in each CCH interval will be 100 ms for the continuous mode or 50 ms subtracted by the Guard Interval (4 ms) for the normal mode:

$$\begin{aligned} \text{Maximum length of SOW} &= 100 \text{ ms (continuous mode) or,} \\ \text{Maximum length of SOW} &= 50 \text{ ms} - \text{GI} \\ &= 46 \text{ ms (normal mode)} \end{aligned} \tag{3.9}$$

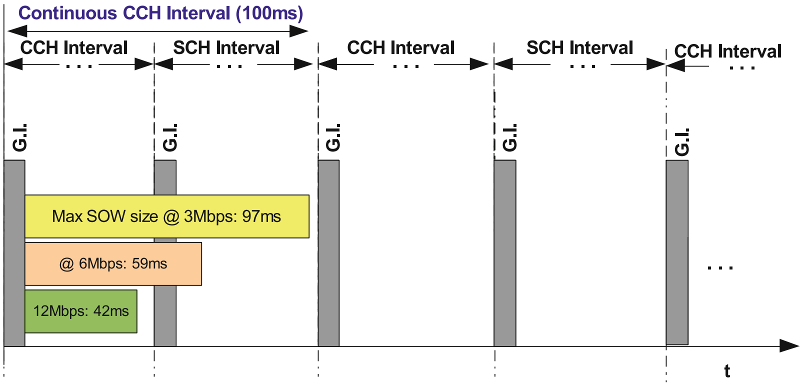


Fig. 3.7 Maximum SOW length for normal traffic using IEEE 802.11-2012 ($n_{lanes} = 4$)

Table 3.4 Maximum number of SOW_{slots} per CCH interval (upper bound)

Bit rate (Mbps)	Continuous mode	Normal mode
3	312	143
6	510	234
12	724	333

The SOW length will in fact be smaller than that, since transmission time for the IW must also be guaranteed, as well as to reserve a free period for non-enabled OBUs. For now, the above values will be used as a maximum reference value for the length of SOW, thus obtaining the following upper bound for the number of SOW_{slots}, shown in Table 3.4.

By comparing Table 3.4 with Table 3.3, it can be seen that the usual bit rates used for safety services, 6 and 12 Mbps [14], are not enough to serve all vehicles in one full Elementary Cycle, particularly in traffic jam scenarios in motorways with more than 3 lanes per travel path. However, the values above are an upper bound and will be redefined later on.

3.2.3 *Infrastructure Window Length*

After determining the SOW length the Infrastructure Window (IW) length will be quantified. The IW is used by each RSU to send the trigger message (TM) along with possible warning messages (WM). Those messages will be included in each RSU transmission slot. Recall that this RSU slot has a fixed size and that SIW is equal to 2, thus meaning that IW will have a duration equal to (3.10):

$$IW = S_{IW} \times (RSU_{slot} + IFS) \tag{3.10}$$

RSU_{ID}	t_{SOW}	t_{ID207}	tr_{S22}	t_{ID007}	tr_{S87}	. . .	t_{ID622}	tr_{S33}

Fig. 3.8 Trigger Message frame

In order to compute the size of RSU_{slot} the length of a TM and a WM must be analysed.

A Trigger Message (TM) starts with an RSU_{ID}, followed by a parameter (t_{SOW}) that indicates how many μs separate the beginning of this TM from the beginning of the SOW, and then a series of temporary OBU identifiers (t_{ID}) and the respective transmission slot (tr_s). An example of a TM frame is shown in Fig. 3.8.

First there is the need to determine how many bits are necessary for RSU identification. Using 8 bit as a starting value for RSU_{ID} is enough to identify 256 distinct RSUs, and allows to cover 168 km of motorway for both travel sides, considering the C_r determined earlier.

In order to define the size of the Trigger Message frame, it is important to quantify the possible maximum value for t_{SOW}. The minimum value occurs in the last RSU_{slot} and corresponds to the duration of the RSU_{slot}. The maximum value occurs in the first RSU_{slot} and corresponds to:

$$\text{Maximum value for } t_{SOW} = IW - IFS \tag{3.11}$$

There is a circular reference because it seems the TM size depends on the TM itself, but it is possible to work around this considering the absurd case where the IW occupies the maximum possible available length in a CCH interval in WAVE, i.e., 100 ms. Since t_{SOW} is is expressed in μs it means at least 17 17 bits are needed to properly define t_{SOW}. This value can be updated later on if needed.

In the previous section t_{ID} was defined to have 16 bit. As for the number of bits necessary for the OBU transmission slot, the worst-case scenario occurs when there is the need to code 724 different OBU transmission slots (please refer to Table 3.4). This means at least 10 bit for tr_s are necessary.

In resume:

- RSU_{ID} has a length of 8 bits;
- t_{SOW} has a maximum length of 17 bits (to be refined later);
- each t_{ID} has a length of 16 bits;
- each tr_s has a length of 10 bits.

In the worst case scenario of a traffic jam, in case there is the need to allow transmission slots for all OBUs, a TM would occupy:

$$8 + 17 + 724 \times (16 + 10) = 18849 \text{ bits} \tag{3.12}$$

This is the case for the higher bit rate. For 3 and 6 Mbps it was determined (Table 3.4) that the number of SOW_{slots} will never exceed 312 and 510 vehicles,

Table 3.5 Size of a Trigger Message (TM) in bits (upper bound)

Bit rate (Mbps)	Continuous mode	Normal mode
3	8137 bit	3743 bit
6	13285 bit	6109 bit
12	18849 bit	8683 bit

Table 3.6 Transmission duration of a TM in an OFDM 10 MHz channel (upper bound)

Bit rate (Mbps)	Continuous mode (ms)	Normal mode (ms)
3	2.86	1.40
6	2.32	1.12
12	1.64	0.79

respectively. This means the TM may have different sizes according to the transmission rate, as is shown in Table 3.5.

In Table 3.6 the time it takes to transmit a maximum size TM using WAVE is shown, for both traffic jam and normal traffic cases. In WAVE the bit rates used range from 3Mbps to 12 Mbps. The time needed to transmit a TM is shown in the next table, based on the IEEE 802.11p/WAVE MAC standard, adding the header and frame check sequence to the message size, and then calculating the padding bits necessary according to the bit rate used.

Message (WM) that is sent from the RSUs to all vehicles travelling in the Safety Zone. Several type of safety events can occur. For example, the Curve Speed Warning event needs a 235 bit payload. A more common safety message must include the following fields:

- eventID.
- sourceID.
- transmitterID.
- location.
- additional info.

16 bits are enough for the eventID field, sourceID and transmitterID are RSUs, so 8 bits for each of them will suffice. For the location 112 bits are used for the GPS coordinates. This means the minimum size of a WM is 144 bits. According to this, Table 3.7 shows the time needed to transmit a minimum WM and a curve speed warning message using IEEE802.11p/WAVE.

In order to quantify the size of an RSU slot, there is the need to find out the maximum number of Warning Messages to be transmitted per EC or CCH interval. This is not the same as asking how many simultaneous safety events can occur in

Table 3.7 Transmission duration of a Warning Message in an OFDM 10 MHz channel

Bit rate (Mbps)	Continuous mode (μs)	Normal mode (μs)
3	200	232
6	124	140
12	82	90

Table 3.8 Transmission duration of a RSU slot using an OFDM 10 MHz channel (upper bound)

Bit rate (Mbps)	Continuous mode (ms)	Normal mode (ms)
3	5.18	3.72
6	3.72	2.52
12	2.54	1.69

Table 3.9 Transmission duration of the Infrastructure Window ($S_{IW} = 2$) using an OFDM 10 MHz channel

Bit rate (Mbps)	Continuous mode (ms)	Normal mode (ms)
3	10.42	7.50
6	7.50	5.10
12	5.14	3.44

a RSU coverage, since RSUs might want to broadcast events that occur outside its coverage area, e.g., an accident that occurs ahead in the path of travel. Imposing a limit of 10 WMs per RSU Slot and considering the scenario of having 10 WMs to be broadcast in each RSU slot, then each RSU slot needs to have a maximum size TM + 10 × WM, which is summarized in Table 3.8.

Based on equation Eq. (3.10) the worst-case maximum size of IW can be determined (results are shown in Table 3.9).

In the previous sub-section it was shown that the SOW could not have the determined size (refer to Fig. 3.7) since it exceeds the CCH interval. A limit for the SOW maximum size was determined, based on the full length of CCH interval, and consequently a new upper bound for the IW (since the SOW size influences the TM size and the RSU slot).

In the beginning of this Sect. 3.2.3 we found we would need 17 bits for tsow and left this value to be later refined. After determining a more realistic upper bound value for the Infrastructure Window we can safely reduce the size of t_{SOW} from 17 to 14 bits.

In the beginning of this Sect. 3.2.3 the value of 17 bits for t_{SOW} was determined but this value was to be later refined. After determining a more realistic upper bound value for the Infrastructure Window tsow can safely be reduced from 17 to 14 bits. This means that TM will have its upper bound size reduced by 3 bits. However, after using these new values, these extra 3 bits do not make any difference in the

Table 3.10 Transmission duration of a regular WSA using an OFDM 10 MHz channel

Bit rate (Mbps)	Duration (µs)
3	304
6	172
12	106

transmission duration of a TM due to the usage of pad bits in OFDM. The TM equation will nevertheless be updated to

$$8 + 14 + \mathrm{SOW}_{\mathrm{slots}} \times (17 + 10) \tag{3.13}$$

3.2.4 *Free Period (FP) Length*

In this sub-section the length of the free period is discussed. This length will be variable, since it depends on the number of vehicles that are present in the area covered by the RSUs. There is the need of defining a minimum free period length, in order to guarantee transmission opportunities for non-enabled vehicles and for Wave Service Announcements or non-safety applications, as shown in (3.14):

$$\mathrm{FP}_{\mathrm{min}} = (\sigma) \times (\mathrm{CCH}_{\mathrm{Interval}}), \text{ where } 0 < \sigma < 1 \tag{3.14}$$

Considering σ equal to 10 %, it means 5–10 ms are reserved for Wave Service Announcements (WSA) or other communications. Taking into account the example of a WSA given in [12] the duration of a transmission of a regular WSA is shown in Table 3.10. This allows for 16 to 32 WSAs to be transmitted in one CCH interval, which is acceptable for non-urban scenarios.

In some particular cases, the FP length can be reduced to zero, if emergency communications need to use the whole Elementary Cycle.

3.2.5 *Synchronous OBU Window Adjustments*

Considering $\mathrm{FP}_{\mathrm{min}}$ to have a value of 10 % the CCH interval, the SOW maximum size and TM sizes will be recalculated, as shown in (3.15):

$$\mathrm{SOW} = \mathrm{E} - \mathrm{GI} - \mathrm{IW} - \mathrm{FP}\ (\mathrm{GI} = 0 \text{ in continuous mode}) \tag{3.15}$$

Table 3.11 Transmission duration of a TM using an OFDM 10 MHz channel

TM	$FP_{min} = 10\,\%$ of CCH interval		No free period	
Bit rate (Mbps)	Continuous mode (ms)	Normal mode (ms)	Continuous mode (ms)	Normal mode (ms)
3	2.34	1.08	2.60	1.21
6	1.94	0.91	2.16	1.01
12	1.41	0.67	1.56	0.75

Table 3.12 Transmission duration of IW using an OFDM 10 MHz channel IW

TM	$FP_{min} = 10\,\%$ of CCH interval		No free period	
Bit rate (Mbps)	Continuous mode (ms)	Normal mode (ms)	Continuous mode (ms)	Normalpg mode (ms)
2	9.39	6.86	9.90	7.12
5	6.74	4.68	7.18	4.89
12	4.68	3.20	4.99	3.36

Table 3.13 Time left for SOW transmission using an OFDM 10 MHz channel

TM	$FP_{min} = 10\,\%$ of CCH interval		No free period	
Bit rate (Mbps)	Continuous mode (ms)	Normal mode (ms)	Continuous mode (ms)	Normal mode (ms)
3	80.60	34.13	90.10	38.88
6	83.26	36.32	92.82	41.11
12	85.32	37.80	95.01	42.64

Table 3.14 Number of SOW_{slots} per CCH interval

TM	$FP_{min} = 10\,\%$ of CCH interval		No free period	
Bit rate (Mbps)	Continuous mode (ms)	Normal mode (ms)	Continuous mode (ms)	Norma l mode (ms)
3	251	106	281	121
6	424	185	473	209
12	618	273	688	309

Because of the relationship between TM and SOW_{slots} the length of SOW is recalculated as well as its respective SOW slots, assuming the initial IW length. Since the number of SOW_{slots} is slightly reduced so does the TM length and the IW length. By reintroducing this new IW length a more accurate SOW length is obtained, repeating the whole process until the values are close enough to the previous iteration. In the end, the following tables are obtained (Tables 3.11, 3.12, 3.13 and 3.14) for TM length, IW and SOW length.

By comparing the results of Table 3.1 with Table 3.14 it can be seen that in some exceptional cases it might be worth using the whole CCH interval for the V-FTT protocol, not allowing the existence of a free period (for a short amount of time) in order to accommodate more vehicles in the SOW. For larger motorways there is the need of a scheduling mechanism to fairly allocate OBUs to SOW slots and also to allocate RSU slot time between trigger messages and warning messages.

3.3 Analysis of Impact of Worst Case Scenario

It is important to analyse the impact of a worst case scenario in the V-FTT protocol, particularly what happens due to the expiry of transmission chance before the maximum tolerable delay for an application. For this analysis, the packet loss probability derived from transmission losses are excluded as well as other factor such as packet collisions.

Consider that the number of OBUs in $\mathrm{S_{IW}}$ RSUs coverage is $\mathrm{n_v}$, where:

$$\mathrm{n_v} = 1 \text{ to } \mathrm{N} \tag{3.16}$$

The ratio of denied transmissions ($\mathrm{t_{dn}}$) due the expiry of CCH interval can then be determined by Eq. (3.17):

$$\mathrm{t_{dn}} = \begin{cases} 0 & \text{if } \mathrm{n_v} \le \mathrm{SOW_{slots}} \\ \left(1 - \frac{\mathrm{SOW_{slots}}}{\mathrm{N}}\right) & \text{if } \mathrm{n_v} > \mathrm{SOW_{slots}} \end{cases} \tag{3.17}$$

Whenever the number of vehicles fits in the existing Synchronous OBU Window there will be no denied transmissions since all OBUs can transmit within a CCH interval. If the number of vehicles exceeds the number of slots in SOW then the probability of not having a transmission opportunity in the current CCH interval will be higher.

Based on Table 3.14 and the previous equation the results for two typical vehicular safety applications are shown next: the Emergency Electronic Brake Light (EEBL) (refer to Fig. 3.9) with a maximum latency of 100 ms and the Post crash warning (refer to Fig. 3.10) with a maximum latency of 500 ms.

Results show that the ratio of denied transmissions due to the expiry of transmission chance is acceptable when using the higher transmission bit rate for the safety applications with tighter latency constraints. An obvious conclusion is that choosing not to use the Free Period for non-enabled vehicles decreases this ratio since there is the possibility to accommodate more OBUs in the SOW. For the safety applications with higher latency the V-FTT protocol is perfectly suitable even with lower bit rates. In the next sub-sections the worst-case delay scenario for the V-FTT protocol applied to IEEE802.11-2012 will be investigated.

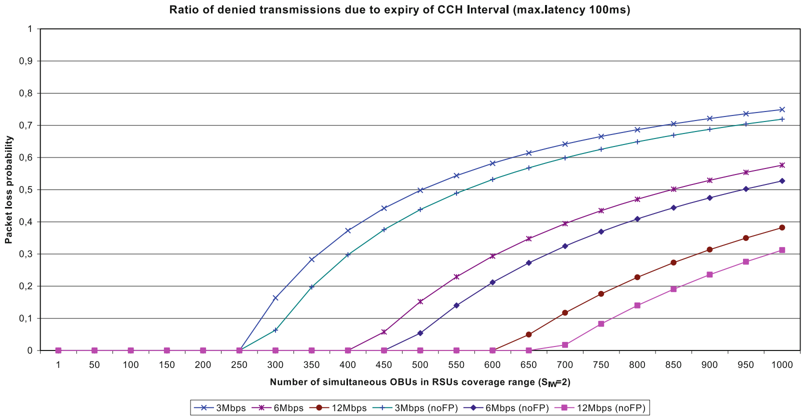


Fig. 3.9 Ratio of denied transmissions due to CCH Interval expiry for EEBL application

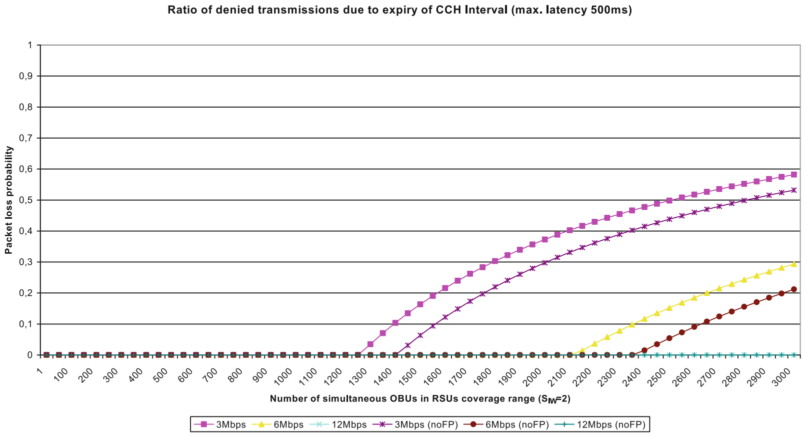


Fig. 3.10 Ratio of denied transmissions due to CCH Interval expiry for Post-Crash Warning application

3.4 V-FTT Protocol Worst Case Delay Analysis

The V-FTT protocol will now be analysed in terms of the time that passes between the instant of occurrence of an event and the instant a vehicle is warned of the event, i.e., the end-to-end delay.

Consider that within the set of vehicles travelling in the safety zone, a vehicle detects a safety event (e.g. accident, problem with vehicle). The worst case in terms of time occurred between an event detection and the instant of time the last vehicle in the Safety Zone is warned by the RSUs is an important parameter, therefore there is the need to analyse the involved times:

- t_{V2I}—period of time that occurs since the detection of an event by an OBU until the event transmission to an RSU.
- t_{valid}—period of time that occurs since the RSU is effectively warned until the RSU considers the event is valid.
- $t_{schedule}$—period of time that occurs since the RSU validates an event and schedules the TM and WM according to the event.
- t_{I2V}—period of time that occurs since a TM and/or WM is scheduled by an RSU until the transmission of a warning message by the RSUs.

To simplify this reasoning, consider for now that transmissions of WM are always received successfully by all OBUs inside the coverage area.

3.4.1 Uplink Time (t_{V2I})

The worst-case for t_{V2I} occurs when an OBU detects the event just after it transmitted its Basic Safety Message (BSM). This means the OBU will have to wait for its next allocation slot to transmit. Consider this OBU the emitter OBU just for reasoning purposes. Consider the simplest fair scheduling scheme where all OBUs have one transmission opportunity and will have the second transmission opportunity after all the others had their first. Then the worst case scenario occurs when the emitter OBU is only allowed to transmit after all the remaining OBUs in the same coverage area of the Safety Zone have had their chance to transmit. How many OBUs are involved? The worst-case is when the Safety Zone is completely filled with vehicles. Those numbers were presented in Table 3.3. The maximum number of OBUs travelling in the Safety Zone depends on the motorway topology, i.e., on the number of existing lanes per travel path. This means the maximum waiting time for the emitter OBU will be variable. Consider that the maximum number of OBUs present in the same coverage area than the emitter OBU is named M_{OBU}. The value of M_{OBU} is in fact the value of Table 3.3 subtracted by one, which is the emitting OBU. Those numbers are shown in Table 3.15.

Since for each Elementary Cycle there is a limit of maximum SOW_{slots} available, the emitter OBU will have to wait for some ECs until it has the chance to transmit. This is named W_{EC}, number of waiting Elementary cycles. W_{EC} is shown in (3.18), and is equal to the floor of the division of M_{OBU} by the maximum number of SOW_{slots}

Table 3.15 Maximum number of OBUs in the same coverage area than the emitter OBU ($S_{IW} = 2$, $C_r = 750$ m)

N_{VRSU}	Normal traffic	Traffic jam
1 lane	75	179
2 lanes	151	359
3 lanes	227	539
4 lanes	303	719
5 lanes	379	899

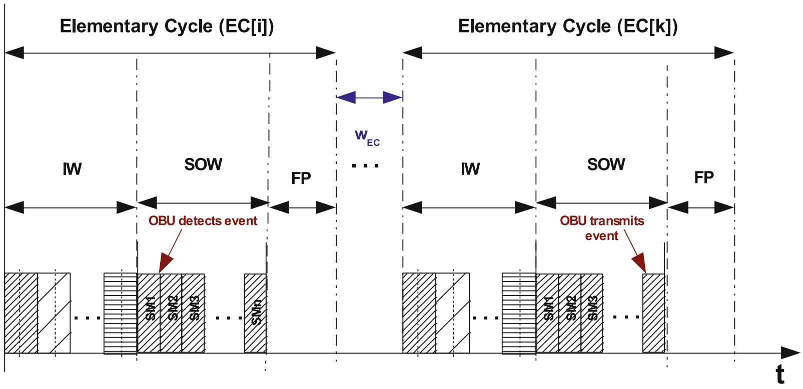


Fig. 3.11 Worst case OBU transmission instant (t_{V2I})

available (refer to Table 3.14).

$$W_{EC} = \left\lceil \frac{M_{OBU}}{SOW_{slots}} \right\rceil \tag{3.18}$$

The worst case OBU transmission instant is depicted in Fig. 3.11, where OBU detects the event just after it transmitted its Basic Safety Message (BSM) and will have to wait for its next available slot to transmit.

If scheduling is made per elementary cycle, the only guarantee the emitter OBU will have is that it will be scheduled in the SOW after W_{EC}. The worst case happens when it is scheduled in the last slot and is shown in (3.19):

$$t_{V2I} = SOW + (W_{EC} + 1) \times E \tag{3.19}$$

Figures 3.12 and 3.13 show the results of our calculations for two scenarios, normal traffic and traffic jam, considering that the free period has no minimum length, as it was shown earlier that this is the worst-case scenario. The EC can have a duration of 50 ms (N-normal mode) or 100 ms (C-continuous mode).

As the number of lanes increases, so does the maximum possible number of vehicles, which leads to an increase of uplink time. It is interesting to find out that the continuous mode of operation leads to higher uplink time for the case of smaller motorways (two lanes per travel path or less). This is due to the fact that all vehicles transmissions can be accommodated in one SOW, and OBUs have to wait a full EC to transmit. It can also be seen that 3 Mbps is insufficient for large motorways and dense scenarios, hence the ITS-G5 determination of using 6 and 12 Mbps for safety applications [14]. These results also reinforce the fact that a scheduling mechanism is needed, since straightforward fair slot allocation can lead to intolerable values for some safety applications.

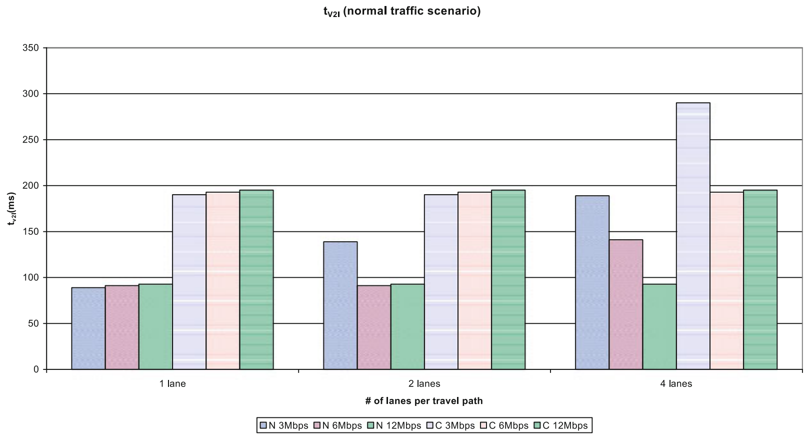


Fig. 3.12 Uplink time (t_{V2I}) worst case for normal traffic scenario (FP = 0 %, C_r = 750 m)

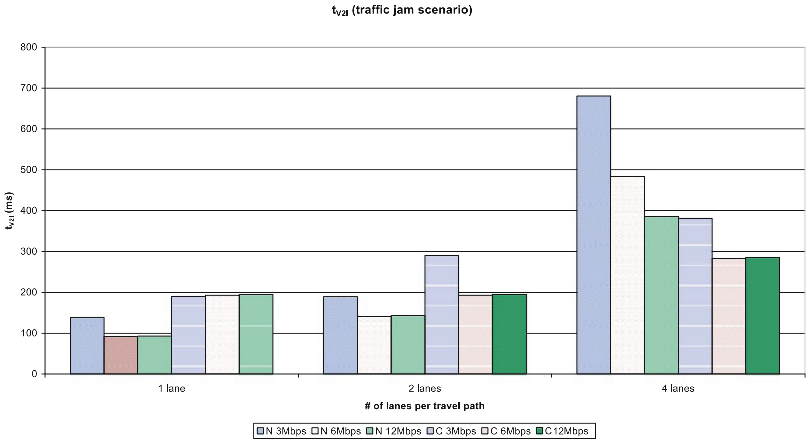


Fig. 3.13 Uplink time (t_{V2I}) worst case for traffic jam scenario (FP = 0 %, C_r = 750 m)

3.4.2 Validation Time (t_{valid}) and Scheduling time ($t_{schedule}$)

The validation time is the period of time that occurs since the RSU has received the event warning, until it considers the event is valid. The validation time depends on several factors, since the RSU must compare the information received from several sources, such as induction sensors, cameras, radar or even other OBU messages, in order to validate the event.

The scheduling time is the period of time that occurs since the RSU validates an event and schedules the TM and WM according to the event.

Both times are usually combined. The worst case happens when the RSU receives the information in the last slot of SOW. For the case the RSU has the first RSU slot, it means that the RSU must perform the validation, schedule and build its TM and

WM during the Guard Interval, i.e., in less than 4ms. RSUs should have sufficient computation power to achieve this goal.

3.4.3 Downlink Time (t_{I2V})

The worst case downlink time happens when the RSU receives the information from OBUs in the first SOW slot and it will have to wait until the next Elementary Cycle (EC) for the chance to transmit (Fig. 3.14).

In conclusion, the validation time and scheduling time is included in t_{I2V}.

It was shown earlier that the SOW duration is variable and has a maximum value whenever FP = 0. This means the worst-case of t_{I2V} is in fact equivalent of a full duration of an Elementary Cycle (E) subtracted by the duration of a TM (refer to (3.20)).

$$t_{I2V} = (E - TM) \tag{3.20}$$

The results are summarized in Table 3.16.

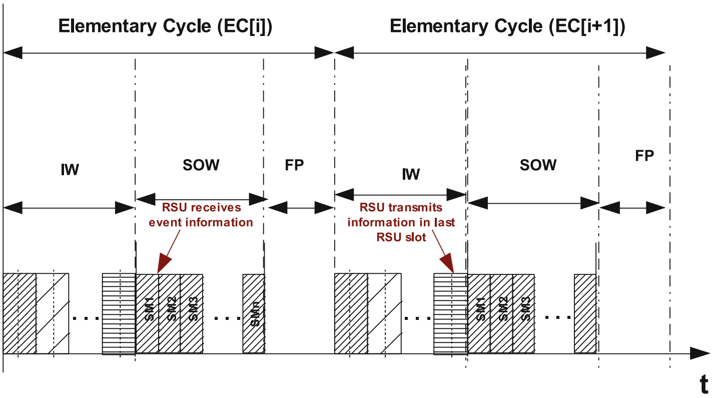

Elementary Cycle (EC[i])
Elementary Cycle (EC[i+1])
IW
SOW
FP
IW
SOW
FP
RSU receives event information
RSU transmits information in last RSU slot
SM1
SM2
SM3
SMn
SM1
SM2
SM3
t

Fig. 3.14 Worst case of (t_{I2V})

Table 3.16 Worst case value of validation, schedule and downlink time ($S_{IW} = 2$, $C_r = 750$ m)

Bit rate (Mbps)	Normal mode (ms)	Continuous mode (ms)
3	48.79	97.40
6	48.99	97.84
12	49.25	98.44

3.4.4 Worst Case Time Between Event Detection and OBU Warning (t_{worst})

After determining all the times involved, it can now be determined the worst case in terms of time occurred between an event detection and the instant of time the last vehicle in the Safety Zone is warned by the RSUs. This will be referred as $\mathrm{t_{worst}}$. At first, one might think that tworst is obtained by adding the uplink time worst case with the downlink time worst case, which is shown by (3.21):

$$\begin{aligned} \mathrm{t_{worst}} &= (\mathrm{t_{V2I}} + \mathrm{t_{I2V}}) \\ &= \mathrm{SOW} + (\mathrm{W_{EC}} + 1) \times \mathrm{E} + \mathrm{E} - \mathrm{TM} \\ &= \mathrm{SOW} - \mathrm{TM} + (\mathrm{W_{EC}} + 2) \times \mathrm{E} \end{aligned} \tag{3.21}$$

Equation (3.21), however, can only be used as an upper bound, since $\mathrm{t_{V2I}}$ and $\mathrm{t_{I2V}}$ worst case times never occur simultaneously.

Consider that $\mathrm{t_{V2I}}$ occurs, meaning the OBU transmits in the last slot in the SOW. This means the remaining maximum value for $\mathrm{t_{I2V}}$ will be approximately equal to FP + IW, assuming that the value of FP will not change significantly from one elementary cycle to another. This means that in that case the value of $\mathrm{t_{worst}}$ can be determined by (3.22):

$$\begin{aligned} \mathrm{t_{worst}} &= \mathrm{SOW} + (\mathrm{W_{EC}} + 1) \times \mathrm{E} + \mathrm{FP} + \mathrm{IW} \\ &\cong (\mathrm{W_{EC}} + 2) \times \mathrm{E} \end{aligned} \tag{3.22}$$

If in the other hand, the $\mathrm{t_{I2V}}$ worst case happens, this means the OBU managed to transmit its event in the first transmission slot, meaning that the value of $\mathrm{t_{V2I}}$ will be equal to: $\mathrm{SOW} + \mathrm{FP} + \mathrm{W_{EC}} \times \mathrm{E} + \mathrm{IW}$, which is the same as: $(\mathrm{W_{EC}} + 1) \times \mathrm{E}$. This means that $\mathrm{t_{worst}}$ will be determined by (3.23):

$$\mathrm{t_{worst}} = \mathrm{E} + (\mathrm{W_{EC}} + 1) \times \mathrm{E} \cong (\mathrm{W_{EC}} + 2) \times \mathrm{E} \tag{3.23}$$

In conclusion, the worst case time between event detection and OBU warning is linearly dependent on the duration of the Elementary Cycle (E), but keep in mind that $\mathrm{W_{EC}}$ depends on the number of maximum $\mathrm{SOW_{slots}}$ per EC, which in turn depends on E, so reducing E would also reduce $\mathrm{SOW_{slots}}$ and increase $\mathrm{W_{EC}}$. The results are summarized in Tables 3.17, 3.18 and depicted in Figs. 3.15 and 3.16, where it is shown the difference between using different bit rates for the normal mode of operation or the continuous mode of operation, where E is longer in the later situation.

Table 3.17 Worst case warning time for normal traffic (no FP)

Normal traffic	1 lane		2 lanes		4 lanes	
Bit rate (Mbps)	Continuous mode (ms)	Normal mode (ms)	Continuous mode (ms)	Normal mode (ms)	Continuous mode (ms)	Normal mode (ms)
3	200	200	200	300	300	400
6	200	200	200	200	200	300
12	200	200	200	200	200	200

Table 3.18 Worst case warning time for traffic jam (no FP)

Traffic jam	1 lane		2 lanes		4 lanes	
Bit rate (Mbps)	Continuous mode (ms)	Normal mode (ms)	Continuous mode (ms)	Normal mode (ms)	Continuous mode (ms)	Normal mode (ms)
3	200	300	300	400	400	700
6	200	200	200	300	300	500
12	200	200	200	300	300	400

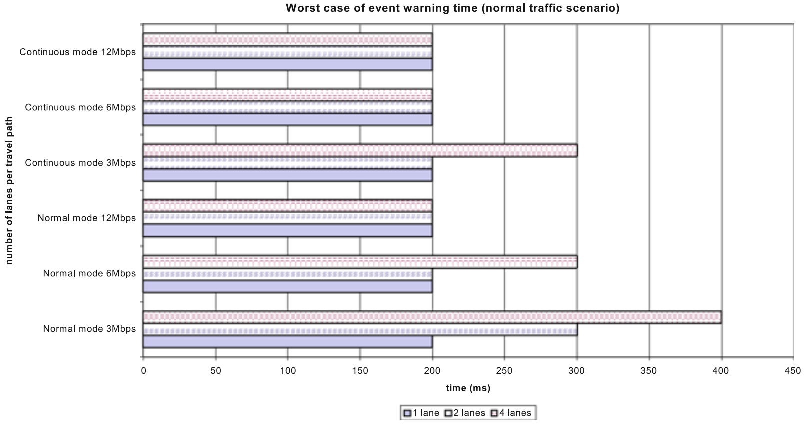


Fig. 3.15 Worst case of event warning time per number of lanes (normal traffic)

Analysing the results, it is obvious that for the lowest bit rates the worst-case results are not tolerable for the most stringent delay safety applications. However, some of those maximum latency delays (e.g. Emergency Electronic Brake Light) were computed for a particular high speed scenario (more than 100 km/h). For the traffic jam scenario, vehicles are not expected to travel at such high speeds. The use of the highest bit rate should nevertheless mitigate the problem. Since worst-case results are correlated with the duration of the Elementary Cycle, smaller ECs can have better results for the cases where the number of OBUs fits inside one SOW, not exceeding one EC. However, if using IEEE 802-11-2012, the EC must be fixed and

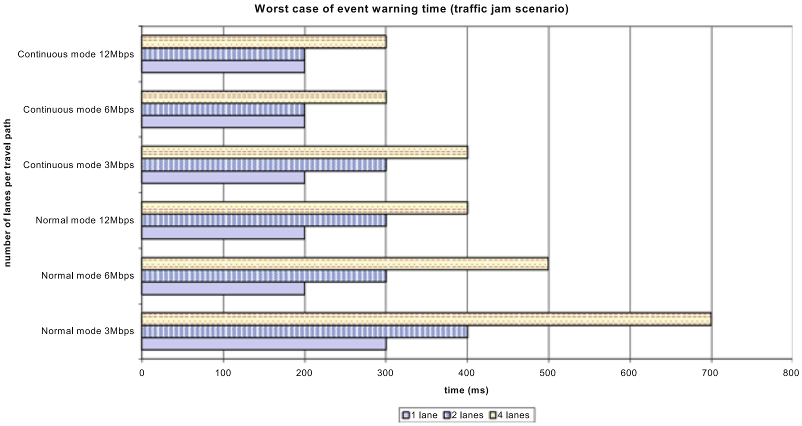


Fig. 3.16 Worst case of event warning time per number of lanes (traffic jam)

equal to the CCH interval. For other standards, the effect of having a smaller EC in the normal situation latency would have to be studied.

3.5 Application Scenario: A5—Auto-Estrada da Costa Do Estoril

In this section a realistic application scenario is presented: the A5—Autoestrada da Costa do Estoril, which is one of the busiest motorways in Portugal. The V-FTT protocol is applied to A5 motorway using theoretical worst-case calculations and MATLAB together with an event generator.

3.5.1 A5 Motorway General Description

The A5 motorway connects Lisbon to Cascais and is 25 km long. The average daily traffic load, based on monthly values in 2009 and first three months of 2010, is close to 74,000 vehicles, although in some sections of the A5 it can reach up to 134,000 vehicles [16]. The A5 motorway concessionary, BRISA SA, kindly provided data from peak hour traffic in October 2013. The number of lanes varies throughout its course, as can be seen in Table 3.19.

The motorway locations where serious accidents occur or where accidents occur more frequently are named blackspots. From 1996 to 2006, several blackspots were identified in the A5 motorway [10]. The author decided to join contiguous blackspots reaching a final number of 22 blackspots. The kilometre numbering is the same used

Table 3.19 A5 motorway characteristics (Adapted from [10] and [16])

A5 Subsection	Distance (km)	Number of lanes	ADT (average daily traffic)	Number of accidents (2003–2006)	Highest monthly peak hour traffic
Viaduto Duarte Pacheco to Miraflores	4.0	4	>120,000	177	18,728
Miraflores to Linda-a-Velha	1.5	3	>120,000	253	7398
Linda-a-Velha to Estádio Nacional	2.7	3	>120,000	216	6862
Estádio Nacional to Oeiras	5.4	3	>120,000	32	6956
Oeiras to Estoril	9.0	3	>67,000	42	6738
Estoril to Cascais	5.3	2	>38,000	N/A	N/A

Table 3.20 A5 Motorway blackspots (Adapted from [10])

Blackspot	km	Blackspot	km
1	0.1–0.6	12	6.0–6.1
2	0.8–0.9	13	6.3–6.4
3	1.0–1.1	14	6.8–7.2
4	1.5–1.6	15	7.3–7.6
5	1.8–1.9	16	7.8–8.1
6	2.0–2.2	17	8.5–8.6
7	2.4–2.6	18	8.8–9.1
8	2.8–3.1	19	10.0–10.1
9	3.8–4.5	20	11.8–11.9
10	4.7–5.0	21	14.3–14.4
11	5.8–5.9	22	14.5–14.6

in A5, where 0 km corresponds to Lisbon and 27.4 km to Cascais. Refer to Table 3.20 for more details.

Considering that overlapping of RSU coverage will exist, the 22 blackspots presented in Table 3.20 can be converted in the following three Safety Zones:

- Safety Zone 1 would cover km 0 to km 3.1.
- Safety Zone 2 from km 3.8 to km 5.
- Safety Zone 3 would cover black spot 11 (km 5.8 and 5.9).

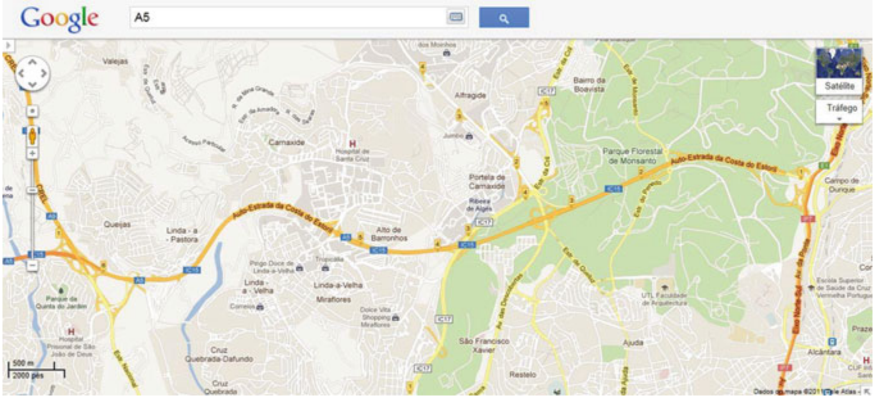


Fig. 3.17 Safety Zones suggestion for A5 motorway (Adapted from [9])

Table 3.21 Average vehicle dimensions (Adapted from [6])

Vehicle type	Average width (m)	Average height (m)	Average length (m)
Passenger light vehicle	1.75	2.06	4.58
Bus	2.50	3.45	11.08
Truck	2.45	4.00	9.00
Lorry with trailer	2.55	4.00	15.60

In Fig. 3.17 the three Safety Zones are drawn upon the A5 motorway.

In order to better understand the A5 motorway environment the following information about the Portuguese law is provided:

- The maximum allowed speed in Portuguese roads is 120 km/h.
- The maximum vehicle dimensions are [4]:
  - Maximum width: 2.6 m;
  - Maximum height: 4 m;
  - Maximum length (passenger vehicle): 12 m;
  - Maximum length (truck): 18 m;

Average vehicle dimensions can prove to be useful for further calculations, therefore typical average vehicle dimensions are shown in Table 3.21.

In the next sub-sections the V-FTT feasibility in the A5 motorway will be analysed.

3.5.2 V-FTT Feasibility Using the A5 Motorway

It is useful to quantify some of the V-FTT variables in what refers to its application on the A5 motorway scenario. This is done by using Eq. (3.4) from Sect. 3.2.1:

Table 3.22 Maximum simultaneous number of vehicles in each A5 motorway Safety Zone

Safety Zone	Normal traffic	Traffic jam
Safety Zone 1 (3100 m)	359	850
Safety Zone 2 (1200 m)	139	329
Safety Zone 3 (100 m)	12	28
A5 Motorway	3170	7518

Table 3.23 Number of RSUs to place in A5 motorway ($C_r = 750$ m, $S_r = 1312.5$ m)

Safety zone	Number of RSUs per travel path
Safety Zone 1 (3100 m)	4
Safety Zone 2 (1200 m)	2
Safety Zone 3 (100 m)	1
A5 Motorway	22

$$n_{S_z} = \frac{l_{S_z}}{\left(\left(V_{\text{length}} \times \left(1 - Tr_{\text{perct}}\right) + Tr_{\text{length}} \times \left(Tr_{\text{perct}}\right)\right) + v_{\text{spacing}}\right)} \times n_{\text{lanes}} \qquad (3.24)$$

For the case of Safety Zone 1, $l_{S_z} = 3100$ m, $n_{\text{lanes}} = 4$, $V_{\text{length}} = 4.58$ m, $Tr_{\text{length}} = 9$ m, v_{spacing} varies between 10 m (traffic jam) and 30 m (normal traffic) [19] and $Tr_{\text{perct}} = 0\,\%$, since the worst-case scenario occurs when more vehicles are inside the Safety Zone.This results in 359 vehicles that can fit in Safety Zone 1 in normal traffic conditions, rising to 850 in case of a traffic jam.

Considering that in the future one might extend the Safety Zone to the whole A5 motorway, re-using Eq. (3.24) with $l_{S_z} = 27{,}400$ m a maximum of 7518 vehicles per travel path is obtained. Table 3.22 summarizes the results for the three Safety Zones in A5.

The spacing between RSUs was determined in Eq. (3.3) and is equal to 1312.5 m. This means that the number of RSUs placed in each Safety Zone can be determined. Results are shown in Table 3.23:

3.5.2.1 Worst-Case Calculations for A5 Safety Zone 1

In Table 3.22 it was shown that Safety Zone 1 can have a maximum of 850 simultaneous vehicles. Since at least 4 RSUs are deployed for Safety Zone 1 it means that worst-case scenario defined in Table 3.3 for each RSU coverage will not be reached. Dividing the 850 vehicles equally throughout the entire Safety Zone (since this is a traffic jam scenario) there will be slightly more than 411 vehicles per RSU coverage, but since RSUs coverage overlap it means that there will be approximately 360 vehicles per RSU. Results are obtained by repeating the same reasoning and calculations from Sect. 3.4 and shown in Table 3.24

Table 3.24 t_{worst} value for A5 motorway scenario (4 lanes) with traffic jam, $S_{IW} = 2$

Bit rate (Mbps)	Normal mode (ms)	Continuous mode (ms)
3	400	300
6	300	200
12	300	200

Table 3.25 MATLAB V-FTT parameters

Parameter	Values
Lane width	3 m
Number of lanes	4
Vehicle length	4.58 m
Vehicle spacing average	10/30 m
RSU coverage range	750 m
Safety Zone length	3100 m
Elementary Cycle	100 ms
Modulation	BPSK $\frac{1}{2}$ (3 Mbps)/QPSK $\frac{1}{2}$ (6 Mbps)/16-QAM (12 Mbps)
S_{IW}	2/3
Vehicle speed	Randomly selected between 50 and 120 km/h (constant afterwards)

The main conclusion is that worst-case values are smaller for the A5 motorway scenario and are applicable for the Cooperative Awareness Messages (CAM) defined by ETSI, since the maximum time interval between CAM generations is 1 s (1000 ms). CAM are used for the same purpose as our Basic Safety Message. Still, the worst-case values are above the maximum latency of some of the safety critical applications such as Emergency Electronic Brake Light, as least when using the lowest bit rate. V-FTT guarantees a bounded delay but some scheduling mechanism should be used in order to achieve more reasonable latency values.

3.5.2.2 MATLAB Scenario for A5 Safety Zone 1

In order to evaluate the V-FTT protocol in the A5 motorway, MATLAB was used together with an event generator [5] with the parameters shown in Table 3.25.

The results show the percentage of the Elementary Cycle that is available after the SOW and IW. Choosing the minimum value of that percentage and multiplying by the elementary cycle the results in Table 3.26 ($S_{IW} = 2$) and Table 3.27 ($S_{IW} = 3$) are obtained.

Analysing the results in the previous table it is clear that in all cases all of the OBUs travelling in the Safety Zone are scheduled within one Elementary Cycle.

Table 3.26 Minimum available EC length MATLAB results for Safety Zone 1 (3100 m), $S_{IW} = 2$

Bit rate (Mbps)	Traffic jam (ms)	Normal traffic (ms)
3	66.96	66.92
6	76.16	76.12
12	89.14	89.04

Table 3.27 Minimum available EC length MATLAB results for Safety Zone 1 (3100 m), $S_{IW} = 3$

Bit rate (Mbps)	Traffic jam (ms)	Normal traffic (ms)
3	73.26	72.28
6	80.05	79.42
12	82.78	82.80

Table 3.28 t_{worst} value for A5 motorway scenario with traffic jam, $S_{IW} = 3$

Bit rate (Mbps)	Minimum event dissemination time (ms)	t_{worst} (ms)
3	73	200
6	80	200
12	83	200

Using that information the worst case of event validation time is computed and shown in Table 3.28.

Please keep in mind that the values of t_{worst} are the possible worst case scenario which happens in rare situations.

3.6 Conclusions

In this chapter it was studied how the V-FTT protocol can be applied to the IEEE 802.11p/WAVE standard for safety applications in vehicular environments. Several quantifications were made: the coverage range of an RSU should be 750 m and to ease the handover RSUs should have at least 25 % of overlapping range, meaning that a possible maximum spacing between RSUs is 1312.5 m. Several parameters of the V-FTT protocol were also quantified using a worst case scenario approach, particularly the length of Trigger Messages, Infrastructure Window and a maximum value for the Synchronous OBU Window. The process was done by matching the Elementary Cycle (EC) to IEEE802.11p/WAVE CCH interval and doing calculations made for WAVE normal mode (CCH interval = 50 ms) and WAVE continuous mode (CCH interval = 100 ms) for a worst case where all OBUs need to be served in one EC for two different scenarios: traffic jam and normal traffic. In emergency situations, it might be worth to reduce the Free Period duration to zero for a small amount of time in order to serve more vehicles.

The impact of using a worst case scenario on the ratio of denied transmissions due to the expiry of transmission chance before the maximum tolerable delay for an application was shown, concluding that the V-FTT protocol works well below 450 OBUs in the RSUs coverage area and also that the lower data rate offered by WAVE (3 Mbps) is insufficient for high dense scenarios, which reinforces the option of ITS-G5 of using 6 Mbps and 12 Mbps for safety communications.

It was demonstrated that the V-FTT protocol has a maximum bounded delay and the worst-case delay for transmission of an event (using a fair scheduling mechanism) was analysed, concluding that there is the need for an appropriate scheduling mechanism, since results show that for the worst case in some bit rates the delay is above 300 ms, which is not acceptable for the most demanding safety applications.

A real application scenario, which is the A5 motorway (from Lisbon to Cascais) was presented, along with a possible model for RSU deployment in this motorway. The V-FTT protocol can be used in the A5 motorway, concluding that for peak hour traffic V-FTT still guarantees a bounded delay.

References

1. 802.11-2012—IEEE Standard for Information Technology Telecommunications and information exchange between systems Local and metropolitan area networks Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Technical report IEEE Std 802.11^{TM}-2012. IEEE-Inst. doi:10.1109/IEEESTD.2012.6178212. http://ieeexplore.ieee.org/servlet/opac?punumber=6178209
2. V. Ancusa, R. Bogdan, A method for determining ad-hoc redundant coverage area in awireless sensor network, in *2nd International Conference on Networking and Information Technology, IPCSIT*, vol. 17 (2011)
3. Best Practices for Deploying Polycom® SpectraLink® 8400 Series Handsets White Paper. Technical report POLYCOM, Aug 2011
4. Diário da República. Portaria 1092/97 - Limites de peso e dimenso dos veículos (Código da Estrada). https://dre.pt/application/file/107488.1997
5. D. Dinis, Vehicular flexible time triggered simulator. Technical report, Telecommunications and Informatics, Aveiro University, Department of Electronics, Aug 2013
6. en Wegenbouw en de Verkeerstechniek Centrum voor Regelgeving en Onderzoek in de Grond- Water-.Recommendations for Traffic Provisions in Built-up Areas: ASVV. C.R.O.W. record. Centre for Research and Contract Standardization in Civil Engineering (1998). ISBN:9789066282650. https://books.google.pt/books?id=4aYqAQAAMAAJ
7. ETSI ITS-G5 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. Technical report ETSI (2011)
8. B. Gallagher, H. Akalsuka, H. Suzuki, Wireless communications for vehicle safety: radio link performance and wireless connectivity methods. IEEE Veh. Technol. Mag. **1**(4), 4–24 (2006). http://ieeexplore.ieee.org/xpls/abs%5C_all.jsp?arnumber=4149621
9. Google Maps Search, A5 Motorway Portugal. http://www.google.com/maps.2012

10. T. Guerreiro, Análise da Sinistralidade Rodoviária em Portugal–estudo de duas vias: EN6 e A5, Dissertação para obtenção do grau de Mestre em Engenharia Civil. M.A. thesis. Instituto Superior Técnico, September (2008)
11. IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999) (June 2007), pp. 1–1076. doi:10.1109/IEEESTD.2007.373646
12. IEEE Standard for Wireless Access in Vehicular Environments (WAVE)—Networking Services Corrigendum 1: Miscellaneous Corrections. Undetermined. doi:10.1109/IEEESTD.2012.6239546
13. IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE)—Multi-Channel Operation. IEEE Std 1609.4-2006 (2006). doi:10.1109/IEEESTD.2006.254109. http://dx.doi.org/10.1109/IEEESTD.2006.254109
14. K. Katzis, D.A.J. Pearce, D. Grace, Fixed channel allocation techniques exploiting cell overlap for high altitude platforms, in *The Fifth European Wireless Conference Mobile and Wireless Systems beyond 3G* (2004)
15. A. Perallos, *Intelligent Transport Systems: Technologies and Applications* (Wiley, Chichester, 2015). ISBN:9781118894781
16. Relatório de Tráfego na Rede Nacional de Auto-estradas - 1° *trimestre*. Technical report INIR—Instituto Nacional de Infra-Estruturas Rodoviárias (2010)
17. L. Stibor, Y. Zang, H.-J. Reumerman, Evaluation of communication distance of broadcast messages in a vehicular ad-hoc network using IEEE 802.11p, in *Proceedings of Wireless Communications and Network Conference (WCNC) 2007.* Hong kong, China, Mar 2007, p. 5. ISBN:1-4244-0659-5. http://www.comnets.rwth-aachen.de
18. Vehicle Infrastructure Integration (VII), VII Architecture and Functional Requirements, version 1.0. Technical report, U.S. Department of Transportation, Federal Highway Administration, Apr 2005
19. Q. Xu et al., Vehicle-to-vehicle safety messaging in DSRC, in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks. VANET'04* (ACM, Philadelphia, PA, USA, 2004), pp. 19–28. ISBN:1-58113-922-5. doi:10.1145/1023875.1023879. http://doi.acm.org/10.1145/1023875.1023879

Chapter 4
STDMA-based Scheduling Algorithm for Infrastructured Vehicular Networks

Luis Silva, Paulo Pedreiras, Muhammad Alam and Joaquim Ferreira

Abstract A huge research effort has been devoted to the transportation sector in order to make it safer and more efficient, leading to the development of the so-called Intelligent Transportation Systems (ITS). In ITS there is a closed loop interaction between vehicles, drivers and the transportation infrastructure, supported by dedicated networks, usually referred to as vehicular networks. While some of the enabling technologies are entering their mature phase, the communication protocols proposed so far aren't able to fulfill the timeliness contraints of many ITS services, specially in road congestion scenarios. In order to tackle this issue, several medium access protocols (MAC), either relying on infrastructure or based on direct ad-hoc communication, have been designed. A great number of these protocols employ Time Division Multiple Access (TDMA) techniques to manage communications and attain some degree of determinism. Although the use of spatial reuse algorithms for TDMA protocols (STDMA) has been extensively studied as to increase the efficiency of standard ad-hoc and mesh networks, ITS networks exhibit a combination of features and requirements that are unique and aren't addressed by these algorithms. This chapter (This chapter is an extended work of [21]) discusses some of the most relevant challenges in providing deterministic real-time communications in ITS vehicular networks as well as the efforts that are being taken to tackle them. Focus on TDMA infrastructure-based protocols and on the challenges of employing spatial reuse methods in vehicular environments is placed. A novel wireless vehicular communication architecture called V-FTT, which aims at providing deterministic communications in vehicular networks, is also presented. The chapter concludes with the design of

L. Silva (✉) · P. Pedreiras · M. Alam
Instituto de Telecomunicações, Campus Universitário de Santiago, Aveiro, Portugal
e-mail: luis.silva.ua@gmail.com

P. Pedreiras
e-mail: pbrp@ua.pt

M. Alam
e-mail: alam@av.it.pt

J. Ferreira
Instituto de Telecomunicações, ESTGA-Universidade de Aveiro, Aveiro, Portugal
e-mail: jjcf@ua.pt

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_4

a traffic scheduling analysis, a STDMA slot assignment algorithm and a Matlab simulator for V-FTT.

Keywords ITS · MAC · TDMA · STDMA

4.1 Vehicular Networks

Wireless vehicular networks for cooperative Intelligent Transport Systems (ITS) have raised widespread interest in the last few years due to their potential applications and services. Cooperative applications with data sensing, acquisition, processing and communication provide an unprecedented potential to improve road safety, passengers' comfort and traffic management. In order to support such visionary scenarios, communication between applications deployed in vehicles and applications deployed in the back offices of emergency services, road operators or public services is required. These applications run unattended, reporting information and taking commands from counterpart applications in the vehicle or network.

The European Telecommunications Standards Institute (ETSI) EN 302 665 standard has defined a "Basic Set of Applications" (BSA) which is composed of three main application classes [6, 11]:

- **Road/Traffic Safety Applications**: Aim at reducing the risk of car accidents and at minimizing the damage of unavoidable accidents;
- **Traffic Efficiency Applications**: Aim at improving traffic fluidity;
- **Other Applications (Value-Added)**: Aim at providing comfort and entertainment for the users.

Each of the aforementioned classes exhibit different levels of quality of service (QoS) that must be provided so as to ensure their correct operation. For example, due to their nature, safety related applications typically require maximum guaranteed latencies under 100 ms, while most infotainment applications prioritize high data rates. Besides latency, safety applications also require a high level of determinism and reliability in their communications. For example, a vehicle involved in an accident should be able to access the medium in a timely manner in order to transmit warning messages even in congested road scenarios. Thus, a proper management and design of vehicular networks is essential.

The mobile units of a vehicular network are equivalent to nodes in a traditional wireless network. Besides the ad-hoc implementation of a network consisting of neighboring vehicles, commonly known as Vehicular Ad hoc NETworks (VANETs), in which participants engage in Vehicle-to-Vehicle (V2V) communication, there is also the possibility of a more traditional wireless network setup with base stations along the road sides. These stations act as access points, managing the flow of information of Vehicle-to-Infrastructure (V2I) communications. They can also provide access to external networks and services. Devices that enable the aforementioned types of communication and host ITS applications are called On Board Units (OBUs), when deployed in vehicles, or Road Side Units (RSUs), when deployed in road side stations.

The IEEE 1609 family of standards for Wireless Access in Vehicular Environments (WAVE) defines an architecture and a standardized set of services and interfaces that collectively enable V2X wireless communications. WAVE supports two protocol stacks: the traditional Internet Protocol version six (IPv6), used to accommodate standard IP applications, and a proprietary WAVE Short Message Protocol (WSMP), that accommodates high-priority applications with strict temporal and reliability requirements (e.g. safety-critical applications). Both stacks share the same physical and data-link layer, while differing in the network and transport layers. WAVE's physical and data-link layers rely on the IEEE 802.11-2012 Amendment 6 [20] also known as 802.11p; the equivalent European standard is known as ETSI ITS G5 [12]. IEEE 802.11p adopts the same carrier sense multiple access with collision avoidance (CSMA/CA) mechanism as IEEE 802.11a, as well as IEEE 802.11e Enhanced Distributed Channel Access (EDCA) QoS technique. IEEE 802.11p also introduced simplifications to IEEE 802.11 scanning, association and authentication procedures in order to better cope with the fast topology variations found in vehicular networks [28].

Despite being the emergent standard to enable vehicular communication, IEEE 802.11p presents some issues and limitations that impair its performance and ability to provide reliable and fair communications with timeliness and QoS guarantees. IEEE 802.11p MAC, being reliant on the CSMA/CA technique, exhibits nondeterminism due to the nature of its back-off mechanism. This issue is more severe in heavy traffic scenarios, where unbounded medium access delays and high packet drop rates are frequent. Moreover, since safety messages are transmitted in broadcast mode, no acknowledge (ACK) messages are exchanged to confirm reception and no Virtual Carrier Sensing (VCS) mechanism is employed. This leads to higher collision probabilities in the presence of hidden nodes and decreases communications' reliability [3]. Therefore, IEEE 802.11p does not meet the inherent QoS and reliability requirements for safety applications in vehicular networks, particularly in VANETs. Nevertheless, the probability of collisions occuring may be reduced if the load of the network is kept low, which is difficult to guarantee in vehicular environments, or if a MAC protocol restricts and controls the medium access to provide a deterministic behaviour.

4.2 MAC Protocols for Vehicular Networks

The design of the MAC protocol for vehicular networks is a challenging task since it has to cope with the high mobility and speed of nodes as well as frequent topology changes while being able to support the various QoS requirements demanded by ITS applications. In order to provide a suitable MAC protocol, several key challenges must be addressed [19]:

- **High Speed and Flexibility**: The implemented mechanisms should be able to cope with the high speed of nodes and seamlessly adapt to frequent network topology changes. They must be able to operate in both highway and urban scenarios;
- **Scalability**: MAC protocols should provide efficient channel utilization mechanisms under distinct traffic load conditions and network sizes;
- **Broadcast Support**: Efficient broadcast services for the dissemination of information within a regional scope are required (e.g. for safety-related applications);
- **Hidden and exposed nodes**: Due to the shared nature of the communication medium, the MAC protocol should account for the possible existence of hidden/exposed nodes and employ techniques to avoid or decrease the probability of collisions caused by these nodes;
- **QoS Support**: MAC protocols should provide suitable transmission services to enable the support of applications with strict QoS requirements, i.e. reliable communications with bounded delays, and at the same time, ensure high throughput for bandwidth demanding applications, e.g. infotainment;

The design of a scalable, deterministic MAC protocol for vehicular communications can follow two main possible approaches. It can either rely on infrastructure, or it can be based on ad-hoc vehicle to vehicle communication only, without relying on any infrastructure. A hybrid approach that takes advantage of both models can also be pursued. Strict real-time behaviour and safety guarantees are typically difficult to attain in ad-hoc networks, but they are even harder to achieve in high speed mobility scenarios, where the response time of distributed consensus algorithms, e.g. for cluster formation and leader election, may not be compatible with the dynamics of the system. Therefore, the presence of an infrastructure, e.g. road-side units and a backbone cabled network, may add a degree of determinism that is useful to enforce real-time and dependability at the wireless end of the network.

The obvious advantage of vehicular ad-hoc solutions is that no infrastructure is required and thus, they are cheaper and easier to deploy, making them attractive for rural areas and developing countries. Moreover, the latency on these networks is, in principle, lower than on an infrastructure-based solution since the communication is directly from source to destination [34]. However, ad-hoc solutions present some strong disadvantages in what concerns safety applications [5, 25]:

- In order to relay a message to the destination multi-hopping might be required, leading to an increase of the end-to-end delay;
- To enforce determinism in V2V communications, different flavours of distributed consensus (e.g. membership and leader election) are required. These typically add complexity and consume extra resources;
- Badly intended users can broadcast false information that cannot be mediated/validated by the infrastructure;
- Alarm showers, also known as broadcast storm, can occur, overloading the medium, unless some protocol is enforced to avoid that situation.

Despite the higher deployment costs of infrastructure-based solutions, these exhibit several particularities which can be exploited in order to gain benefits, especially for high density and vehicle speed scenarios:

- The received vehicle data can be analysed by the infrastructure, which can then edit and validate the reported events by cross-examining with other sources of information (e.g. cameras, induction loops.). This minimizes vulnerability issues;
- The access to the medium can be managed by the infrastructure, preventing collisions and avoiding the aforementioned broadcast storm issue;
- Infrastructures and/or entities that coordinate them can have a global vision of the covered area and therefore, make better decisions;
- Processing overheads can be confined to the infrastructure, reducing the complexity and costs of OBU equipments.

Moreover, the road-side infrastructure can be seen as an instantiation of the *wormhole* metaphor [35] where it is assumed that uncertainty is neither uniform nor permanent across all system components i.e. some parts are more predictable than others. In this way, the more predictable parts of the system can be seen as *wormholes* since they will execute certain tasks faster or more reliable than apparently possible in the other parts of the system. Thus, it can be argued that I2V communications could be made safer than pure V2V as the presence of the infrastructure, i.e. road-side units and a backbone cabled network, adds a degree of determinism needed to enforce real-time and safety at the wireless end of the network. To secure determinism at the backbone cabled network, real-time network technology, usually at layer 2, or resource reservation protocols to extend the guarantees to multiple networks and higher layers, are required. These technologies are out of scope of this chapter.

MAC protocols generally fall into one of two broad categories [19]: contention-based and contention-free. In contention-based protocols, each node tries to gain access to the medium using carrier sense mechanisms, while contention-free protocols assign access to the channel to a single node at any given time. In contention-based protocols, such as on IEEE 802.11p, multiple neighboring nodes can sense the medium as free and transmit at the same time, causing collisions at the destination nodes and impairing the real-time performance of the vehicular network. Contention-free protocols can provide bounded-delays and better QoS support at the cost of complexity: they usually require the exchange of periodic control messages between all nodes in the network to maintain synchronization and build/update scheduling tables.

The design of contention-free protocols can be based on three main techniques [19]: Frequency Division Multiple Access (FDMA), Code Division Multiple Access (CDMA) and Time Division Multiple Access (TDMA). These protocols allow vehicles to access the medium using a unique frequency band, code sequence or time slot respectively, avoiding collisions between vehicles in the same two-hop neighborhood.

FDMA-based protocols are typically complex and exhibit high communication overheads since they require frequency synchronization mechanisms at the transmitter and receiver. In order to negotiate frequencies, these mechanisms rely on the exchange of control messages using a dedicated frequency control channel. In contrast, CDMA-based protocols share the same frequency channel between different vehicles, however, communication between a given pair/set of vehicles is encoded using unique code sequences. An algorithm to negotiate and assign codes for every communication is required, increasing transmission delays and adding overheads. TDMA-based protocols share the same frequency without any coding but assign different temporal slots to each vehicle. I2V/V2I communications can be efficiently supported by TDMA-based protocols, with the infrastructure creating, managing and disseminating the slot reservation schedules. Moreover, slots can be managed online and in a flexible way; for example, several different time slots can be assigned to the same vehicle to increase bandwidth or redundancy. The infrastructure can also predict vehicles' movement and position and assign the same slot to vehicles that aren't within the interference range (spatial reuse). Due to its attractive characteristics, the use of TDMA-based solutions on vehicular networks is currently an emerging area of research.

The remaining of the chapter will place focus on infrastructure TDMA-based MAC protocols due to their importance in vehicular networks as well as their relatedness to our proposal.

4.2.1 Infrastructure TDMA-based Deterministic MAC Protocols

As previously discussed, TDMA-based MAC protocols are currently under a great focus by the research community. This section presents a discussion and analysis of the current, most relevant proposed protocols in the infrastructure TDMA-based category.

Guo et al. [17] proposed an Adaptive Collision-Free MAC (ACFM) for vehicular networks, based on a centralized dynamic TDMA mechanism. In ACFM, time is divided by the RSUs into cycles comprising several frames; each frame contains one RSU slot (RS), used to broadcast control messages and slot schedules to vehicles, and 36 data slots (DS) which can be used by vehicles to broadcast data. RSUs dynamically assign DS to vehicles under their coverage and use a cycle length expansion and shrinking mechanism to adjust the number of frames within a cycle according to traffic density. An extension scheme of ACFM, named Risk-Aware Dynamic MAC Protocol for Vehicular Cooperative Collision Avoidance System (R-MAC), was later on proposed by the authors [18]. This extension added a contention-based segment for the transmision of warning messages in emergency situations. Although these proposals show improvements in the average access delay and packet loss ratio when compared to IEEE 802.11p, they present some drawbacks such as being limited to periodic messages and requiring two-hop neighboring RSUs to operate in different frequencies to avoid interference.

Zhang et al. [38] proposed a Unified TDMA-based Scheduling Protocol (UTSP) for V2I communications with the goal to optimize throughput for non-safety applications. In the proposed solution, RSUs collect information regarding the channel quality, speed and Access Category (AC) of the vehicles under their coverage area.

RSUs then assign time slots to vehicles by taking into account weights derived from the previously collected information. The channel quality is used to maximize throughput, the speed factor is used to ensure fair access to the medium for vehicles with different speeds, and the vehicle AC is used to distinguish access priorities. Simulations show that UTSP has good throughput and fairness performance when compared to IEEE 802.11p, however it was only evaluated for one RSU; the effects of interference between vehicles in overlapping regions are not considered and explained.

Several authors [1, 4, 24] have proposed deterministic Medium Access Control (MAC) schemes for V2I communications by extending the IEEE 802.11-A6 [20] commonly known as IEEE 802.11p standard. They introduced a collision-free phase in which a coordinator, in this case a Road Side Unit (RSU), takes the responsibility of scheduling the data traffic and polling the mobile nodes. In this way the channel is assigned to each vehicle equipped with an On Board Unit (OBU) for a specific period of time and real-time data traffic is scheduled in a collision-free manner by each RSU.

Böhm and Jonsson [4] assign each vehicle an individual priority based on its geographical position, its proximity to potential hazards and the overall road traffic density. This is done by introducing a real-time layer on top of the normal IEEE 802.11p. A super frame is created in order to obtain a Collision Free Phase (CFP) and a Contention Based Period (CBP). In the CFP, RSUs assume the responsibility of scheduling the data traffic and polling mobile nodes for data. Vehicles then send their heartbeats with position information and additional data (such as speed, intentions, etc.). RSUs periodically transmit a beacon to mark the beginning of a super frame, stating the duration of the CFP, so that each vehicle knows when the polling phase ends and when to switch to the regular CSMA/CA from IEEE 802.11p, which is used in the CBP. The length of CFP and CBP is variable. Real-time schedulability analysis is applied to determine the minimum length of CFP such that all deadlines are guaranteed. The remaining bandwidth is used for best-effort services and V2V communications. In order for RSUs to start scheduling vehicle transmissions, vehicles must register themselves by sending out connection setup requests (CSR) as soon as they can hear an RSU. This is done in the CBP, so a minimum risk exists of vehicles failing to register. They can, however, receive information from RSUs and communicate using the CBP. Böhm refers that vehicles might want to increase the number of heartbeats sent during lane change or in certain risk areas, but this is not clearly explained. Another interesting issue is that a proactive handover process is defined, based on the knowledge of road path and RSUs locations. Nothing is mentioned about RSU coordination and how it is done.

Bohm's protocol has many similarities with Tony Mak et al. [23] who proposed a variant to 802.11 Point Coordination Function (PCF) mode so that it could be applied to vehicular networks. A control channel is proposed in which time is partitioned into periodic regulated intervals (repetition period). Each period is divided into a contention free period also named CFP by the author (with the same meaning as Collision Free Phase used by Böhm) and an unregulated contention period (CP). The scheme is similar to Böhm's, where each vehicle is polled by an RSU or Access

Point (AP) during the CFP, similarly to the PCF of a regular IEEE 802.11. Vehicles need to register and deregister so the polling list is kept updated. For this purpose a group management interval is created so that vehicles entering and leaving the region can notify the RSU. However this beacon is sent in the CP and contends with other communications. The authors propose that the beacon is repeated to decrease the probability of reception failure of the beacon. No schedulability analysis is made in [23] but the authors claim that the time between consecutive polls for vehicles in the RSU coverage area is bounded.

Meireles et al. [1, 24] proposal has a lower overhead compared to the individual pool-reply scheme adopted in [4] as it transmits the schedule of multiple OBU transmissions in a single RSU message. Meireles et al. [24] protocol, called the vehicular flexible time-triggered (V-FTT), adopts a master multi-slave time division multiple access (TDMA) in which the road-side units act as masters and schedule the transmissions of the on-board units. This protocol has some interesting properties like dynamic online scheduling in which there is the possibility of adopting multiple scheduling policies, strict event and time-triggered traffic isolation and online admission control. An overview of V-FTT and discussion of its mechanism is provided in Sect. 4.3.

4.2.1.1 Spatial TDMA

Real-world scenarios often require the use of multiple RSUs, not only to cover wide geographical areas but also to cover the existence of natural or artificial obstacles that usually block or hinder communications. Since the density of vehicles is usually high in urban areas, it is also important to use the available bandwidth in an efficient manner. Spatial reuse of communication slots in wireless TDMA architectures (STDMA), first proposed in [26], has been studied extensively in wireless mesh and ad hoc networks, including scenarios in which real-time service guarantees are sought. The underlying idea of STDMA is to increase the communication capacity by permitting, when possible, concurrent transmissions in the same slot as issued by different nodes. The slot sharing is possible when nodes are geographically separated and the resulting interference is small.

STDMA scheduling algorithms can be categorized into link or broadcast/node scheduling algorithms [31]. In link scheduling, algorithms assign time slots to certain links, i.e. a communication flow between a transmitter and a receiver, while node algorithms assign slots to individual nodes. Link scheduling is suitable for unicast traffic, while node scheduling is better suited for broadcast traffic.

The basic approach of node-based schedulers for building a conflict-free TDMA schedule starts by identifying nodes and their interfering range as well as the temporal characteristics of their transmissions. Different time slots are then assigned to identified conflicting nodes in order to prevent interference. In order to increase efficiency and reduce the number of necessary slots, the scheduler can assign a given time slot to several non-conflicting nodes. In contrast, link-based schedulers start by identifying the links between every source and destination nodes, as well as the communication requirements (number, size and temporal characteristics of the messages). Then, by using appropriate channel information, the interfering/conflicting links are identified. Afterwards an appropriate algorithm is used to generate the TDMA schedules subject to some specific criteria (e.g. minimizing the TDMA cycle, minimizing the lateness, etc.). Since the number of possible link patterns can be significantly larger than the number of nodes in the network, link-based schedulers can exhibit a much higher complexity. Moreover, the resulting TDMA frame can be too large to be efficiently implemented/supported by the network.

Although the aforementioned STDMA schedulers are more or less simple in concept, the problem of obtaining optimal STDMA schedules that meet specific application constraints is far from trivial. In fact, the minimization of the TDMA cycle in packet radio networks was proven to be an NP-Complete problem for both node and link-based approaches [14, 31].

A plethora of STDMA-based schedule generating algorithms, both centralized and distributed in terms of coordination, can be found in literature. For example, in [15], Funabikiy and Takefuji proposed a centralized algorithm based on neural networks while Pond et al. [29] proposed a distributed protocol by considering multi-hop TDMA broadcast packet radio network. Lloyd et al. [30] presented an algorithm aiming at generating minimum schedules by considering both the link and node scheduling cases. Hafeez et al. [8] proposed a "high spatial-reuse distributed slot assignment protocol" in which nodes compute the slot assignment based on local topological data. Nodes are granted slots according to their number of neighbors thus implementing a priority scheme.

More recently there was an effort to address more realistic channel models. Two categories are usually considered: protocol interference model and physical interference model. In the former model the communication between two nodes, i.e. *s* (sender) and *r* (receiver), is considered successful if there are no concurrent transmissions in a predefined interference range of node *r*. The physical interference model is more realistic since it considers a successful transmission only if the Signal to Interference and Noise Ratio (SINR) at the receiver *r* is above a certain threshold. This model is more realistic since it accounts for the interference of several nodes in the channel but at the expense of greater complexity. Xue et al. [36] proposed a greedy algorithm for link scheduling by considering the physical interference model, which improves the greedy approach presented in [7]. Gore et al. [16] proposed a STDMA link-based scheduler for ad hoc networks based on a graph model of the network as well as SINR computations. A node-based slot assignment for STDMA mesh networks based on a SINR channel model is proposed by Chen and Lea [9].

Another recent research line that has recently drawn the attention of the scientific community is the link allocation routing and scheduling in hybrid networks. For instance, in [32] the authors proposed a methodology to optimize the throughput by placing free space optics links at appropriate places and deriving the routing and schedules in an integrated manner.

As discussed above, STDMA is a deeply studied subject with abundant results. However, relevant literature on resource scheduling for vehicular networks is scarce to date. Recent research on this area include the work of Yu et al. [37], which proposed and simulated a decentralized self-organizing TDMA algorithm for V2V communications in highway scenarios. Although not a pure STDMA algorithm, it employs a slot reuse mechanism for distant vehicles that is triggered when no free slots are available. Zhang et al. proposed a weighted, centralized TDMA scheduling algorithm for V2V communication links. The scheduling weight factor depends on perceived channel quality and vehicle's speed and access priority. Slot reuse for distant vehicles is also applied.

However, the V2I/I2V scenario addressed in this chapter has a combination of features and requirements that are unique and distinctive. More specifically:

- Due to the fast mobility of vehicles, the schedule must constantly be updated so a computationally complex algorithm cannot be used;
- It is also possible to predict, to some extent, the position of vehicles in near future. Such information can be used in the schedule update;
- The system has global knowledge. Different RSUs share information in real-time. The system may perform both in a distributed or centralized manner;
- The communication is exclusively broadcast and single-hop;
- The scheduling is carried out over nodes, not links;
- Messages have distinct priorities;
- The size of all messages and timeslots is equal.

None of the research work found so far in the literature addresses such combination of issues in an integrated manner, thus opening a way for further research in the context of infrastructure-assisted wireless vehicular networks.

4.3 V-FTT Protocol Overview

Recently, a proposal for deterministic medium access control (MAC) for vehicular environment was presented in [1, 24] called the "Vehicular Flexible Time-Triggered (V-FTT) Protocol". This protocol adopts a multi-master multi-slave spatial time division multiple access (STDMA) in which road side units act as masters and schedule the transmissions of on-board units. As depicted in Fig. 4.1, the protocol is divided into periodic elementary cycles (ECs) where each EC starts with an infrastructure window (I2V) containing trigger and warning messages.

The V-FTT protocol inherits most of its concepts from the original Flexible Time-Triggered protocol definition [2], while presenting some new features to it to cope with the wireless vehicular scenario. In particular, it adopts redundant scheduling for OBUs transmissions to increase reliability and to cope with the variations of the propagation patterns of the RSUs caused by atmospheric and traffic conditions. According to the proposed redundant scheduling scheme, a single OBU is scheduled by a configurable number of RSUs for the same transmission slot. As RSUs cooperate to schedule OBUs safety communications, they must be able to coordinate their own transmissions, avoiding possible mutual interferences. To support RSU coordination, it is assumed that they are fully interconnected by a back-hauling network. It is also

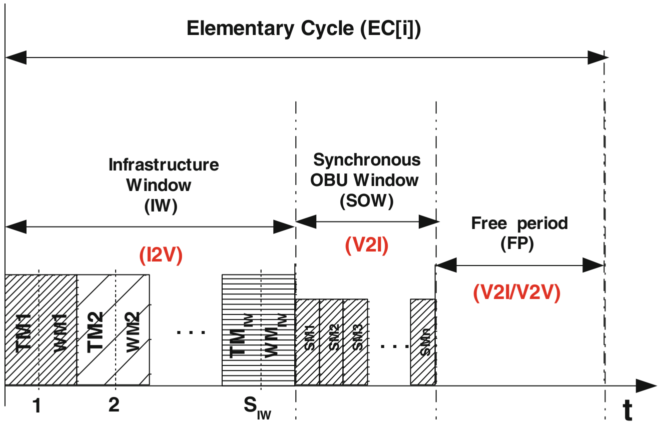


Fig. 4.1 Elementary cycle of vehicular flexible time-triggered protocol [24]

assumed that RSUs are able to receive messages from vehicles traveling in both directions and that vehicles can receive messages from various adjacent RSUs.

Each RSU will transmit its Trigger Message (TM) in its transmission slot to schedule the OBUs transmission slots, using just one message. This scheme is known as master multi-slave, as a single master (RSU) message triggers the transmission of a number of slave (OBU) messages as opposed to the traditional master-slave in which each master message triggers just one slave reply. As a configurable number of RSUs cooperate to redundantly schedule the transmissions of the same OBU, it can be said that V-FTT adopts a multi-master multi-slave spatial TDMA. In this RSU coordination proposal, RSUs transmit the OBUs scheduling in a reserved window called the Infrastructure Window. Within this window time slots are reserved for each RSU. As RSUs are synchronized, they are able to respect the time slot boundaries.

The infrastructure window is followed by the synchronous OBU window where OBUs have the opportunity to transmit information to RSUs (V2I). Each OBU will have a fixed size slot to transmit vehicle's information (speed, acceleration, heading, etc.) and/or a safety event. The Synchronous OBU Window duration is variable. The elementary cycle ends with an optional free period window, a period where non V-FTT enabled OBUs are able to transmit safety messages and RSUs and OBUs are able to transmit non-safety messages.

In V-FTT, roadside units are responsible for two main operations:

- To schedule the transmission instants of vehicle OBUs in what concerns the safety frames they have to broadcast;
- To receive information from OBUs, edit that information and publish the edited safety information in the adequate places and instants (might be a broadcast or might be a communication to selected vehicles).

From the communications point of view, the OBUs must:

- Listen to the RSU transmissions (at least one RSU should be heard), retrieve the safety information and dispatch this information;
- Always transmit its specific safety frame in the time slot allocated by RSUs.

Constant sized Cooperative Awareness Messages (CAM) [13] are transmitted during the Synchronous OBU Window. CAM messages are broadcast messages that include several possible data elements (e.g., CrashStatus, Dimension, Heading, Latitude, Longitude, Elevation, Longitudinal Acceleration, Speed). CAM messages are transmitted periodically and have strict timing requirements. They are generated by the CAM Management service and passed to lower layers according to some set of rules [13] which are checked every 100 ms. A CAM message is dropped whenever the channel access request does not result in actual channel access before the next message is generated. There will be temporary reduction in the performance efficiency of the application if a periodic message misses its time limit.

Non-registered OBUs will also receive safety information from RSUs. However, they are not able to transmit information according to the proposed protocol, although they can still contend for transmission during the free period, but without any guarantees.

The information broadcasted by the RSUs must be trustworthy, thus, RSUs must validate OBUs' events before being broadcasted to vehicles. This validation must obviously be performed in bounded time so that the results could be transmitted to the OBUs in real-time.

Road segments covered by RSUs running the V-FTT protocol are called Safety Zones (SZ). Whenever a vehicle enters a SZ, it registers itself with the infrastructure so that RSUs can assign an identifier to each vehicle (OBU) and schedule their transmissions. The responsibility of scheduling vehicles moving along a road equipped with a roadside infrastructure is passed from RSU to RSU in a cooperative and distributed way. This handover process must also be dependable and timely.

4.4 Traffic Scheduling

The deployment of safety wireless vehicular communications in the scope of ITS applications supported by roadside back-hauling networks requires an end-to-end deterministic behaviour. For example, a vehicle involved in an accident should be granted timely access to the wireless medium to transmit a safety message, which once validated by the roadside infrastructure, should trigger the timely transmission of warning messages to other vehicles approaching the accident site. Therefore, it is important to have a proper scheduling of the communication channel to allow critical information to be transmitted with minimum latency. Moreover, the vehicles close to the accident or driving in its direction could be within the coverage area of different RSUs, each enforcing a real-time MAC protocol for the vehicles in their coverage area. In this scenario the responsibility of scheduling the vehicles' transmissions is passed to nearest RSU, thus requiring a deterministic handover to extend the local (RSU) real-time guarantees to the whole roadside infrastructure.

V-FTT provides a deterministic MAC protocol to support the real-time guarantees as mentioned above. The next sections present an analytic framework that enables managing the traffic appropriately.

4.4.1 Problem Statement and System Model

For the system model, a geographical region covered by one or more RSUs is considered, in which RSUs are interconnected by a deterministic real-time network. Vehicles that are managed by the system integrate an OBU. Vehicles traveling in urgent mission such as ambulances, police and fire fighters receive privileged access. Moreover, vehicles that are involved in accidents and/or that report information about accidents or other abnormal events also receive privileged access to the communication channel. The temporal validity of the information is variable, therefore there is an associated deadline.

More formally, the system under consideration can be described as follows:

$$R = \{RSU_1, RSU_2, \ldots, RSU_N\}, N \geq 1 \tag{4.1}$$

R is the set of N RSUs that cover some geographical region A. Each RSU covers a given sub area A_i, such that $A = \{A_1 \cup A_2 \cup \cdots \cup A_n\}$. Due to the high dynamics of the system (vehicles may travel at a relatively high speed) and the high number of vehicles that may have to be managed, a simpler protocol interference model is adopted instead of a more accurate, but complex, physical interference model. Consequently, in scenarios with multiple RSUs, it is possible to define a binary matrix I, which defines the interference ranges, as illustrated in Table 4.1. This table assumes the existence of 4 RSUs with a homogeneous range, in which each cell may interfere with the adjacent ones. Thus, $I(i, j) = 1$ if a vehicle in area A_i, covered by RSU_i may interfere with a vehicle in area A_j, covered by RSU_j, and vice-versa.

The diverse sub areas overlap partially meaning that OBUs may be in the communication range of more than one RSU. OBUs have an individual and unique identifier (ID) and send CAM messages that contain a system-wide fixed amount of data (W bytes), which take C seconds to transmit.

Table 4.1 Interference range matrix

R4	0	0	1	1
R3	0	1	1	1
R2	1	1	1	0
R1	1	1	0	0
RSU	R1	R2	R3	R4

The V-FTT protocol is configured with a fixed EC duration of $LEC = 100\,ms$, which corresponds to synchronization interval of IEEE 1609.4. The SOW window is configured to accommodate up to S safety messages, each one assigned to a slot of size C. Slot size and message size are equal to impede the transmission of alien (i.e. non V-FTT) messages during the SOW.

The set of nodes in the covered area generates a message set M as follows:

$$M = \{m_i, m_i = \{ID_i, X_i, C, T_i, P_i, D_i\}\}, i = 1 \ldots O \tag{4.2}$$

ID represents the unique OBU identifier, $X_i \in A_i$ the vehicle position, T_i the message periodicity, P_i the message priority, D_i the deadline and O is the number of OBUs in the system. Note that T_i, P_i and D_i are the dynamic parameters that are managed by the system according to the vehicle conditions and the overall system load.

4.4.2 Basic Problem—Single RSU

The model used to schedule synchronous OBU messages (CAM) traffic in V-FTT is very similar to the one presented in [2]. According to this model message periods and deadlines are integer multiples of a basic cycle duration (LEC) where message transmission times are shorter than LEC. Message activations are always synchronous with the start of the cycle and the synchronous traffic is confined to a sub-window of the EC with maximum length $L = S * C$.

As shown in [2], a simple technique to model the effects mentioned above is to inflate the message transmission times by a factor equal to $\frac{LEC}{L}$, which is equivalent to expanding the SOW up to the whole EC. Applying this transformation to the original message set results in a new virtual set (M^v) as defined in Eq. 4.3, where all the remaining parameters except the transmission time are kept unchanged. Since all CAM messages have size C, this adaptation only requires one simple computation carried out once, independently of the number of vehicles in the area.

$$M^v = \{M_i, M_i = \{ID, X_i, C^v, T_i, P_i, D_i\}\}, i = 1 \ldots O, C^v = C * \frac{LEC}{L} \tag{4.3}$$

CAM messages coming from priority vehicles are more important and are transmitted as often as possible. Thus, there is a direct association between the priority of messages and its rate, resulting in the adoption of an implicit Rate-Monotonic priority assignment. This observation, together with the transformation shown in Eq. 4.3, allows the use of the simple Liu & Layland utilization test [22], indicated in Eq. 4.4.

$$\sum_{i=1}^{O} \left(\frac{C^v}{T_i} \right) < O(2^{\frac{1}{O}} - 1) \tag{4.4}$$

With the same adaptation the use of some other eventually more exact schedulability tests such as Response Time Analysis is also possible. The adaptation is fairly standard and is shown in Eqs. 4.5 and 4.6. The absence of a blocking term due to the synchronous activation of CAM messages is noticeable.

$$R_{wc_i} = I_i + size(IW) + C^v \tag{4.5}$$

$$I_i = \sum_{k \in hep(i)} (\lfloor \frac{I_i}{T_k} \rfloor + 1) * C^v \tag{4.6}$$

As usual, Eq. 4.6 is iterated until convergence ($I_i^j = I_i^{j-1}$) or until a deadline is violated ($R_{wc_i} > D_i$). The term $size(IW)$, present in Eq. 4.5, models the fact that CAM messages are confined to the SOW window, which always follow the IW in EC.

The worst-case response time for events can be deduced from the message worst-case response time as computed in Eq. 4.5. Events are generated by the environment and are asynchronous with respect to the V-FTT network. Therefore, any event that happens after the transmission of the CAM message of the associated node has to wait for the activation of the next CAM message before entering in arbitration. This delay corresponds to a dead time designated as σ_i^{dead}. Then, the node has to wait a given number of ECs, due to interference, as computed in Eq. 4.5. Finally, in the last EC it may be transmitted in any slot within the SOW. Thus, the worst case delay happens when it is assigned the last slot(σ_i^{block}). Figure 4.2 illustrates the origin of the dead and block times for a simple scenario in which the CAM message has a period of one EC. Therefore, the computation of the worst-case response time of an event j associated with a node i is given by Eqs. 4.7, 4.8 and 4.9.

$$Rwc_i^j = \sigma_i^{dead} + w_i + \sigma_i^{block} \tag{4.7}$$

$$\sigma_i^{dead} \leq LEC - size(IW) + T_i - 1 \tag{4.8}$$

$$\sigma_i^{block} \leq size(IW) + L \tag{4.9}$$

The worst case dead time happens if the transmission of the CAM message is carried out in the first slot of the first EC after the CAM message activation. In this

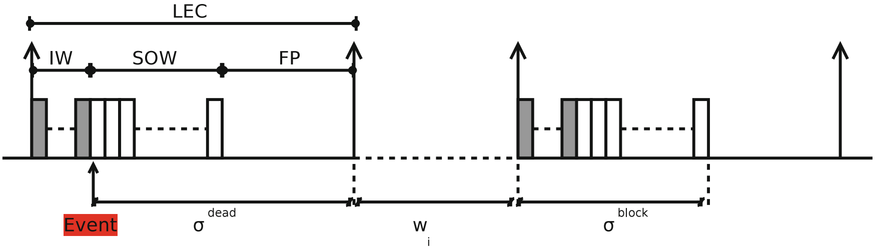


Fig. 4.2 Event delay components

case the event has to wait part of the first EC ($LEC - size(IW)$) plus the remaining ECs ($T_i - 1$), as indicated in Eq. 4.8.

In reality, CAM messages are confined to the SOW window. Thus, Eq. 4.5 is pessimistic as it allows CAM messages to be scheduled at any point of the EC due to the use of inflated message times. To reduce this pessimism, it is possible to use Eq. 4.5 to compute the integer number of ECs due to interference from high or same priority messages (w_i). The worst case delay, suffered by the CAM message in the last EC, happens if this message is the last one of the SOW, a value upper bounded by Eq. 4.9. The integer number of ECs due to the interference is given by Eq. 4.10.

$$w_i = \lfloor \frac{Rwc_i}{LEC} \rfloor * LEC \tag{4.10}$$

Both methods above allow the schedulability evaluation of a message set and thus, can be used for admission control.

4.4.3 Realistic Scenario—Multiple RSUs

As previously mentioned, due to the high dynamics of the system and the potentially high number of vehicles that must be managed concurrently, it is mandatory to select expedite techniques suitable for real-time operation. For this reason, the interference model is adopted instead of the physical interference model, which is more accurate but far more complex to manage.

The network is represented by a graph $G = (V, E)$ such that V represents the vertices (nodes i.e., OBUs and RSUs) and E is the set of edges (wireless links between OBUs and RSUs). It is assumed that all edges are bidirectional, thus, for any two distinct vertices $\{i, j\} \in V$, $\{i, j\} \in E$ if i and j can communicate with each other.

As nodes are the basic entities of the system a node assignment scheme is adopted. The objective is to derive a schedule that will guarantee a conflict free node transmission. A slot reuse mechanism is also adopted in order to increase the number of vehicles in the system and to make an efficient use of resources, i.e. communication medium.

Let Z be defined as an arbitrary set of nodes. $\Psi(z)$ defines the set of logical neighbours of node $z \in Z$ that reach the same RSU as node z is registered with, and thus, are able to generate a transmission conflict, i.e. collision. A necessary condition to permit the allocation of the same slot to the nodes in Z is:

$$Z \cap \Psi(Z) = \phi \tag{4.11}$$

That is, no two nodes in Z can reach the same RSU. The computation of $\Phi(Z)$ is easily carried out from the Interference matrix as defined in Sect. 4.4.1, and from the message set M (Eq. 4.2), which define respectively the conflicts between adjacent areas and the area in which each vehicle is positioned.

As mentioned in Sect. 4.2.1.1, the problem of finding a slot allocation is NP-Complete. There are some heuristics in the literature but they are not directly applicable to this scenario, as they don't address multi-hop networks nor application-defined priorities. They also aren't able to produce useful results in scenarios of very high mobility. For these reasons, an algorithm that takes into consideration these constraints is hereby presented.

Algorithm 1 takes as its inputs a set of system configuration parameters (set of RSUs, maximum number of CAM slots in the SOW and the interference matrix) and the dynamic state of the system as defined by the message set. The output of the algorithm is a matrix (*Sched* matrix) that contains the set of message IDs corresponding to CAM messages that shall be scheduled by each RSU in the following EC. Due to the possibility of slot reuse, *Sched* takes a matrix form since its contents may vary from RSU to RSU.

Algorithm 1 Slot Assignment for Multiple RSUs

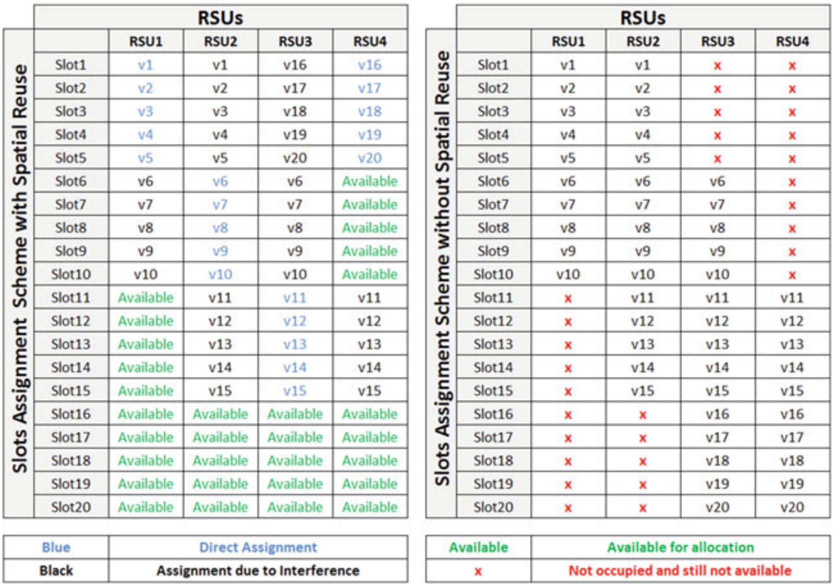
```
Inputs
R: set of RSUs
M: message set
S: maximum number of slots on SOW
I: interference matrix

Outputs
Sched[R,S]: array of CAM IDs (EC schedule)

Sched=$\phi$
Sort M by priority
For each message $m_i \in M$
   if $m_i$ is ready and $d_i \le D_i$
      $R^u = R^{main} \cup R^I$
      For each $s \in [1 : S]$
         Found = TRUE
         For each $r \in R^u$
            If $Sched[r, s] \ne FREE$
               Found = FALSE
            endIf
         endFor
         If Found == TRUE
            For each $r \in R^u$
               $Sched[r, s] = ID_i$
            endFor
            Break
         endIf
      endFor
      If Found == FALSE return NON_SCHED
   endIf
endFor
```

The first step is to sort the messages by priority (line 2) assuring that the highest priority nodes are privileged. Then each ready message m_i is processed (lines 3,4). The interference range (R_u) of each message m_i is then defined (line 5). Afterwards, the availability of slots in the adjacent interfering RSUs is identified (lines 7–12). If a slot is available it is allocated to the message m_i (lines 13–16) otherwise a NON-SCHED message m_i is returned (line 20). It should be noted that the above algorithm executes concurrently in all RSUs. However, since the algorithm is deterministic, decisions will be consistent as long as the system state, as viewed by the different RSUs, is kept consistent and the scheduling is carried out synchronously.

For illustration purposes, let us consider the scenario of four RSUs presented in Table 4.1. Consider also that each RSU currently has five vehicles in its coverage area so the total number of nodes are $(V_1 \ldots V_{20})$. Without spatial reuse it would be necessary to use a total of 20 slots (one per vehicle/CAM message) in each RSU.

Assuming, without loss of generality, that priorities are $Pr(V_1) \geq Pr(V_2) \cdots \geq Pr(V_{20})$. The output of Algorithm 1 results in the slot assignment depicted in Fig. 4.3.

As can be seen the number of assigned slots was reduced significantly. RSU{1,4} use only 10 slots each while RSU{2,3} use only 15 slots each. Thus the bandwidth used is reduced from 25 to 50 % making it possible to accommodate more vehicles into the system and/or free bandwidth for other traffic classes and ultimately adding to the overall system capacity and throughput.



Slots Assignment Scheme with Spatial Reuse

	RSUs			
	RSU1	RSU2	RSU3	RSU4
Slot1	v1	v1	v16	v16
Slot2	v2	v2	v17	v17
Slot3	v3	v3	v18	v18
Slot4	v4	v4	v19	v19
Slot5	v5	v5	v20	v20
Slot6	v6	v6	v6	Available
Slot7	v7	v7	v7	Available
Slot8	v8	v8	v8	Available
Slot9	v9	v9	v9	Available
Slot10	v10	v10	v10	Available
Slot11	Available	v11	v11	v11
Slot12	Available	v12	v12	v12
Slot13	Available	v13	v13	v13
Slot14	Available	v14	v14	v14
Slot15	Available	v15	v15	v15
Slot16	Available	Available	Available	Available
Slot17	Available	Available	Available	Available
Slot18	Available	Available	Available	Available
Slot19	Available	Available	Available	Available
Slot20	Available	Available	Available	Available

Blue	Direct Assignment
Black	Assignment due to Interference

Slots Assignment Scheme without Spatial Reuse

	RSUs			
	RSU1	RSU2	RSU3	RSU4
Slot1	v1	v1	x	x
Slot2	v2	v2	x	x
Slot3	v3	v3	x	x
Slot4	v4	v4	x	x
Slot5	v5	v5	x	x
Slot6	v6	v6	v6	x
Slot7	v7	v7	v7	x
Slot8	v8	v8	v8	x
Slot9	v9	v9	v9	x
Slot10	v10	v10	v10	x
Slot11	x	v11	v11	v11
Slot12	x	v12	v12	v12
Slot13	x	v13	v13	v13
Slot14	x	v14	v14	v14
Slot15	x	v15	v15	v15
Slot16	x	x	v16	v16
Slot17	x	x	v17	v17
Slot18	x	x	v18	v18
Slot19	x	x	v19	v19
Slot20	x	x	v20	v20

Available	Available for allocation
x	Not occupied and still not available

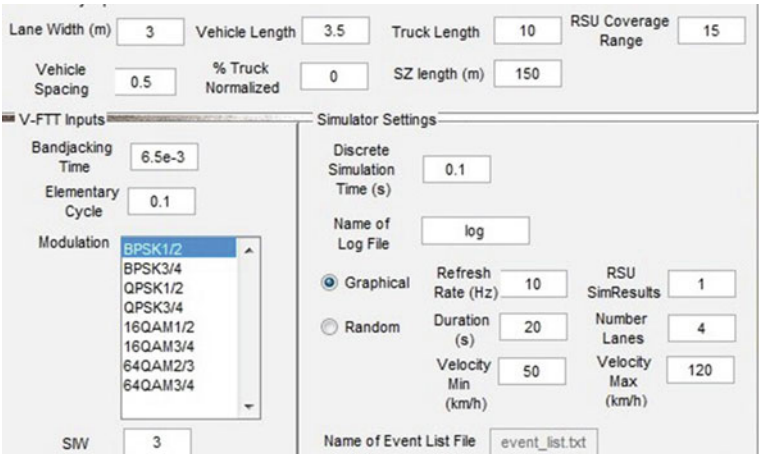
Fig. 4.3 Scheduler slot assignment output with and without spatial re-use

4.5 Development of a New Scheduling Simulator

A number of software tools for network simulations, such as SUMO [33], ns-3 [10] and iTETRIS [27], are nowadays available as free to use, open source tools. However, using the aforementioned tools for the development of a scheduling algorithm for an infrastructure-based ITS network running the V-FTT protocol would have been a time consuming and arduous task. Thus, the development of a new simple-to-use software package, designed specifically for the peculiar infrastructure based ITS to simulate the traffic patterns in the scope of V-FTT protocol, was opted. The main goal of the simulator is to integrate simple traffic scenarios with V-FTT based scheduling without explicitly simulating the radio channels as these software packages would require. It should be noted that with V-FTT based redundant scheduling, in which each OBU is scheduled by 2 or 3 RSUs, the probability of an OBU not receiving a trigger message is very low. An integration with SUMO is planned at a later stage so the scheduling simulator could be fed with more realistic traffic patterns. The simulator integrates wireless communications and road traffic platforms in an environment that could be easily tailored to scenarios that allow a performance analysis of V-FTT protocol. This simulator is discussed in detail in the following section.

4.5.1 *The User Interface and Inputs of the V-FTT Matlab-based Simulator*

The user interface of the V-FTT protocol based Matlab simulator is shown in Fig. 4.4 with the default inputs as required for simulation. The user is required to choose



Lane Width (m) 3
Vehicle Length 3.5
Truck Length 10
RSU Coverage Range 15
Vehicle Spacing 0.5
% Truck Normalized 0
SZ length (m) 150

V-FTT Inputs

Bandjacking Time 6.5e-3
Elementary Cycle 0.1
Modulation BPSK1/2
SW 3

Simulator Settings

Discrete Simulation Time (s) 0.1
Name of Log File log
● Graphical
○ Random
Refresh Rate (Hz) 10
RSU SimResults 1
Duration (s) 20
Number Lanes 4
Velocity Min (km/h) 50
Velocity Max (km/h) 120
Name of Event List File event_list.txt

Fig. 4.4 User interface of the vehicular flexible time-triggered simulator (with default inputs)

certain parameters from three input sections, i.e. inputs related to the Motorway (SZ), the V-FTT protocol and the Simulator settings. From the Motorway inputs section, a user can choose variables such as lane width, vehicle length, vehicles spacing and the RSU coverage range (all units are in meters), depending on the specific scenario that a user wishes to feed into the simulator. The V-FTT input section enables a user to select the V-FTT settings, e.g. the size of the elementary cycle (EC) and modulation type. The simulator settings section allows a user to choose whether the results produced be in graphical format or in the form of a textual log file. The user also has the option of running this simulator in a non-random fashion with a list of previously known inputs. In that case, the simulator picks up the data from an input file called event-list.txt.

All input parameters related to the V-FTT protocol, the safety zone and the simulation settings are used to initialize the auxiliary structures and statistical counters defined inside the simulator. Based on the values of these input parameters and the correlation existing among them, as defined by the mathematical expressions inside the simulator, the outputs are generated. These outputs can either be in form of a graph or a text file as opted by the user before the start of the simulation. The flowchart in Fig. 4.5 shows the flow of events taking place inside the simulator.

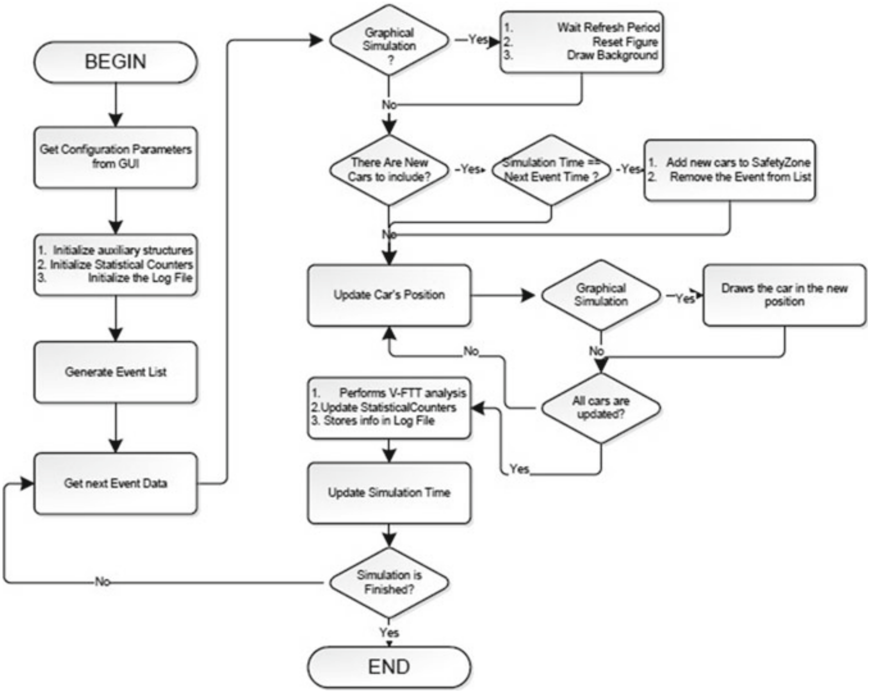

BEGIN
Get Configuration Parameters from GUI
1. Initialize auxiliary structures
2. Initialize Statistical Counters
3. Initialize the Log File
Generate Event List
Get next Event Data
Graphical Simulation ?
Yes
1. Wait Refresh Period
2. Reset Figure
3. Draw Background
No
There Are New Cars to include?
Yes
Simulation Time == Next Event Time ?
Yes
1. Add new cars to SafetyZone
2. Remove the Event from List
No
Update Car's Position
Graphical Simulation
Yes
Draws the car in the new position
No
No
All cars are updated?
Yes
1. Performs V-FTT analysis
2. Update StatisticalCounters
3. Stores info in Log File
Update Simulation Time
Simulation is Finished?
No
Yes
END

Fig. 4.5 Flowchart of events of Matlab-based V-FTT simulator

At the beginning of the simulation, all the auxiliary variables and statistical counters are initialized according to the input parameters. Based on the values of these input parameters an event list is generated which specifies the time of entry of each OBU into the Safety Zone. The simulator then checks whether the user has opted for an output in a graphical format or in the form of a file. In the former case, the simulator draws the background with all the RSUs installed on road sides, the number of lanes of the motorway and the number of OBUs entering at discrete times. If the graphical mode is not chosen, the simulator moves on without drawing the whole scenario in GUI format. The simulator then starts picking up OBUs from the event list and adding them to the safety zone according to their entry times. While adding an OBU into the SZ, it is removed from the event list at the same time as it has already been served. With the updating of the simulation clock, the positions of cars in the SZ are updated and the V-FTT analysis performed. The results generated are stored in temporary variables to produce outputs in the form of textual file logfile.txt and graphs. The simulation time is constantly checked for its finish time before which the simulation continues to generate results at discrete times.

4.5.2 Output Details of the V-FTT Matlab-based Simulator

After briefly discussing the user interface, architecture and input parameters of the V-FTT protocol simulator, this section discusses the outputs generated by the simulator. Figure 4.6 shows the graphical output of the simulator for the default input parameters as shown in Fig. 4.4. The SZ is taken to be 150 m with a 3 m width for each lane as default values. The EC duration is equal to 100 ms and the modulation type is BPSK1/2. When the simulation starts a screen as shown in Fig. 4.6 is presented. The

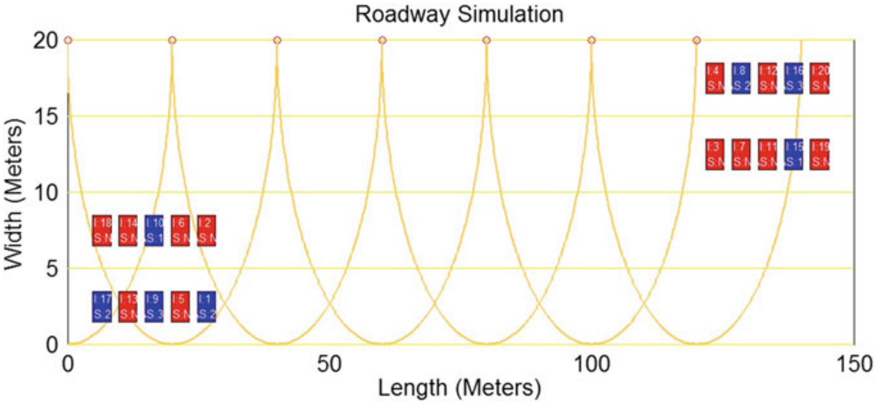


Fig. 4.6 Graphical output of vehicular flexible time-triggered protocol

screen shows the graphical view of the traffic situation as entered by the user through the input parameters.

The parallel (yellow) lines represent the lanes of the motorway whereas the parabolic (yellow) circles denote the geographic regions covered by RSUs (SZ). The motorway (by default) is a two-way four lanes road where nodes travel in two directions. The moving squares represent OBUs or nodes going from RSU to RSU across the SZ in both directions. While travelling across the safety zone, these mobile nodes can be seen changing their colours. The blue coloured nodes are the ones that have been allocated slots in the SOW of the current EC whereas the red coloured nodes are the OBUs that have not been allocated slots in the SOW of the current EC. As nodes move on, they change their colour from red to blue as they get slots in the next EC. The moving nodes also give information about the temporary IDs and the slot number allotted to it by each RSU. As the simulation continues all this information is updated whenever a significant event occurs.

Figures 4.7, 4.8 and 4.9 show the simulation results for different SIW values, where the rest of the inputs parameters are taken to be the same. SIW is the number of slots to be used in the RSU Infrastructure Window (one slot per RSU) corresponding to the maximum number of simultaneous RSU transmissions that an OBU can listen to.

The graphs in Figs. 4.7, 4.8 and 4.9 show that when congestion is low (which happens during the start of simulation) the efficiency of slot allocation is high as almost all the nodes (OBUs) are allotted slots by the V-FTT scheduler. However, as the number of nodes in the SZ increases, the efficiency of slot allocation falls accordingly. Similarly, the efficiency is higher for larger SIW values whereas the efficiency is lower for smaller SIW values.

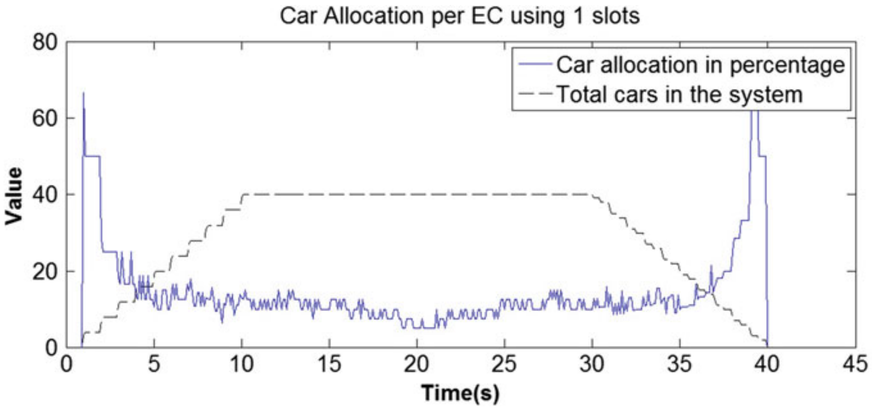


Fig. 4.7 Percentage of allocated slots with SIW = 1

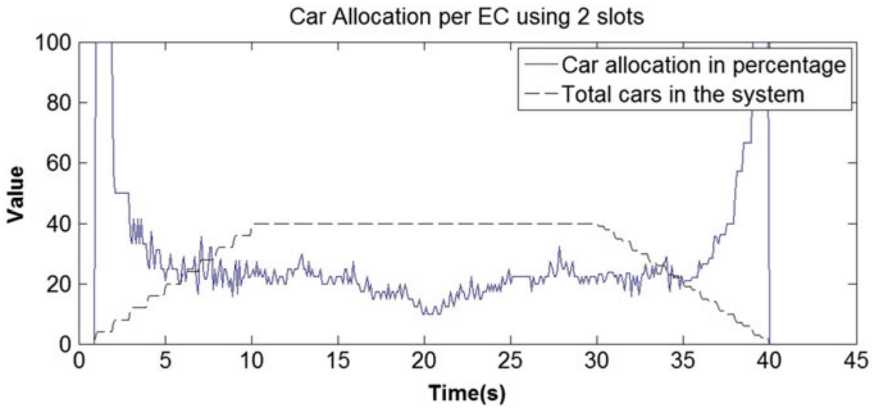


Fig. 4.8 Percentage of allocated slots with SIW = 2

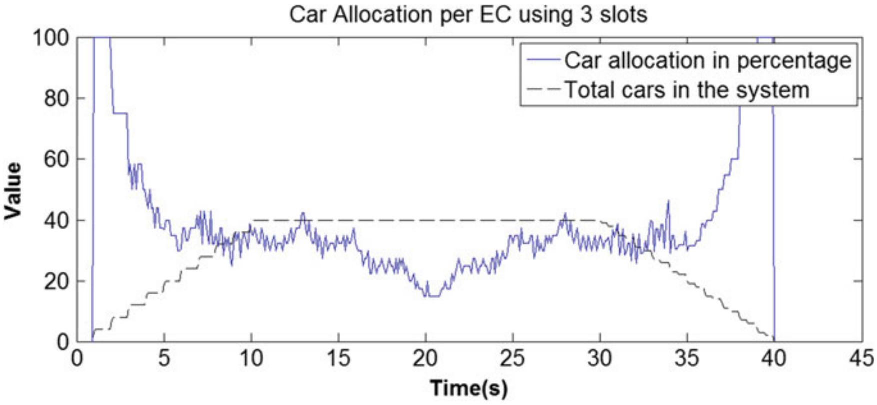


Fig. 4.9 Percentage of allocated slots with SIW = 3

4.6 Conclusion

This chapter presented a proposal for scheduling safety messages in the scope of wireless vehicular communications based on the Vehicular Time-Triggered Protocol. The proposed solution is an instance of the spatial time division multiple access MAC technique that relies on a deterministic network of road side units. Related work was analyzed and the Vehicular Flexible Time Triggered Protocol, which is an adaptation of the FTT protocol to wireless vehicular communications, was briefly discussed. V-FTT protocol guarantees road safety, data privacy and safety events timeliness in high vehicle density scenarios.

The proposed scheduling policy aims to increase the reliability of the safety messages transmission through the adoption of redundant scheduling while minimizing its impact on the bandwidth utilization by the implementation of slot reuse. For that purpose a slot assignment algorithm for multiple RSUs scenario was described.

Future work includes the definition of new heuristics to improve slot reuse, the design of a protocol to handle vehicles' registration when entering on a safety zone and a deterministic mobility mechanism to handover OBU's sessions between adjacent RSUs.

Acknowledgments This work is funded by the European Union's Seventh Framework Programme (FP7) under grant agreement n. 3176711 and by BRISA, under research contract with Instituto de Telecomunicações - Aveiro.

References

1. ICSI FP7 317671. Design and performance evaluation of a real-time MAC for IEEE 802.11p—D3.1.1: Fundamental Design Decisions of the Deterministic MAC Protocol. http://www.ict-icsi.eu/deliverables.html, online, as in June 20 2014 (2013)
2. L. Almeida, P. Pedreiras, J.A.G. Fonseca, The FTT-CAN protocol: why and how. IEEE Trans. Ind. Electron. **49**(6), 1189–1201 (2002)
3. K. Bilstrup, et al., Evaluation of the IEEE 802.11p MAC method for Vehicle-to-Vehicle Communication (2008)
4. A. Böhm, M. Jonsson, Handover in IEEE 802.11p-based delaysensitive vehicle-to-infrastructure communication. Technical Report IDE—0924. Halmstad University, Embedded Systems (CERES), (2009)
5. A. Böhm, M. Jonsson, Real-time communication support for cooperative, infrastructure-based traffic safety applications. Int. J. Veh. Technol. (2011)
6. R. Bossom et al., Deliverable D31 European ITS Communication Architecture—Overall Framework (2009)
7. G. Brar, D.M. Blough, P. Santi, Computationally efficient scheduling with the physical interference model for throughput improvement in wireless mesh networks, in *Proceedings of the 12th annual international conference on Mobile computing and networking*. September 2006
8. M.H. Chaudhary, B. Scheers, High spatial-reuse distributed slot assignment protocol for wireless ad hoc networks, in *Military Communications and Information Systems Conference (MCC'2012)*. (October 2012), pp. 1–8
9. W. Chen, C.-T. Lea, A node-based time slot assignment algorithm for STDMA wireless mesh networks. IEEE Trans. Veh. Technol. **62**, 272–283 (2012)
10. NS-3 Consortium. NS-3 Simulator. http://www.nsnam.org/
11. ETSI, ETSI TR 102 638 V1.1.1: Basic Set of Applications—Definitions (2009)
12. ETSI, 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 5GHz frequency band. November 2011
13. ETSI, Technical Specification 102 637-2: Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service, v.1.2.1. March 2011
14. S. Even et al., On the np-completeness of certain network testing problems. Networks **14**, 1–24 (1984)
15. N. Funabikiy, Y. Takefuji, A parallel algorithm for broadcast scheduling problems in packet radio networks. IEEE Trans. Commun. **41**(6), 828–831 (1993)
16. A.D. Gore, S. Jagabathula, A. Karandikar, On high spatial reuse link scheduling in STDMA wireless ad hoc networks, in *IEEE Global Telecommunications Conference (GLOBECOM'07)*. December 2007

17. W. Guo et al., An adaptive collision-free mac protocol based on TDMA for inter-vehicular communication, in International Conference on Wireless Communications and Signal Processing (WCSP) (June 2012), pp. 1–6
18. W. Guo et al., R-mac: Risk-aware dynamic mac protocol for vehicular cooperative collision avoidance system.Int. J. Distrib. Sens. Netw. (2013)
19. M. Hadded et al., TDMA-based MAC protocols for vehicular ad hoc networks: a survey, qualitative analysis and open research issues. IEEE Commun. Surv. Tutor. (2015)
20. IEEE, IEEE Standard for Information Technology: Telecommunications and information exchange between systems. Local and metropolitan area networks: Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (2012)
21. S. Khan, P. Pedreiras, J. Ferreira, Improved real-time communication infrastructure for ITS. Simpósio de Informática INFORUM (2014)
22. C.L. Liu, J.W. Layland, Scheduling algorithms for multiprogramming in a hard-real-time environment. J. Assoc. Comput. Mach, **20**(1) (1973)
23. T.K. Mak, K.P. Laberteaux, R. Sengupta, A Multi-channel VANET providing concurrent safety and commercial services, in *Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks*, pp. 1–9 (2005)
24. T. Meireles, J. Fonseca, J. Ferreira, The Case For Wireless Vehicular Communications Supported by Roadside Infrastructure, *Intelligent Transportation Systems Technologies and Applications* (Wiley, 2014)
25. V. Milanes et al., An intelligent V2I-based traffic management system. IEEE Trans. Intell. Trans. Syst. **13**(1), 49–58 (2012)
26. R. Nelson, L. Kleinrock, Spatial TDMA: a collision-free multihop channel access protocol. IEEE Trans. Commun. **33**(9), 934–944 (1985)
27. Open Simulation Platform For Intelligent Transport System (ITS) Services. http://www.ict-itetris.eu/index.html
28. M. Picone et al., *Advanced Technologies for Intelligent Transportation Systems* (Springer International Publishing, 2015)
29. L.C. Pond, V.O.K. Li, A distributed time-slot assignment protocol for mobile multi-hop broadcast packet radio networks. in *IEEE Military Communications Conference (MILCOM'89)*.vol. 1. (October 1989) pp. 70–74
30. S. Ramanathan, E.L. Lloyd, Scheduling algorithms for multihop radio networks. IEEE/ACM Trans. Netw. **1**(2), 166–177 (1993)
31. S. Ramanathan, Scheduling Algorithms for Multihop Radio Networks. Ph.D. thesis. Faculty of the University of Delaware, 1992
32. Y. Tang, M. Brandt-Pearce, Link allocation, routing, and scheduling for hybrid FSO/RF wireless mesh networks. IEEE/OSA J. Opt. Commun. Netw. **6**(1), 86–95 (2014)
33. Institute of Transportation Systems. SUMO—Simulation of Urban MObility. http://sumo-sim.org/
34. A.N. Vegni, T.D.C. Little, Hybrid vehicular communications based on V2V-V2I protocol switching. Int. J. Veh. Inf. Commun. Syst. **2**, 213–231 (2011)
35. P. Verissimo, *Uncertainty and Predictability: Can they be Reconciled?* (Springer, Berlin, 2003), p. 2584
36. D. Yang et al., A simple greedy algorithm for link scheduling with the physical interference model, in *Global Telecommunications Conference, 2009. GLOBECOM 2009*. IEEE. November 2009, pp. 1–6
37. H. Yu, Z. He, K. Niu, STDMA for Vehicle-to-Vehicle communication in a highway scenario, in *IEEE 5th International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications (MAPE 2013)*, pp. 133–138
38. R. Zhang et al., A novel centralized tdma-based scheduling protocol for vehicular networks, in *IEEE Transactions on Intelligent Transportation Systems*, (August 2014), pp. 1–6

Chapter 5
Medium Access Control (MAC) Techniques for Safety Improvement

Nuno Ferreira and José Fonseca

Abstract Vehicular networks are now an emergent field of research and applications. Using wireless communications in these networks offers a wide range of possibilities, but at the same time poses demands in terms of bounded delay, particularly in safe-ty-related applications. This chapter elaborates on the efficiency of MAC protocols based on IEEE 802.11p/WAVE standard to timely deliver safety messages. It covers several aspects of an infrastructure-based MAC protocol, and also details the characteristics needed for a safety-critical and bounded delay MAC protocol within a specific scenario. On the other side of the spectrum, an alternative solution is relying solely in V2V-based communications to disseminate safety messages. In this sense, it is also presented an approach for cases where the infrastructure may not be accessible (e.g., tunnels), or even not feasible to have total RSU coverage.

5.1 Introduction

Vehicular safety applications have stringent real-time requirements, namely they typically require low channel access delay with a well-defined upper bound. For example, a vehicle that breaks suddenly should emit a warning message which should be received by other vehicles within a specific period of time; otherwise, there is the risk that such information becomes useless. These requirements are mainly addressed through transmissions scheduling and medium reservation, functions performed in a sub-layer of the OSI model Data Link layer, the Medium Access Control (MAC) layer.

N. Ferreira (✉)
FCEE, Universidade da Madeira (UMa), Largo Do Município, 9000 Funchal, Portugal
e-mail: nunofabio@uma.pt

J. Fonseca
Instituto de Telecomunicações, Universidade de Aveiro, Aveiro, Portugal
e-mail: jaf@ua.pt

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_5

When dealing with road traffic, a dense automotive scenario is most common in urban areas. This relates to the absolute number of vehicles in the road. In such context, that scenario is less common in highways. However, we can get a so-called dense scenario in which the meaning of “dense” is not directly related with the absolute number of vehicles. So, taking as context the delivery of safety messages within an appropriate time bound, “dense” refers to a situation where the available bandwidth/medium for scheduling a new safety message transmission is almost or even fully filled. Therefore, the MAC layer protocol plays a major role in scheduling safety messages transmissions in order to timely deliver them. Typically, the MAC protocol is designed to suite a specific network topology and communication model.

The design of a MAC protocol for emergency message dissemination in a typical Vehicle Ad hoc Network (VANET), with ad hoc network topology as the name suggests, is challenging for several reasons. Until nowadays, most commercial wireless networks were designed to be used in a centralized topology/control and unicast based communication, with feedback allowable due to the point-to-point connection between nodes. In contrast, when dealing with VANETs, the nodes are always mobile and broadcast based communication is used in a decentralized network topology. The MAC protocol thus needs to be:

- Fully distributed and self-organizing, since there is no base station that coordinates scheduling in a centralized fashion, and because vehicles’ movement leads to constant changes of nodes;
- Scalable, since there is no centralized control, scalability is a very important issue to address, in the sense that the number of vehicles cannot be restricted. This means the MAC protocol should not block communication and should have the capabilities to cope with overloaded situations.

As stated in [6], the data traffic models found in VANETs are different from, for example, Wi-Fi or 3G. The predominant traffic type for newly born safety applications is periodic messages (short status message with the position and speed of a vehicle), with an update rate of 1–10 Hz, which will coexist with event-triggered hazard warnings when road traffic safety networks reach full penetration. Therefore, the communication model has some important features:

- It is mainly continuous time-triggered with broadcasts (contrarily to the predominant event-driven model of the centralized commercial networks in existence till today);
- Transient high network loads must be supported due to the repetition (rebroadcast) of safety messages to increase reliability. This is due to the fact that using broadcasts impairs the use of techniques such as Automatic Repeat reQuest (ARQ), in which an acknowledgement (ACK) of all packets is used;
- Unpredictable delays (on channel access or transmission collisions) should not exist since they could be intolerable because of the real-time deadlines emergency messages have;
- Packets leading to high overload can deteriorate the fast data exchange required, by limiting the available bandwidth.

Access to the channel in a timely and predictable manner is needed in order to meet a bounded real-time deadline. If a Carrier Sense with Multiple Access (CSMA) based method is used, since adaptive transfer rate can't be used due to the lack of ACK feedbacks, an increase in the number of nodes will result in more simultaneous transmissions, which will lead to decreased packet reception probability and excessive channel access delay, thus jeopardizing road traffic safety applications requiring upper bounded access delay and high reliability. The main argument for CSMA is that VANETs rarely experience high network loads, and traffic smoothing techniques can be used to keep data traffic acceptable. However, such techniques are commonly used in centralized networks (and only reduce the average delay) or geographically restricted networks, neither of which is applicable to VANETs due to their highly dynamic nature. As so, the problem with unbounded worst case delay still remains. Also, when using the original CSMA algorithm, hidden terminal situations may occur in centralized networks using an Access-Point (AP). This is due to collisions at the only receiver, which may be attenuated using RTS/CTS control packets, or in ad hoc networks independently of the MAC algorithm used. However, in the context of a VANET where a safety message is broadcasted, it may not be very harmful since there is more than one intended receiver and it is not likely that all nodes experience problems. Moreover, due to vehicles' high mobility, it is possible that broadcasts are received in perfect conditions, by the nodes experiencing problems in the prior transmission of the safety message. Also, due to 5.9 GHz band usage and multipath/diffraction characteristics, it is more likely that hidden node's problem degrades performance in urban scenarios than in highways.

The typical broadcast-based applications used in VANETs affects 802.11 ability to recover from collisions since there are no ACKs and the backoff procedure is invoked, at most, only once during the initial carrier sensing, therefore losing the advantage of increasing the CW to augment the number of backoff values.

Using 802.11p MAC, the most frequent case of simultaneous transmissions leading to collisions occurs when the nodes reach a backoff value of zero. Since the number of backoff values available to randomly select is smaller for higher priority classes, the probability of simultaneous transmissions in such classes is higher. The IEEE 802.11e EDCA scheme was also subject to performance analysis in several other works. Although there are also existing proposals on improving the performance of IEEE 802.11e, they cannot eliminate the intrinsic shortcoming of IEEE 802.11e, which is that it only supports "statistical" priority for specific flows but not "strict" priority for individual packets.

There are several works in which the IEEE 802.11p MAC method was studied in terms of real-time performance. In [1], simulations using a realistic highway scenario showed that vehicles using 802.11p MAC method (CSMA/CA) can experience unacceptable channel access delays, thus meaning this MAC method does not support real-time communications. Also, in [10] the DSRC/IEEE 802.11p MAC method was simulated on a highway road scenario with periodic broadcast of packets in V2V situation. The simulation results show that a specific vehicle is forced to drop over 80 % of its messages because it could not get access to the channel before the next message was generated.

5.2 Related Work

There are several literature proposals to deploy safety services in the vehicular environment. We present summarily some of the most relevant next. In [18], the authors show that a new feature from 3GPP Release 6, multimedia broadcast/multimedia services (MBMS) is able to provide I2V services efficiently on top of the UMTS network. In [15] the authors go even further and propose a unified V2I and V2V architecture using UMTS, claiming that when the High Speed Packet Access (HSPA) technology is fully functional, latency times will be small enough to allow V2V safety applications. They define a peer to peer (P2P) approach over cellular network, organizing vehicles in different traffic zones or clusters, where each vehicle communicates with a roadside entity responsible for that traffic zone. However, tests with current UMTS technology showed insufficient results for message propagation delay between vehicles.

The authors in [20] also propose a P2P overlay, but on top of an ad hoc network, using the concept of a supernode or super-vehicle per cluster. By adding this extra layer, unnecessary V2V communications are reduced. Although they had different intents, it is the same idea behind the cluster-based DSRC architecture proposed in [17], in which the super-vehicle is named cluster-head. Each DSRC channel is attributed a specific function allowing each vehicle to handle three tasks, cluster-membership management, real-time traffic delivery and non-real time data communications.

The author in [11] goes a little further and proposes a hybrid architecture, adding V2I communications to the P2P approach, but considering that only a super-vehicle can carry out communications between the infrastructure and other vehicles in its cluster.

In [2], the authors propose an extension of the local peer groups (LPG) concept for ad hoc P2P networking of neighboring vehicles described by the authors in [3]. A LPG is a kind of cluster organization with two degrees of coordination: Intra-LPG communication supports near-instantaneous safety applications (100 ms latency) and Inter-LPG communication for applications that somehow extend the driver's view. We find again the same concept of super-vehicle described earlier, this time named group header (GH). The GH periodically broadcasts a Heartbeat (HB) message to other vehicles (Group Nodes (GN)) within the LPG. Also in [2], it was added the presence of RSUs and V2I communications. They assume that V2V and V2I communications use different channels. Depending on the RSU network architecture, RSUs can be an extension of LPG, assuming the role of GH, behaving like regular GNs or even performing as an inter-LPG relay. RSUs can also assist V2V communications in order to help established LPGs and help create new ones.

Taking into account the parallelism that can be drawn between an RSU and an AP of IEEE 802.11, various authors have proposed coordination schemes between IEEE 802.11 APs. In [19], it was introduced an intra-access point synchronization scheme to allow cooperation between APs whilst providing guaranteed QoS using Point Coordination Function (PCF). PCF provides low delay and jitter, while allowing a fair bandwidth sharing. However, their scheme suffers from scalability issues.

Another important issue was taken in account in the work in [12]; they proposed a faster handoff scheme between 802.11 APs, reducing delays in the handoff process. The authors in [4] extended that scheme [12] in order to solve the problem of beacon collision between APs. APs have to be synchronized in order to transmit their beacons one after the others in the same channel, allowing mobile stations to get the beacons of available APs in the same channel. The authors in [21] proposed a coordination method between APs for IEEE 802.11 mesh net-works, to improve the throughput fairness for stations in different Basic Service Sets (BSS) of an infrastructure based WLAN network.

Despite having some related concepts (e.g., beacon collision between APs), none of these proposals are specific to WAVE. In this sense, it is here outlined a proposal of an RSU coordination scheme, somewhat similar to the method proposed in [4] about the APs synchronization, but taking into account the use of WAVE and a vehicular environment.

5.3 Improved MAC Techniques

With the focus on the prime goal of vehicular communications, which is safety-related applications, there is a need for meeting stringent real-time requirements. In this sense, the design of a MAC protocol is of utmost importance in order to access the channel timely. When using the IEEE 802.11p standard, which uses CSMA/CA as the MAC method, some enhancements are needed in order to meet real-time deadlines.

When dealing with vehicular applications, communications can rely solely on V2V or also I2V. The next two sub-chapters will devise some enhancements to the base MAC protocol, for each of the two cases. As already referred, due to the problem of meeting real-time deadlines, a TDMA based solution is pursued. It is assumed that the IEEE 1609.2 standard is also implemented, which means security services are used for all applications. Thus, data is not sensitive to service attacks trying to jam the communication medium, and anonymity, authenticity and confidentiality are assumed as granted in every message.

5.3.1 A Place for TDMA and Infrastructural Solutions

It is a fact that V2V communication is very promising and has numerous potential. However, taking into account the world economic crisis along with slow vehicle renewal rate, V2V solutions are facing slow implementations. As stated in 2013 by the technology market intelligence company, ABI Research, the V2V technology will gradually be introduced in new vehicles, resulting in a penetration rate of 61.8 % by 2027. Thus, it will take some time to be able to see the real safety benefits. Also, using Road Side Units (RSUs) can increase the range of communication by sending, receiving and forwarding data from one node to another, or benefit from their ability to process special applications forming V2I communication [13]. For instance, if traffic is congested in a specific highway zone, vehicles further behind without visual perception of the event may be informed by RSUs coordinating with each other and forwarding the information.

These are factors that favour I2V communications instead of purely V2V communications. When using this type of solutions for safety applications, it can be assumed that vehicles will be equipped with a communication device, as already used in electronic toll collection, which implements the specified MAC protocol. In addition, GPS devices are used in modern vehicles for positioning and other related purposes. Furthermore, this type of solution is somewhat resilient in the sense that safety event dissemination remains possible even in the case of a vehicle crashing and destroying its communication equipment, after the initial broadcast. Thus, the RSUs take part in the network as a special element in this kind of solution. In this type of solution advantage can be taken of the already installed infrastructure without being dependent on the large design cycle of vehicles. If needed in some zones, a relatively easy deployment of infrastructure as done in [14] is assumed feasible.

Considering IEEE 802.11p MAC standard as the base technology, a first protocol proposal is presented in [8]. The fundamental assumption is that non-enabled and enabled vehicles would coexist in the first stage of the technology growth. The enabled vehicles, equipped with OBUs, are able to communicate with other enabled vehicles and RSUs. Focusing on an already deployed infrastructure, the highway (or at least the accident-prone areas) is assumed to be fully covered by several RSUs, deployed by the respective operator.

As previously mentioned, the defined parameter set of EDCA is capable of prioritizing messages. However, with the increasing number of nodes sending messages of highest Access Category (AC), the collision probability increases significantly [5]. In densely populated scenarios or in case of filled MAC queues, native IEEE 802.11 MAC cannot ensure time-critical message dissemination. Proposals found in the literature are to integrate a re-evaluation mechanism for messages to continuously reduce the number of high priority messages and prevent long queues. In addition, the use of different EDCA parameters could mitigate the high collision probability. To reduce the number of high priority messages, it would seem appropriate the definition of a new AC (so-called “Safety AC”) within EDCA, reserved for collision and hard braking warning messages (it is not likely to exist several of these simultaneously), where the AIFS along with the CW value should be less than the AIFS of video AC. In this case it would be guaranteed that no contention between those messages and video AC messages would occur. However, this would not comply with the IEEE 802.11p standard.

So, the approach is to use a slotted based approach, with beacons transmitted by RSUs, to adequately reduce the collision probability in V2I (initial broadcast after a safety event) and I2V (rebroadcast in the target area by RSUs) communications. The idea, depicted in Fig. 5.1, is to have RSUs coordinating the rebroadcasting of safety messages with bounded delay and no contention in the target area.

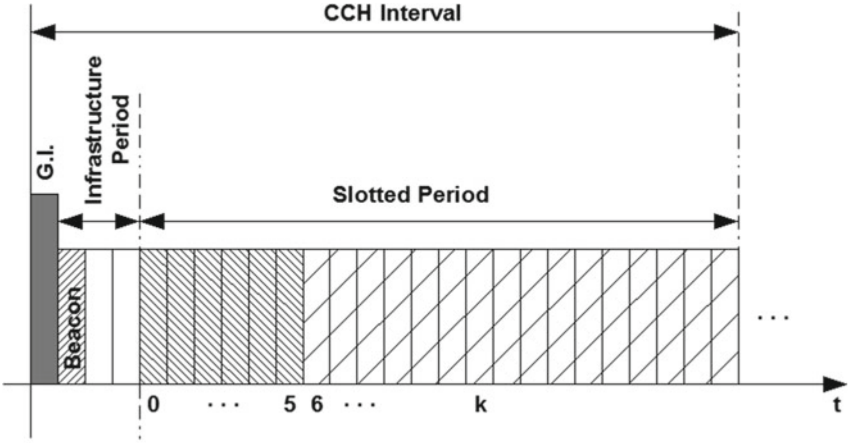


Fig. 5.1 Slotted based approach with beacons

The idea is to divide every Control Channel (CCH) interval into an Infrastructure Period (InfP) and a Slotted Period (SloP). The former is reserved for coordination between RSUs, and for beacon transmission. In this period all vehicles should listen to the channel. Regarding the SloP, the initial six slots are reserved for RSUs and are used by them if there are safety messages to rebroadcast. A safety message may need to be rebroadcasted by two adjacent RSUs depending on the target area of the message (distance intended to disseminate the warning from the safety event location). Each RSU uses one time slot for each event. The RSUs' beacon contains general information, such as the position of the RSU, and also information about the possible slots allocated by RSUs within SloP. The remainder of SloP is free and available to vehicles wishing to send messages (periodic or event-driven). Vehicles that generate an event broadcast the corresponding message on an empty slot (it should be noted that vehicles have knowledge of SloP occupation by listening the beacons in the beginning of the CCH interval). The RSUs will know the time the event was triggered and, by using beacons, will inform in the next CCH interval the specific slots being used to rebroadcast the message.

Despite it is not likely that two simultaneous events are generated, we can have three distinct situations: a clean transmission, a collision and an idle situation. In the first, reliable information is rebroadcasted in one or more slots, regarding the target transmission area. In case of transmission collision, a problem exists and in the corresponding slot it is rebroadcasted a warning.

By using this approach another advantage arises. Considering a vehicle brakes suddenly (generating an event), and after that collides destroying the communication equipment. In this case, the event will still be disseminated by RSUs despite the event originator cannot communicate further. The detailed definition of the coordination between adjacent RSUs, whose coverage areas are overlapped, is done on the following chapter. The Infrastructure Period duration is still dependent on the infrastructure deployment. Part of this work was presented in [7].

5.3.1.1 The I-TDMA (Infrastructure with TDMA Based Approach)

As referred previously, the RSUs play a major role in rebroadcasting warning messages adequately, i.e., avoiding contention in order to timely deliver the messages. Therefore, a critical issue is the coordination between RSUs, which is addressed here. Recall that the aforementioned approach was made in order to fully handle the problem of uploading (V2I) safety critical messages that could contend for the medium, and the problem of guaranteeing that the safety information arrives to the vehicle (I2V) within a specified time bound.

Taking into account the CCH interval organization defined previously, which can be seen in Fig. 5.1, every CCH interval is divided into an Infrastructure Period—reserved for RSUs coordination and for beacon transmission by RSUs—and a Slotted Period where the initial part is used for rebroadcasting safety messages, and the remainder is used as a contention period for short status messages, WSAs and safety event-driven messages. Using the InfP for RSUs' beacons may intuit us to use the SloP with a defined slot schedule done by RSUs, and informed within their beacons, for vehicle's utilization. However, this would require some kind of register/association, and the standard explicitly defines there is no association procedure in a WAVE context [9]. More importantly, it could jeopardize the timing requirements since the vehicle would first have to "register" itself, and only in the following CCH interval transmit a safety message within its reserved slot.

We are now concerned about the Infrastructure Period organization and how the RSUs will coordinate with each other in order to rebroadcast safety messages adequately. So, the focus here is only in the I2V message dissemination. The issue of slot selection for the initial broadcast (V2I) by the vehicle generating the event will be subject of analysis ahead.

The message target distance, d_{mt}, can be used to define the number of adjacent RSUs that will rebroadcast such message. Assuming the RSUs have a coverage range of dcr (radius), and each one is in the radio range from its adjacent, the total distance covered by n consecutive RSUs will be given by Eq. 5.1.

$$d_{mt} = (n + 1).\, d_{cr} \tag{5.1}$$

Being the interest in safety-related applications, the distance covered by three RSUs, each with a typical transmission range of $d_{cr} = 500\,\mathrm{m}$, is enough to disseminate the warning message and alert other drivers. This scenario is depicted in Fig. 5.2. A safety message should be rebroadcasted by several consecutive synchronization intervals. Thus, each RSU participating in the rebroadcast procedure maintains a counter, n_{ret}, which is decremented by one in every retransmission (i.e., in every synchronization interval) until it reaches zero, meaning the end of re-transmissions. The counter value is related with the message's lifetime, t_{lf}, which is the time a safety event must be rebroadcasted.

We are considering the road in a similar way as used by road authorities and car rally races, i.e., the road position is a linear function starting in 0 and ending in the road length, D_{rl}, as seen in Fig. 5.3.

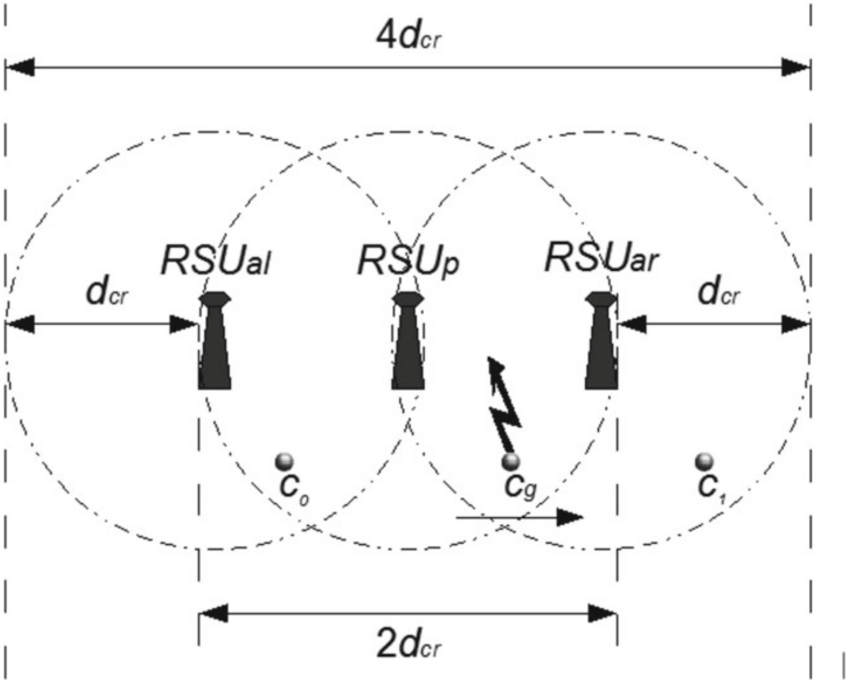


Fig. 5.2 Three RSU coverage range

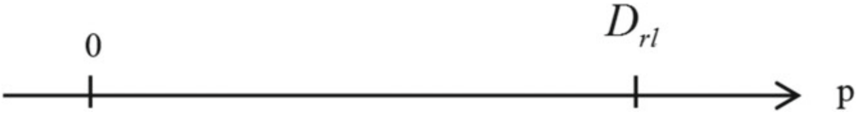


Fig. 5.3 Road position (p) as a linear function

Therefore, it is possible to know the driving direction information of a vehicle through two consecutive position measures, thus indicating if it is driving back or forward along the road (using Eq. (X.10), shown ahead).

When the vehicle, denoted as C_g in Fig. 5.2, is the only one to generate and broadcast a safety message in a synchronization interval, the message will be received by the so called primary RSU (RSU_p), and by an adjacent RSU (RSU_{ar}). From both, the RSU_p will be the one to rebroadcast the message since RSUar detects that the vehicle is moving towards it (through the driving direction information and vehicle position fields in vehicle's message). If, in the same synchronization interval, two other vehicles, one between RSU_{al} and RSU_p (C0 in Fig. 5.2), and the other ahead of RSU_{ar} (C1 in Fig. 5.2), also generate a safety event, this will lead to all those three RSUs having to rebroadcast the message. The beacon transmitted by each in the following Infrastructure Period contains information relative to the event ahead of

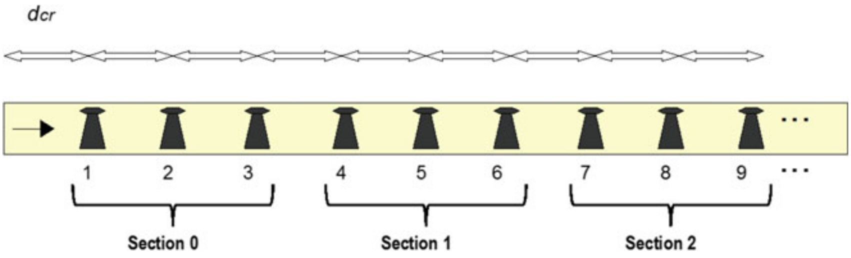


Fig. 5.4 RSUs numbering and sections

each. In order to allow proper announcement of the safety events through the beacons, from all RSUs, contention must be avoided between them. If the slot choice by RSUs was random, collisions would happen. Collisions caused by the same slot chosen by adjacent RSUs, or collisions caused due to the hidden node problem despite those RSUs are not at the communication range of each other (an example is shown in the following chapter). The following proposal is devised to cope with this issue. In summary, the Infrastructure Period will have five slots. The first three slots are used for coordination between RSUs. The last two slots are used for message dissemination through several adjacent RSUs.

5.3.1.2 Coordination for Beacons Transmission

It should be noted that RSUs do not share a physical connection like a back-bone. Instead, they also use the WAVE technology to communicate with each other. Each RSU has a number corresponding to its sequence along the road. Also, there are sections identified by a number. Each section contains three RSUs and each RSU belongs only to one section (an example can be seen in Fig. 5.4, from the start of the road, indicated by the arrow, and road direction from left to right).

Each RSU will use its InfP slot to transmit its beacon, whether it has listened or not a safety event broadcasted by a generator (OBU) (this will allow minimizing the collision probability between vehicles broadcasting a message within SloP, as explained ahead). In order to avoid contention between adjacent RSUs, and to avoid hidden terminal collisions (e.g., RSU_1 and RSU_3 transmitting a beacon in the same InfP slot, and causing RSU_2 to hear a collision), each RSU chooses its InfP slot using its own number (RSU_{nr}) and its section number ($Section_{nr}$), as devised in Eq. 5.2.

$$InfP_{slot} = RSU_{nr} - (3 \times Section_{nr}) \tag{5.2}$$

This guarantees that all RSUs along the road will transmit their beacons without collisions. To verify the correct functioning when RSUs allocate slots accordingly to the procedure explained above, Fig. 5.5 shows the transmission slots used for the nine consecutive RSUs. In this case, no collisions will occur.

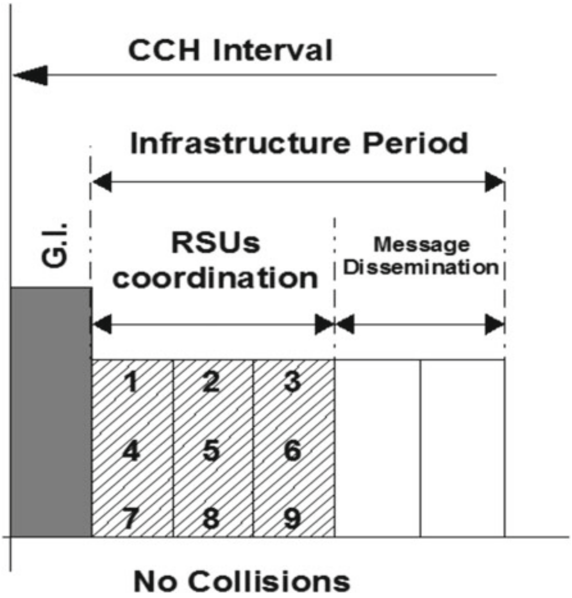


Fig. 5.5 Infrastructure period slot allocation by nine consecutive RSUs

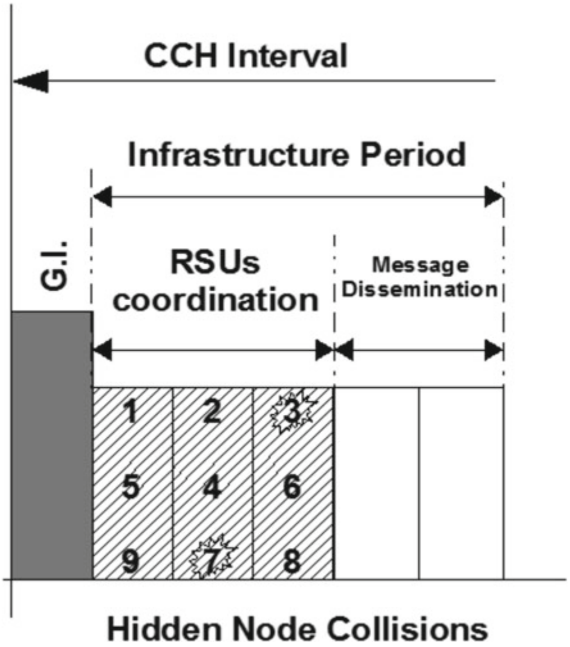
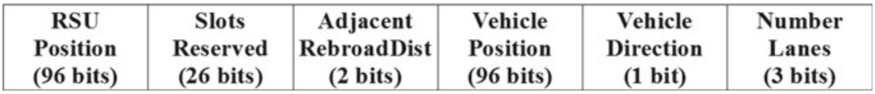


Fig. 5.6 Incorrect choice of InfP slots lead to hidden node collisions

An example of an incorrect choice of slot allocation is shown in Fig. 5.6. Although RSU_2 and RSU_4 are not in the communication range of each other, and thus do not listen each other's transmissions, if it happens that they choose randomly slot 2 to transmit a beacon, it will cause RSU_3 to listen a collision since it hears both beacons



RSU Position (96 bits)	Slots Reserved (26 bits)	Adjacent RebroadDist (2 bits)	Vehicle Position (96 bits)	Vehicle Direction (1 bit)	Number Lanes (3 bits)

Fig. 5.7 Beacon frame data fields (within WSM data field)

at the same time. This is represented in the Fig. 5.6 by surrounding RSU_3 with a ray type line. The same goes for RSU_6 and RSU_8 choosing slot 3 and causing RSU_7 to hear a collision.

Beacons are basically a WAVE Short Message from the WSMP within the WAVE protocol stack. As so, the information needed for protocol implementation will be contained in the WSM Data field of the WSM. The data fields of beacons sent (Fig. 5.7) are the following:

- "RSU Position" indicates the position of the RSU. It will be used by vehicles in the process of choosing a slot to transmit a message within SloP, when not using a random method. The number of bits needed for "RSU Position" is defined by GPS coordinates.
- "SlotsReserved" indicates how many and which slots are reserved in the Slotted Period. Since each vehicle listens at most two RSUs simultaneously, and assuming each can rebroadcast three events, this field uses two bits to define how many slots are reserved, and three fields of eight bits each, to define the slot number used for the event. Thus, if the first two bits are 0 it means this is a pure beacon and no safety event has occurred, and the following fields are ignored.
- "AdjacentRebroadDist" with a value of n_{td}, is used for message dissemination as explained in the next section.
- "VehiclePosition" contains the GPS coordinates in order to obtain vehicle's position in case of a safety message was received. A conversion from that to road position is done to get the vehicle position in road length (Fig. 5.3).
- "VehicleDirection" indicates the direction the vehicle is travelling (fi), in case a safety message was received.
- "NumberLanes" indicates the number of lanes in each direction of the highway. It will be used by vehicles in the process of choosing a slot to transmit a message within SloP, when not using a random method.

It is possible that two RSUs detect events that happened in an instant that leads to schedule the transmission in the same reserved SloP slot. In this situation, and since each RSU listens the beacons from adjacent units, the one with a higher value for "VehiclePosition" will maintain its slot allocation within the SloP (if the vehicle is moving forward, relative to road direction, otherwise the one with lower value). The other waits for the next InfP in order to allocate other slot(s). This gives higher priority for the further ahead event regarding the vehicles direction. Alternatively, a solution would be having each RSU allocating two slots for each event and using always the first of them, leaving the second for the RSU having the lower value of "VehiclePosition" in its beacon. However, this will lead to medium resources poor utilization (since two events so close in time may have low probability of occurrence), and the ability to deal with a lower number of events. Dependently on the timing requirements of the safety application, if waiting for the next InfP to announce the event is time jeopardizing, the alternative solution should be forced.

5.3.1.3 Message Dissemination

After the initial broadcast done by the event generator, the corresponding safety message should be appropriately spread throughout the road. This is done by RSUs. The dissemination of the safety event is done by analyzing the "AdjacentRebroadDist" field in the beacon. When an RSU listens a beacon with an n_{td} value higher than zero, and infers it is behind the vehicle (relatively to the driving direction) by examining the "VehiclePosition" and "VehicleDirection" fields, it will decrement n_{td} value by one and rebroadcast the message on an available slot. It will also send its beacon with the updated n_{td} value in the available of the two final slots of InfP (for each RSU rebroadcasting the message, one of these two slots will be used alternately for the respective beacon). This means that n_{td} is the number of RSUs, other than the originator RSU, retransmitting the message. It could be used to control the target distance of the message.

The global operation relative to RSUs' management (described in the two previous sections) may be seen in Fig. 5.8.

5.3.1.4 Choice of SloP Slot for Generator Initial Broadcast

Broadcasting status messages, service announcements (WSA), or safety events is done by vehicles within the SloP period. The approach may be using WAVE standard random access. As already stated previously, this will subject transmissions to possible collisions. Other possible approach, aiming to minimize transmission collisions, is performing a somewhat "deterministic" slot choice.

In the latter approach, assuming one lane, the slot chosen for a broadcast, slot1lane, is based on the vehicle' current position, xCi(t), and the RSu's position that is behind the vehicle, xRSUb, relative to the direction of travelling, obtained from the beacons heard in InfP. This is given by Eq. 5.3.

$$slot_{1lane} = \left\lfloor SloP(CP).\frac{\left|x_{C_i}(t) - x_{RSUb}\right|}{d_{cr}} \right\rfloor \tag{5.3}$$

SloP(CP) is the number of slots within the Slotted Period that are available for vehicles. The vehicle's current position is not the GPS coordinates, but its conversion to road position, as shown in Fig. 5.3. Similarly for the RSU's position.

This will work fine if it is considered only one lane. However, when considering multiple lanes, as common in highways, some problems may arise. If vehicles are

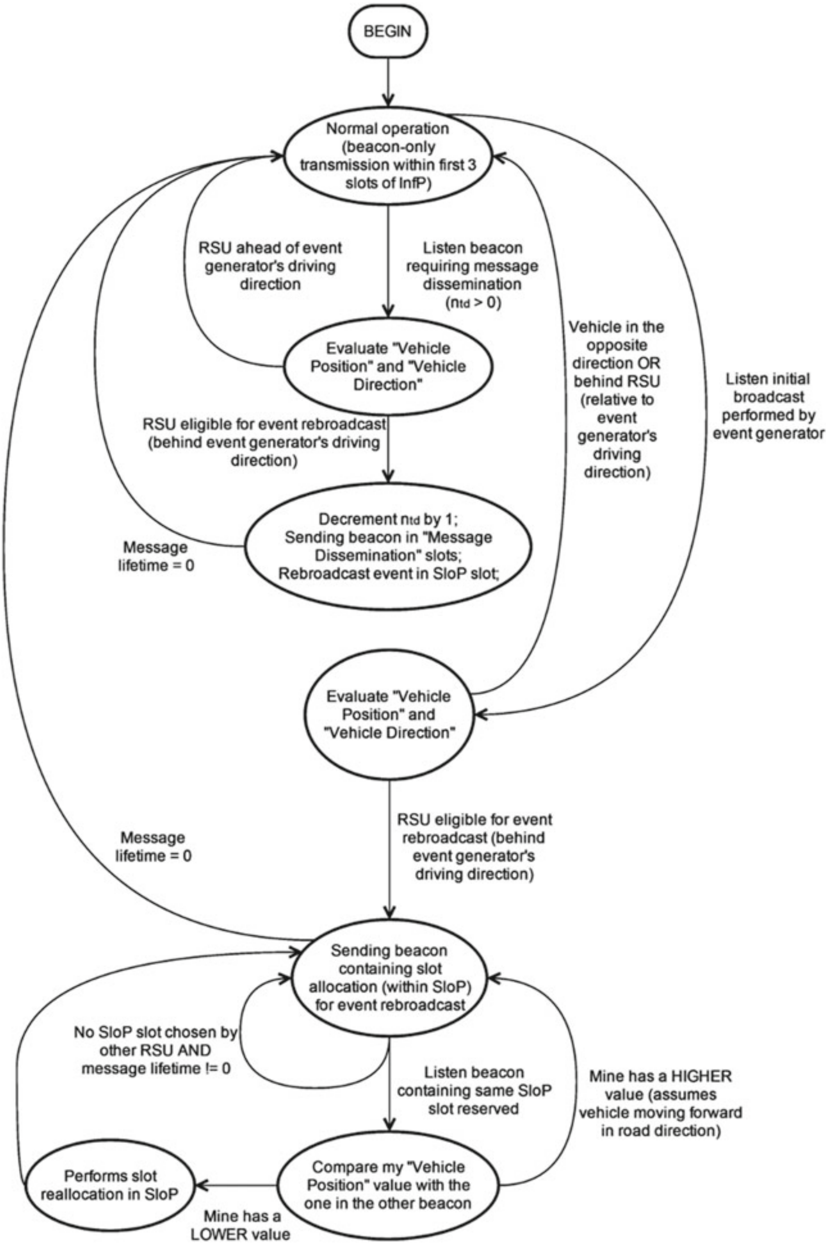

BEGIN
Normal operation (beacon-only transmission within first 3 slots of InfP)
RSU ahead of event generator's driving direction
Listen beacon requiring message dissemination (n_{td} > 0)
Evaluate "Vehicle Position" and "Vehicle Direction"
RSU eligible for event rebroadcast (behind event generator's driving direction)
Decrement n_{td} by 1; Sending beacon in "Message Dissemination" slots; Rebroadcast event in SloP slot;
Message lifetime = 0
Vehicle in the opposite direction OR behind RSU (relative to event generator's driving direction)
Listen initial broadcast performed by event generator
Evaluate "Vehicle Position" and "Vehicle Direction"
RSU eligible for event rebroadcast (behind event generator's driving direction)
Message lifetime = 0
Sending beacon containing slot allocation (within SloP) for event rebroadcast
No SloP slot chosen by other RSU AND message lifetime != 0
Listen beacon containing same SloP slot reserved
Mine has a HIGHER value (assumes vehicle moving forward in road direction)
Compare my "Vehicle Position" value with the one in the other beacon
Performs slot reallocation in SloP
Mine has a LOWER value

Fig. 5.8 RSU operation state machine

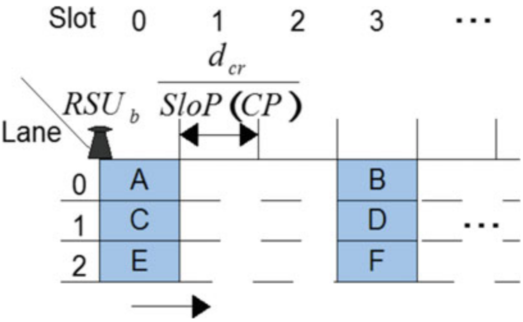


Fig. 5.9 Slot choice based on vehicle position and lane–Problem (a)

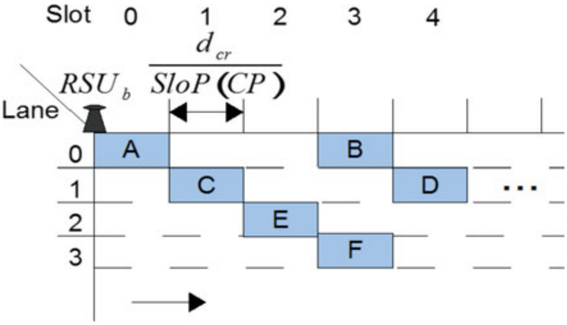


Fig. 5.10 Slot choice based on vehicle position and lane–Problem (b)

travelling in different lanes "side-by-side", their position will result in the same slot derived ($slot_{1lane}$). For example, in Fig. 5.9, vehicles A, C and E will choose slot 0 for message transmission, and vehicles B, D and F will choose slot 3, resulting in a collision if a pair of them (within each "group") have a message to transmit, which is possible. In Fig. 5.10, despite vehicles are not "inphase" in each lane, due to vehicle spacing, the same problem will occur for vehicles B and F. It is assumed that all vehicles travel at the same speed. This will give the worst-case results. If it was the case that vehicles travelling in different highway lanes have different speeds, a less number of vehicles would exist, since it is likely that vehicles travel faster when driving at the "outside" lanes. Thus, with this assumption, the inter-vehicle spacing is the same within all vehicles (for a given traveling velocity), when using for e.g., the Intelligent-Driver Model (IDM).

With the problem stated above, the slot derived by each vehicle should include the lane number the vehicle is travelling, $lane_{nr}$, as well as some method to derive if the vehicle is "out of phase". The lane number is the conversion of the GPS coordinates to an integer number, being 0 the most interior lane, and each consecutive following lane obtained by consecutive unity increments (as shown in Fig. 5.11). Considering the case where it is possible to perform slot allocation without collision (fewer vehicles than available SloP slots), the idea is to allocate the vehicles within the interior lane (lane 0) to the first SloP(CP)/nr_{lanes} slots, the vehicles in the following lane to the second SloP(CP)/nr_{lanes} slots, and so forth. nr_{lanes} is the total number of

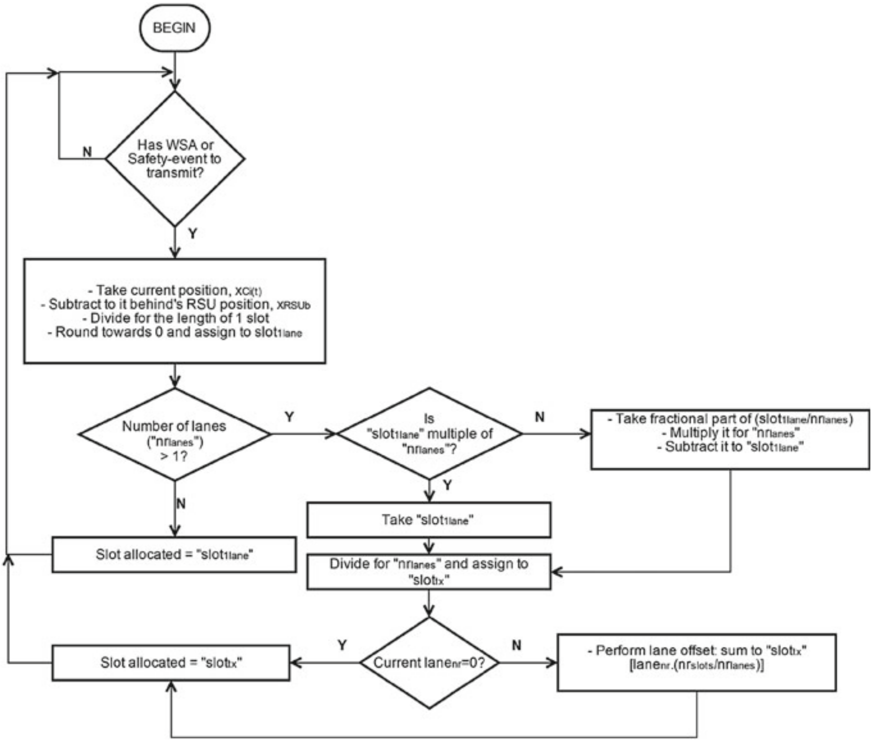

BEGIN
Has WSA or Safety-event to transmit?
N
Y
- Take current position, xCi(t)
- Subtract to it behind's RSU position, xRSUb
- Divide for the length of 1 slot
- Round towards 0 and assign to slot1lane
Number of lanes ("nrlanes") > 1?
Y
N
Is "slot1lane" multiple of "nrlanes"?
N
Y
- Take fractional part of (slot1lane/nrlanes)
- Multiply it for "nrlanes"
- Subtract it to "slot1lane"
Take "slot1lane"
Slot allocated = "slot1lane"
Divide for "nrlanes" and assign to "slottx"
Current lanenr=0?
Y
N
Slot allocated = "slottx"
- Perform lane offset: sum to "slottx" [lanenr.(nrslots/nrlanes)]

Fig. 5.11 Slot allocation procedure for WSMP messages' initial broadcast

lanes in each highway direction. $< x >$ represents the fractional part of x. The total expression used by each vehicle is shown in Eq. 5.4.

$$slot_{tx} = \frac{slot_{1lane} - \left[\left\langle \frac{slot_{1lane}}{nr_{lanes}} \right\rangle \times nr_{lanes}\right]}{nr_{lanes}} + lane_{nr}.\frac{SloP\,(CP)}{nr_{lanes}} \tag{5.4}$$

The flowchart describing the slot allocation procedure for the initial broadcast is depicted in Fig. 5.11.

It should be guaranteed that in a situation where all the slots are occupied and a new event generates a safety message, which transmission delay is critical, the node's transmission is not blocked (delayed) until a slot is available, and immediate access should be granted. In this sense, an improved SloP slot choice by OBUs to reduce collision probability of safety events should be taken. Since it is not likely that several simultaneous events occur within one CCH interval, a small number of slots may be reserved only for safety events broadcast.

Finally, in terms of synchronization, since the devised protocol is “centralized”, the RSU can provide the synchronization. However, it is assumed that all units have a GPS module due to the massive use in today’s vehicles.

5.3.2 *An Alternative V2V Based Solution*

Being V2V communication very promising and far investigated, and taking as base the work done with BRISA–Autoestradas de Portugal SA, a Portuguese highway operator, here it is outlined an alternative solution, where V2V communication plays the major role to accommodate time-critical messages within WAVE, for safety applications in highways. This model is proposed since the solution presented in the previous sections may not be feasible in some cases. First and foremost, full RSU coverage of the highway could not be possible. In addition, highway characteristics, such as tunnels, could limit the appropriate dissemination of safety messages if a warning generator vehicle could not communicate with an RSU. Therefore, the main goal of this model is to do the rebroadcast of safety messages only by vehicles.

Here, it is considered a highway where RSUs are only present in particular areas, namely all the entry and exit zones, near toll equipment and near possible hazardous areas (dangerous curves, bridges or tunnel entrances). In the highway areas that are not covered by RSUs, vehicles’ safety messages can solely rely on V2V communications for being rebroadcasted. The modeled state machine of MAC operation can be seen in Fig. 5.12. EP is the Event Period (time interval within CCH interval) and Lifetime relates to the rebroadcast time of an event, as explained later on this section.

It is considered that a safety event is associated with a vehicle and this vehicle will be the responsible for disseminating such event. The problem of several vehicles considering they are responsible for the same event is left out of the scope of this chapter. In case of an accident involving several cars, the first vehicle to disseminate the event will be considered the event generator, meaning that if other crashed vehicles listen to the generator transmission they will not start an event on their own.

5.3.2.1 Model definition

When the event is recognized, there could be a quantity of vehicles within the distance of interest of the event. This distance of interest depends on the type of event. The model formalization follows.

E(t1) is a safety relevant event that happened in instant *t1*. Equation 5.5 represents an important group of vehicles.

$$C_{dE}\left(t_1\right) = \left\{C_g; c_o, ..., c_n\right\} \tag{5.5}$$

$C_{dE}(t1)$ is the set of vehicles within the distance of interest of event E(t1) which includes the generating vehicle C_g and an n+1 (unknown) number of other vehicles,

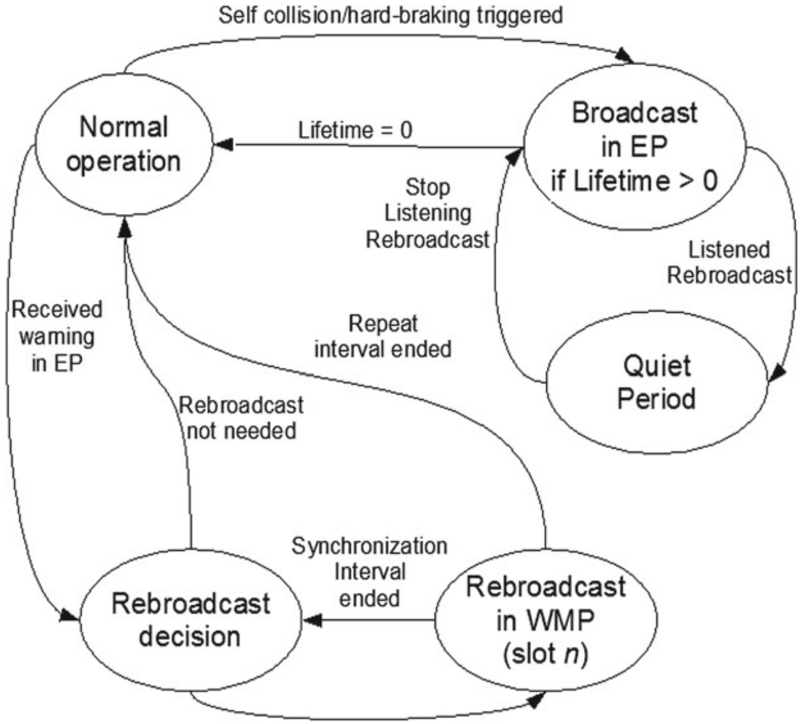

Self collision/hard-braking triggered
Normal operation
Lifetime = 0
Broadcast in EP if Lifetime > 0
Stop Listening Rebroadcast
Listened Rebroadcast
Received warning in EP
Repeat interval ended
Quiet Period
Rebroadcast not needed
Synchronization Interval ended
Rebroadcast decision
Rebroadcast in WMP (slot *n*)

Fig. 5.12 MAC state machine (rebroadcast only performed by vehicles)

all of which must receive the safety message. To avoid confusing vehicles with velocity we will use the letter c to represent vehicles in the equations, as in c for cars.

As already mentioned in Sect. 5.3.1.1, and illustrated in Fig. 5.3, we are considering the road in a similar way as used by road authorities.

When a generating vehicle wants to disseminate an event, it will transmit a frame in one of the safety slots reserved for that purpose. Two situations may arise from the transmission of that frame:

1. No vehicle listens to the frame, because there are not any vehicles within the transmission range.
2. Some vehicles listen to the frame. Defining an expected instantaneous range (in wireless communications this range fluctuates significantly, but here this is not problematic):

- $d_l(t)$–transmission range, in one direction, at instant t;
- $d_l(t2)$–transmission range of the message issued by C_g as a reaction to event E(t1), for t2 $>$ t1 for every t. This is considered constant in any direction, i.e., we are considering circular propagation.

Then, the aforementioned situation 1 means that

$$C_{dl}(t_2) = \left\{C_g\right\} \cup \{\} \tag{5.6}$$

where $C_{dl}(t2)$ is the set that includes the vehicles which are at a linear distance from C_g less than $d_l(t2)$.

Considering now situation 2 mentioned above, we have

$$C_{dl}(t_2) = \left\{C_g; c_o, \ldots, c_k\right\} \tag{5.7}$$

This set includes the vehicles within the transmission range of C_g, i.e.,

$$d\left(C_j\right) < d_l(t_2), \tag{5.8}$$

where

$$0 \le j \le k$$

and $d(C_j)$ is the distance in a straight line from vehicle j to the generator vehicle.

It should be noted that dependently on the distance of interest and the actual vehicles' placement on the road, the set $C_{dE}(t1)$ may have more, less, or the same number of vehicles than the set $C_{dl}(t2)$.

It is important to determine a vehicle's position in the road. It can be derived by the following equation.

$$x_{ci}(t) = d_{gps}(t_k) + (2f_i - 1)\int_{t_k}^{t} v_i \, dt, \quad t > t_k \tag{5.9}$$

where vi is vehicle i (or car i) instantaneous' speed and $d_{gps}(tk)$ is the position of the vehicle i in the road at the last instant where a GPS coordinate has been obtained (e.g., the entrance of a tunnel). The f_i function is used to account the direction vehicle i is travelling.

$$f_i = \begin{cases} 0 \; if \; \left(x_{C_i}(t_1) > x_{C_i}(t_2)\right) \;, \; (t_2 > t_1) \\ 1 \; if \; \left(x_{C_i}(t_1) < x_{C_i}(t_2)\right) \;, \; (t_2 > t_1) \end{cases} \tag{5.10}$$

i.e., if the vehicle is driving back or forward along the road its position goes from D_{rl} to 0 or vice-versa.

Equation 5.9 can be used to determine the vehicle road position in any instant or place, using available GPS information and data available from the vehicle itself, e.g., through the Vehicle On Board Diagnostics II (OBD2) interface.

We can consider the event relevant for vehicles travelling in both directions, or just consider the generating car driving direction. In a motorway this last scenario is often the relevant one. To find out if vehicle i is travelling or not in the same direction as the vehicle that generated the event (C_g), we need to compare f_i and f_g. If they are equal it means that the vehicles are indeed travelling in the same direction.

We now need to restrict this set of vehicles to a distance of interest of the event and driving behind (C_g). The vehicles within the distance of interest, (d_E), of the event are the ones that have

$$I\left(C_i\right) = 1\ if \left\{ \begin{array}{l} \left(x_{C_i}\left(t\right) > \left(x_{Cg}\left(t_2\right) - d_E\right)\right) \ \wedge \ \left(x_{C_i}\left(t\right) < x_{Cg}\left(t_2\right)\right),\ \mathrm{f_g} \ = \ 1 \\ \left(x_{C_i}\left(t\right) < \left(x_{Cg}\left(t_2\right) + d_E\right)\right) \ \wedge \ \left(x_{C_i}\left(t\right) > x_{Cg}\left(t_2\right)\right),\ \mathrm{f_g} \ = \ 0 \end{array} \right. \tag{5.11}$$

I$(C_i) = 0$ otherwise.

Getting back to situation 2, even if C_{dl} (t2) includes other vehicles than C_g, i.e., there are vehicles within the transmission range, it must be verified if each of those vehicles satisfy Eqs. 5.10 and 5.11 to be considered of interest (i.e., travelling in the same direction, behind the event generator, and within the distance of interest). One of the vehicles within this final subset will rebroadcast the event.

It must be noticed that we are ignoring the distance skewing due to vehicles' mobility. Our time scale will validate this assumption. We recall that event E happened at instant t1, the frame transmission at instant t2 and the interest range evaluation at t, where$t > t2 > t1$.

In Fig. 5.13 it is illustrated a hypothetic scenario reflecting situation 2 mentioned above. The event generator vehicle is the one marked with a "G" letter. Other vehicles are given a random number, from 0 to 7. The highway has two directions, which are marked with arrows on the leftmost side. The event generator is driving forward (meaning $f_g = 1$). In this particular case, deriving from Eq. 5.7, we would have a subset of vehicles within transmission range of C_g.

$$C_{dl}(t_2) = \left\{C_g;\ c_o;\ c_2;\ c_3;\ c_4;\ c_5;\ c_6\right\} \tag{5.12}$$

Also, considering only relevant the vehicles driving in the same direction as the generating car (using Eq. 5.10), it means that we are now restricted to vehicles C_0, C_1, C_2, and C_3. Taking also into account the distance of interest, and assuming a safety application with $d_E = 0.5$ km, and also considering only relevant vehicles following Cg (Eq. 5.11), we would finally get the vehicles C_0 and C_2 considered relevant for rebroadcasting the event.

5.3.2.2 Choosing the Event Rebroadcasting Vehicle

Using the aforementioned model, it is important to define some issues. The first should be to decide which vehicle will rebroadcast the safety event, from the set of vehicles chosen as candidates.

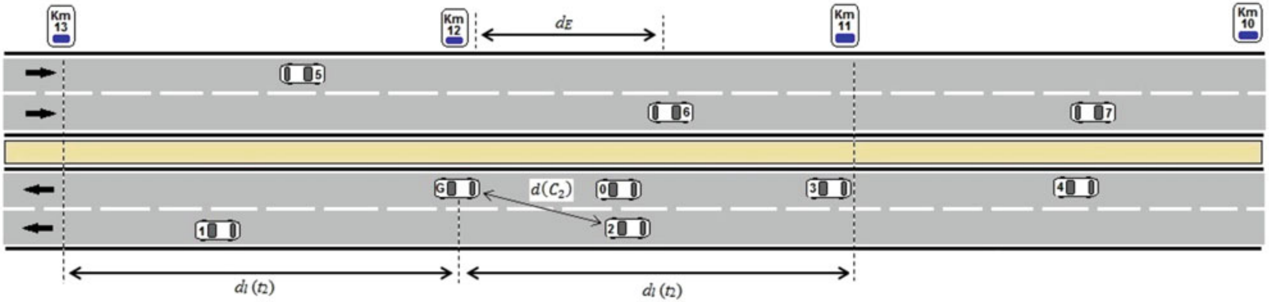


Fig. 5.13 Hypothetical scenario for situation 2

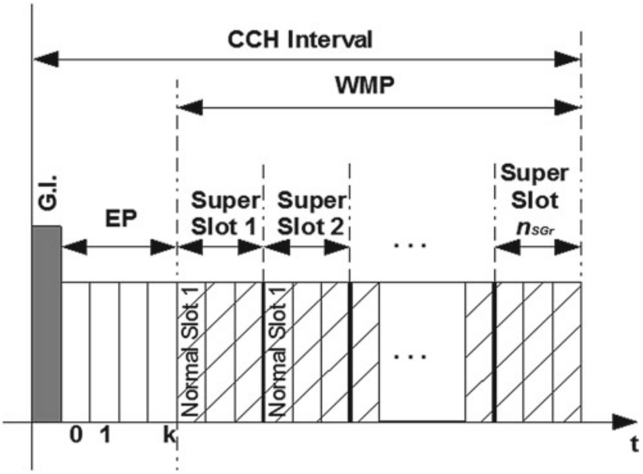


Fig. 5.14 TDMA based approach using WAVE's CCH interval

As it can be seen in Fig. 5.14, the CCH interval is divided into an Event Period (EP) and a Warning Message Period (WMP). The EP is used only by vehicles that generate an event, thus minimizing contention with rebroadcasting vehicles and giving the highest priority to the generator vehicle, C_g. Although it is not likely that simultaneous events occur, it is still possible. So, the EP is determined after fixating the WMP normal slots needed, and it is composed by a bit-rate dependent number of slots, where each event should be transmitted in one of them. To avoid contention, the possible simultaneous event generators perform a random choice of a slot within each EP before transmitting the event. The simultaneous generators that do not win medium contention will listen that an event is being broadcasted and will stop trying to broadcast their event. Another approach would be to perform a sort of "position-based" choice of the EP slot to minimize contention (although a reference point should be used).

The WMP works as a Contention Period (CP) and is used only by vehicles that receive a safety frame and need to rebroadcast such frame. It is intended to attribute different priorities in the slot allocation procedure according to the position of the vehicle, its velocity and also a random number. For this purpose, the WMP is divided as several groups of slots, called Super Slots (SupS). Each SupS has a certain number of Normal Slots (NS). The priority should be higher for larger distances from the generator vehicle (to reach the largest propagation distance with the minimum necessary broadcasts), which is achieved by the SupS. In a case where the distance results in the same SupS of another contending vehicle, a higher priority should be assigned to the vehicle with lower velocity (since it will stay at a higher distance from the generator vehicle). This is achieved using the NS within the SupS derived. In case the velocity is also similar leading to the same NS, a random number is used to avoid a transmission collision–through Sub Slots (SubS). These measures can minimize significantly the transmission collision probability.

One NS is sufficient to transmit a safety frame and to have some idle time. Within each NS, there are several SubS related to the time needed to transmit a bit. So, vehicles receiving a safety frame, that are in the distance of interest behind C_g, (see Eq. 5.11), and moving in the same direction ($f_i = f_g$), should compute the CP slots in which they will try the rebroadcast in the following CCH Interval. This is given by Eq. 5.12. n_{SupS} is the super slot number and is related with vehicle position, n_{NS} is the normal slot number within the chosen super slot and is related with vehicle velocity, and n_{SubS} is the sub slot number within the chosen normal slot and is used to avoid a transmission collision between vehicles having similar positions and velocities.

$$CP_{slots}\begin{pmatrix} n_{SupS}, \\ n_{NS}, \\ n_{SubS} \end{pmatrix} = \begin{pmatrix} \left(n_{SGr} - \left\lfloor \dfrac{\mathrm{n}_{\mathrm{SGr}} \cdot \left| x_{C_i}(t) - x_{C_g}(t_2) \right|}{d_l(t_2)} \right\rfloor \right), \\ \\ \left(\left\lceil \dfrac{vel_i - vel_{\min}}{gap_{vel}} \right\rceil \right), \\ \\ (random\,(0, 1) \cdot k_{SubS}) \end{pmatrix} \tag{5.13}$$

where n_{SGr} is the number of super slot groups, vel_{min} is vehicle's i velocity, vel_{min} is the minimum velocity defined for a vehicle, and

$$gap_{vel} = \frac{vel_{\max} - vel_{\min}}{k_{NS}} \tag{5.14}$$

where vel_{max} is the maximum velocity defined for a vehicle and k_{NS} is the number of normal slots within a super slot. k_{SubS} is the number of sub slots, which should be such that the remaining time in the normal slot is enough to transmit the safety frame and to have SIFS.

It should be noticed that after receiving a safety message, and earning the right to rebroadcast through slot allocation procedure, the rebroadcasting vehicle will act as a new generator vehicle for the vehicles behind it and the process repeats for such vehicles.

Another interesting issue is whether C_g should continue to broadcast the event. We consider appropriate, in sake of medium resources utilization, that when a generating vehicle listens to a rebroadcast, it should stop trying to broadcast itself the safety message. If it never detects a rebroadcast or, after some time, stops listening the rebroadcast, the generator vehicle starts repeating the broadcast if the message's lifetime (t_{lf}) is not zero.

Consequently, we can question what lifetime should the event have, i.e., how long must we continue to rebroadcast the event? Also, at what distance must the event be propagated?

Both of the questions cannot be answered in an absolute manner. This is application dependent. For example, an EEBL message will surely have a shorter lifetime than an accident warning. The same applies to the distance. For example, an accident can cause a traffic jam for various kilometers, while in the case of a sudden braking it is not needed to warn vehicles that are too far away. The message's lifetime should be enough to ensure that at least one vehicle will receive the message, i.e., it should account for an initial absence of vehicles within the transmission range, or connectivity loss due to sudden deceleration.

In order to perform an evaluation for different scenarios, it is useful to deter-mine how many vehicles are in the distance of interest (n_{dE}) of a possible event generator (C_g). This is shown in Eq. 5.15.

$$n_{dE} = \frac{d_E}{\left(C_{length} + C_{spacing}\right)} \times n_{lanes} \tag{5.15}$$

where n_{lanes} is the number of highway lanes in each direction, C_{length} is the vehicle average length, and $C_{spacing}$ is the vehicle separation value.

5.4 Conclusions

Safety-critical applications, e.g., sudden hard-brake or collision warning, require typically low channel access delay with a well-defined upper bound. These requirements pose the burden of message timeliness on the transmission scheduling and medium reservation functions performed by the MAC layer. It was noticed that such goals may not be fulfilled even when using implementations in conformance with the standards. For instance, the WAVE architecture accounts support for safety messages within vehicular networks. However, high collision probability is not negligible, particularly in dense scenarios, which may jeopardize the timing constraints of safety messages

The design of I-TDMA (Infrastructure with TDMA based solution) MAC protocol based solution considers the inclusion of a typical feature used in time-slotted self-organizing MAC protocols contained in several VANETs approaches. This is having the nodes transmitting information about which other nodes they receive information from, or their perception of the current slot allocations. This is done to prevent unintentional slot reuse by hidden terminals. However, it was shown in [16, 21] that in an highway scenario, with a communication channel modeled as a fading channel, hidden terminal situations do not contribute for a major performance deterioration in terms of packet reception probability.

Regarding IEEE 802.11p/P1609.4 MAC utilization and the specificity of CCH and SCH usage, the assumed requirement of using the CCH for safety information dissemination can strongly affect the end-to-end delay depending on the scenario considered (i.e., at what instant the safety event has occurred). If achieved delay is not admissible, the utilization of a mechanism forcing the use of CCH more often than SCH, i.e., stay tuned in CCH in some SCH intervals, could be a solution.

By using RSUs to rebroadcast the safety message, the only problem resides in the initial broadcast, since in the subsequent ones contention is avoided. Adding to this, if a careful slot choice is used by vehicles needing to transmit a message, collisions can be further reduced, and an improved upper bound to the end-to-end delay is achievable.

A preliminary study, which includes the intelligent driver model and also queuing delay, gave some promising results. The number of slots available for safety-related message transmission increases as the bit rate used increases. So, for higher bit rates the possible number of simultaneous transmissions is higher and the collision probability is reduced. Also, the number of slots is a function of the maximum message length (and its consequent duration for a given bit rate) it is intended to be used. The RSUs' beacons duration is approximately constant for beacons as long as 450 bits, losing only one slot in the three lower WAVE bit rates. So, if the latency achieved is not admissible, an eventual solution may be to work at a higher bit rate thus reducing the media access delay.

The collision probability (probability of at least two OBUs having messages to transmit and both choose the same slot), decreases with the increase in vehicle speed and also with the increase in the bit rate used. If the random method is used for slot choice the collision probability is higher than 0,9 for velocities lower than 80 km/h if lower bit rates are used. Contrarily, if the position-based method is used to choose the transmission slot, the collision probability is 0 for speeds higher than 20 km/h, even for the lower bit rates and for peak hour traffic situation.

Analyzing the higher velocities (more dangerous), from 80 to 120 km/h, the average media access delay, considering the specificity of using only the CCH to broadcast safety-event messages, varies from about 31 to 124,5 ms when using the random method for slot choice (the large variation interval is related with varying also the traffic situation–clear or peak hour as well as using the extreme WAVE standard bit rates 27 and 3 Mbps). In general, as the vehicle velocity increases, the average media access delay decreases. If the position-based method is used, the average media access delay is reduced and remains constant at 27,5 ms.

Finally, the total MAC delivery latency (end-to-end delay), when considering vehicle's velocity between 80 and 120 km/h, and ranging from a bit rate of 27 Mbps with clear way traffic to a bit rate of 3 Mbps with peak hour traffic, and a packet generation rate from 4 to 7 packets/s, varies from about 61–476 ms for random slot choice, and varies from about 29–40 ms for position-based slot choice. The traffic condition has a higher impact, in terms of relative increase, on the total end-to-end delay at higher bit rates, for the same speed. Also, the traffic condition has a higher impact, in terms of relative increase, on the total end-to-end delay at higher velocities, for the same bit rate.

References

1. K. Bilstrup et al., On the ability of the 802.11 p MAC method and STDMA to support real-time vehicle-to-vehicle communication. EURASIP J. Wirel. Commun. Netw. 5 (2009)
2. W. Chen et al., Local peer groups and vehicle-to-infrastructure communications. In: *2007 IEEE Globecom Workshops*, (2007), pp.1–6
3. W. Chen, J. Chennikara-Varghese, S. Cai, Local peer group architecture and organization for vehicle communications, in *Vehicleto- Vehicle Communications Workshop (V2VCOM 2005) co-located at ACM MobiQuitous* (2005)
4. S.K. Chui, O.-C. Yue, An access point coordination system for improved voip/wlan handover performance, in *Vehicular Technology Conference, 2006. VTC 2006-Spring. IEEE 63rd*, vol. 1 (IEEE, 2006), pp. 501–505
5. S.U. Eichler, Performance evaluation of the IEEE 802.11 p WAVE communication standard, in *Vehicular Technology Conference, 2007. VTC- 2007 Fall. 2007 IEEE 66th*, (IEEE, 2007), pp.2199–2203
6. ETSI Technical Report 102 862 v1.1.1, Intelligent Transport Systems (ITS); Performance Evaluation of Self-Organizing TDMA as Medium Access Control Method Applied to ITS; Access Layer Part (2011–12)
7. N.F.G.C. Ferreira, J.A.G. Fonseca, Improving safety message delivery through RSU's coordination in vehicular networks, in *2015 IEEE World Conference on Factory Communication Systems (WFCS)*, (IEEE, 2015), pp. 1–8
8. N. Ferreira, J. Fonseca, J. Gomes et al., On the adequacy of 802.11 p MAC protocols to support safety services in ITS, in *IEEE International Conference on Emerging Technologies and Factory Automation, 2008. ETFA 2008*, (IEEE, 2008), pp. 1189–1192
9. IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific re-quirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 6: Wireless Access in Vehicular Environ-ments (2010)
10. V.D. Khairnar, S.N. Pradhan, Simulation based evaluation of highway road scenario between DSRC/802.11 p MAC protocol and STDMA for vehicle-to-vehicle communication (2013)
11. J. Miller, Vehicle-to-vehicle-to-infrastructure (V2V2I) intelligent transportation system architecture, in *Intelligent Vehicles Symposium, 2008 IEEE*, (IEEE, 2008), pp. 715–720
12. I. Ramani, S. Savage, SyncScan: practical fast handoff for 802.11 infrastructure networks, in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 1 (IEEE, 2005), pp. 675–684
13. T.S. Rappaport, *Wireless Communications Principles and Practice*, (Prentice Hall, 2001)
14. L. Reggiani et al., Small LTE Base Stations Deployment in Vehicle-to- Road-Infrastructure Communications, (INTECH Open Access Publisher, 2013)
15. J. Santa, A.F. Gómez-Skarmeta, M. Sánchez-Artigas, Architecture and evaluation of a unified V2V and V2I communication system based on cellular networks. Comput. Commun. 2850–2861 (2008)
16. K. Sjoberg, E. Uhlemann, E.G. Strom, How severe is the hidden terminal problem in VANETs when using CSMA and STDMA?, in *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, Sept 2011, pp. 1–5. doi:10.1109/VETECF.2011.6093256
17. H. Su, X. Zhang, Clustering-based multichannel MAC protocols for QoS provisionings over vehicular ad hoc networks, in *Vehicular Technology, IEEE Transactions on 56.6*, (2007), pp. 3309–3323
18. D. Valerio et al., UMTS on the road: broadcasting intelligent road safety information via MBMS, in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, IEEE, 2008, pp. 3026–3030
19. D.D. Vergados, D.J. Vergados, Synchronization of multiple access points in the IEEE 802.11 point coordination function, in *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, vol. 2 (IEEE, 2004)

20. Y.-C. Yang et al., A real-time road traffic information system based on a peer-to-peer approach, in *IEEE Symposium on Computers and Communications, 2008. ISCC 2008*, (IEEE, 2008), pp. 513–518

21. D. Zhao, Inter-AP coordination for fair throughput in infrastructurebased IEEE 802.11 mesh networks, in *Proceedings of the 2006 International Conference on Wireless Communications and Mobile Computing*, (ACM, 2006), pp. 1363–1368

Chapter 6
Deterministic MAC Protocol Based on Clustering for VANETs

Unai Hernandez-Jayo, Aboobeker Sidhik Koyamparambil Mammu and Nekane Sainz

Abstract This chapter proposes a direction aware cluster-based multichannel MAC (DA-CMAC) protocol for Vehicular Ad-Hoc Networks (VANETs) in which vehicles travelling in the opposite direction may result in a short communication period. Clustering based on the direction of travel can reduce the cost of reconfiguration due to the short communication period. Each cluster consists of Cluster Head (CH), Cluster Members (CMs) and Gateway Vehicles (GV). CH calculates priorities of CMs based on the value of eligibility function that is received from the CM and assign each CM a unique priority based on the future position of CM, CM ID, and eligibility function. The eligibility function is calculated using the number of connected neighbors, average speed deviation, and the average distance between neighbors and itself. Clusters are independently managed and locally reconfigured as vehicles travel. Moreover, the proposed DA-CMAC protocol manages the channel access and allocates time slots to its CMs, reducing access and merging collisions in the channel, by grouping the time slots into two sets based on the direction of movement. Each CMs are allocated with one time slot in both control service channel to achieve fairness in channel access. Simulation results of DA-CMAC are compared with HCA protocol. Simulation results show that the DA-CMAC protocol have higher reliability of packets, fewer CH changes and fewer number of access collisions compared with HCA protocol.

U. Hernandez-Jayo (✉) · A.S.K. Mammu · N. Sainz
Deusto Institute of Technology, University of Deusto, 48007 Bilbao, Spain
e-mail: unai.hernandez@deusto.es

A.S.K. Mammu
e-mail: aboobeker.sidhik@deusto.es

N. Sainz
e-mail: nekane.sainz@deusto.es

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_6

6.1 Introduction

VANETs are an important part of the Intelligent Transportation System (ITS), which are experiencing rapid advancement in terms of technology. VANETs comprise both vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I/I2V) communication links based on wireless local area network technologies [7]. Over this architecture, cooperative vehicle safety systems can be deployed to delivery critical vehicle tracking information thanks to safety applications that obtain and combine information from vehicles and the infrastructure. In addition, non safety applications can be also offered to the drivers and passengers, those ones focused on infotainment or traffic management services. But these applications will improve traffic safety only when the predictability, scalability and reliability of the V2I and V2V communication increases, so that, traffic safety messages should be delivered with the highest priority in VANETs.

To accommodate these applications in Europe, ETSI has reserved a band of 30 MHz at 5.875–5.905 GHz band solely for traffic safety applications and other two bands of 20 MHz at 5.855–5.875 GHz and at 5.905–5.925 GHz for non-safety and future ITS applications. These dedicated bands have been divided into 10 MHz frequency channels. IEEE 802.11p is an approved amendment to the IEEE 802.11 standard to add WAVE [15], a vehicular communication system. In IEEE 802.11p defines a way to exchange data without establishing Basic Service Set (BSS), these kinds of functionality are provided in higher network layers. The protocol layers of WAVE are defined by a set of IEEE 1609 series of standards, but the one interested for the work presented in this chapter is the IEEE 1609.4, which deals the multi-channel operation at the IEEE 802.11p MAC layer. In WAVE, the Synchronization Interval (SI) is of 100 ms that is divided into an equal length CCH interval (CCHI) and a SCH interval (SCHI). The CCHI and SCHI are separated by a guard interval (4 ms), as shown in Fig. 6.1. The channel access in the SCHI can be continuous or alternative.

VANET has some peculiar characteristics, such as high vehicle mobility, dynamic topology and short link period. VANETs are confined to road structure and when vehicles travel in platoons share similar characteristics between neighbors in terms of speed and acceleration. At the same time, some of the vehicles travelling in opposite direction can be neighbors for a short period of time. Then, if vehicles are grouped in clusters according with common characteristics as the same direction, this can increase the stability of the cluster structure. Furthermore, clustering schemes combined with MAC protocols can reduce the channel access delay, data collisions and

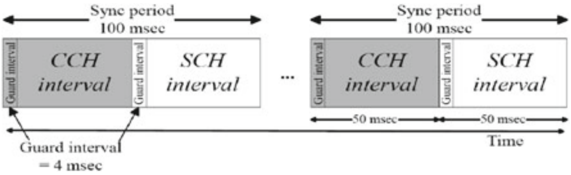


Fig. 6.1 Time division into CCH intervals and SCH intervals, IEEE 1609.4 standard [8]

increase the reliability of safety and non-safety applications. However, it is very challenging to design a reliable and efficient cluster based MAC protocol for VANETs in order to provide a dependable access of all the nodes in the network [9].

Then, this chapter introduces the Direction Aware Clustering Based on Multi-Channel Medium Access Control (DA-CMAC) protocol which aims to reduce the channel access delay and merging collisions in the channel, by grouping the time slots into two sets according to the direction of movement.

6.1.1 Related Work

Clustering is a method to group a small number of vehicles (defined as Cluster Members or CMs) with common characteristics into manageable entities called clusters [14], trying to solve important functions such as bandwidth allocation, channel access and routing. However, a master vehicle (known as Cluster Head or CH) is required to synchronize and schedule channel access to all vehicles in the cluster. Some of the well-known cluster based MAC protocols are discussed in this sub section.

In Hierarchical Clustering Algorithm (HCA) [4], authors proposed a new formation of clusters with a range of maximum four hops. HCA protocol schedules transmissions and channel access within the cluster to ensure reliable communication. However, it is not suitable for real time safety applications because overhead and packet loss is increased due to inter cluster interference. Moreover, HCA does not consider the direction of movement which decreases cluster stability and CH duration.

In [13], authors propose a hybrid MAC trying to limit the total number of clusters formed in the network. A distributed cluster based multi-channel communication protocol is proposed in [16], which collaborates both contention based and contention free MAC protocols with clustering. In papers [6], authors introduced a new type of cluster based MAC to minimize the hidden terminal problem. However, the scheme is not suitable for high density scenarios because the cluster stability decreases when the density of vehicles increases.

In [10], Region-based Clustering Mechanism (RCM) is introduced to improve the scalability of MAC protocol. In RCM, the network is partitioned into a number of space division units where each one is limited to a fixed number of vehicles for avoiding contentions of channels. Additionally, a non-interfering radio channel pool is allocated to a region. As a result, the number of vehicles contenting for the channel is reduced and thereby increase the throughput. However, this method provides low channel pool utilization in case of sparse traffic.

TC-MAC is proposed in [1] to reduce interference and provide fairness in channel access among vehicles in the cluster. TC-MAC is a combination of centralized cluster management technique and TDMA channel access. In TC-MAC, all the vehicles in the cluster are given different time slots for having collision free channel access. However, it is delay intolerant and therefore it is not suitable for safety applications.

ADHOC MAC is based on TDMA and is proposed only for V2V communications [3]. In this protocol time slots are allocated to different CMs and all the time slots are grouped together into virtual frames, then no frame synchronization is needed. However, ADHOC MAC suffers from throughput reduction due to vehicle mobility.

VeMAC [12] is a multi channel TDMA based MAC protocol, where single hop and multi hop broadcast communication is controlled by the control channel. Moreover, VeMAC eliminates the hidden terminal problem. Additionally, disjoint groups of time slots are allocated to vehicles to avoid the collisions arising in the control channel. However, a number of slots gets wasted in case of sparse and dynamic traffic.

In [11], authors propose the Dedicated Multi-Channel MAC (DMMAC) with a frame length of 100 ms, which uses adaptive broadcasting to eliminate transmission collisions and provide predictable delivery of packets. The frame length is divided into two equally-sized intervals CCH and SCH. However, this work does not discuss what happens if a vehicle disconnects from the network for a time period.

According with this analysis, we discuss the following issues that should be considered while designing MAC protocols for vehicular communication. The designed MAC protocol should be able to give fairness in channel access to all vehicles in the network. Moreover, not only it should provide fairness but also ensure predicable channel access in case of safety messages. Furthermore, MAC protocol should ensure reliable transmission during traffic congestions.

As VANETs are ad hoc networks in nature, then each vehicle should has enough knowledge of its neighbours. Employed MAC protocol should be fault tolerant and cope with highly dynamic topology of VANETs, and then, any topological change should deal without any delay in channel access. For example, if a CM linked to a cluster hears packets from another neighbouring cluster, this CM should get a common time slot in both clusters as soon as possible.

At the same time, as vehicles in a VANET struggle from congestion during rush hours of traffic or during road accidents, it would result in congestion on the channel. The MAC protocol designed should be able to assign slots to all vehicles in its one hop neighbours which are travelling in both directions. The non safety messages can be transmitted using service channels when some of vehicle might require more than one slots in SCHs. Moreover, provided protocol should be designed to supply more than one slots for a vehicle in SCHs. Those needs urged us to design a multichannel MAC for VANETs able to address those issues as well as to achieve better MAC performance with respect to existing VANET MAC protocols.

6.1.2 Requirements for MAC Layers in VANETs Networks

According to previous reflexions and in order to clarify them, when designing the MAC protocol for VANETs there are several important factors that have to be taken into account and that are related to the traffic safety and non-safety requirements, especially regarding reliability and delay.

- Vehicle density of VANETs depends upon scenarios; for example density is very high when a traffic accident occurs and a lot of vehicles pile up in the road. In that case the MAC protocol should scale enough with the number of the vehicles joining the network, in order to guarantee to all of them the access to the channel.
- Traffic safety applications are real time communication systems, implying demands on predictable delay for delivery of messages. Therefore, access delay must be bounded, so any messages, especially the safety ones, can access the channel within predictable delay. The worst case channel access delay is essential and should not exceed the message deadline. According to these requirements, the MAC protocol must be predictable when the density of VANETs increases.
- Reliability is coupled with the error probability of packets. Successful communication of the VANETs not only requires a predictable MAC protocol to access the channel, but also depends on packet delivery rate. Non safety messages are tolerant, but safety messages are not and, as high priority messages they need to have 100 % delivery rate. The MAC protocol has to achieve high delivery ratio for both safety and non-safety applications.

Since none of the existing MAC protocols discussed in Sect. 6.1.1 meet the MAC protocol requirements for safety applications. None of the existing protocols are scalable, reliable and predictable. In this chapter, we propose a DA-CMAC protocol that satisfies all the above mentioned requirements.

6.2 Direction Aware Cluster Based MAC Protocol Description

The main aim of Direction Aware Cluster Based MAC protocol (DA-CMAC) is to achieve relatively stable cluster topology, because by grouping vehicles travelling in the same direction increases the lifetime of members and reduces the overhead created due to frequent cluster reconfigurations. Moreover, each cluster needs to elect a CH for scheduling channel access. The problem of stable cluster formation and CH election can be solved by using graph theory. Moreover, the optimal number of CHs can be obtained using the Minimum Dominating Set (MDS) and the problem of stable cluster formation can be solved using a Minimum Connected Dominating set [2]. In DA-CMAC, an undirected graph is used to represent the VANETs using the MDS in G proposed below [5], and reduce the number of CH re-elections caused due to vehicular mobility. First, we introduce the ideas and definitions of each which will be used throughout the rest of the section:

- Undirected graph G = (V, E), where V is a set of vehicles travelling in the same direction and $E \subseteq V \times V$ is a set of links of vehicles which are in each others communication range or whose distance between each other are less than the cluster radius L.
- A MDS (S) of a graph $G = (V, E)$ is a subset of V ($S \subseteq V$) such that each vehicle in S is in the transmission range of at least one vehicle in V.

- A CH is a member of the MDS. CH organizes and schedules channel access for some members or at least one member in the set V.
- A Gateway Vehicle or GV is a vehicle that has a direct link between two vehicles in the S.
- A cluster, as it has been introduced before, is a group of vehicles travelling in the same direction and whose distance between each other is less than or equal to L_r. Each cluster has at least one CH and can have set of GVs. Therefore, a cluster is a subset of vehicles with the same CH.
- CH_j^l is the CH of the jth cluster travelling in L direction and $CM_{j,v}^l$ is the vth CM of jth cluster travelling in L direction having jth transmission time slot, for $v \in V$. Each vehicle in the cluster is allocated in a unique transmission slot. Only one slot is allocated per CCP and SCP.
- Each member in the cluster has a local ID and a global ID. CH allocates slots based on the local ID. Each vehicle in the network carries (gi, li, ch, bch, g, d, sl, s, n, p), where gi: global ID; li: local ID: ch: CH ID; bch: backup CH ID; g: geographic location; d; direction (0 towards head or 1 towards tail), sl: its own slot in the next frame, s: speed of the vehicle, n: number of connected neighbours, p: vehicle priority. In this work we assume that CH assigns each incoming vehicle to a local ID greater than 1. For the clustering purpose, each vehicle maintains a small amount of information of itself and its neighbouring vehicles. Periodically, a vehicle broadcasts the status message, the cluster information is embedded in the status message.

The initial creation of the MDS consists of the following 4 phases:

- Fast selection of a relatively stable CH.
- Fast selection of CMs belonging to the elected CH.
- Fast selection of tail and head list of GVs.
- Allocating slots to all CMs and GVs in both CCP and SCP.

The possible roles and transitions of the vehicle in the network is presented in Fig. 6.2. The states in this figure denote the relevant roles in the network and lines define the transition from one state to another state. In this protocol, five different roles for vehicles are considered: undecided state, CH, CM, BCH and GVs. Undecided state is the initial role of all vehicles executing this algorithm. All CMs periodically sent status messages to its CH. GVs are those vehicles that are linked to more than one CHs, and they are used to forward the control information or slot information of one cluster to another cluster. CH manages and schedules the channel access for members of the cluster (CH is a member of the minimum dominating set of S). The CH maintains two lists of GVs: the Gateway Head List (GHL) and the Gateway Tail List (GTL).

The GTL set is a group of all GVs whose position is behind the CHs position sorted in descending order. The GHL list contains all gateway vehicles whose position is greater than the CHs position sorted in ascending order. The CH keeps updating these lists according to changes on the network's topology in order to ease the formation of clusters, in which the maximal distance from a CH to any other vehicle in its cluster

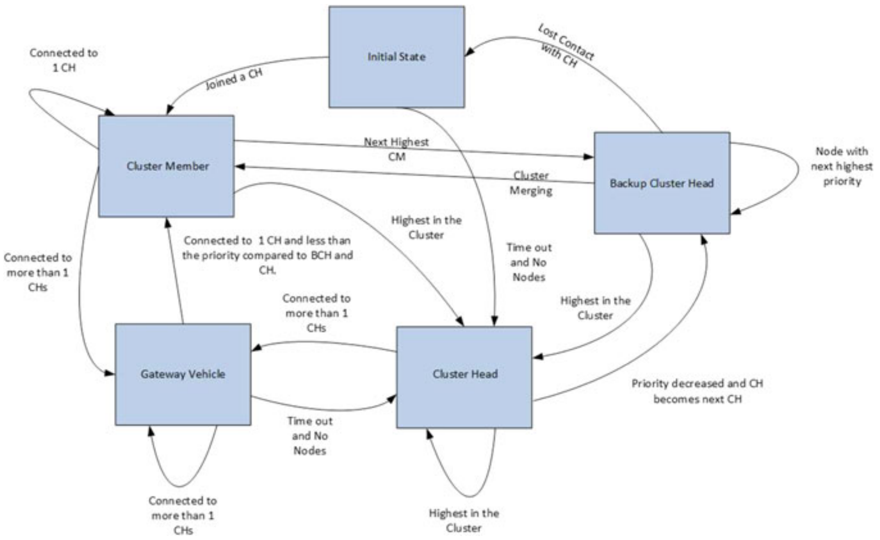


Fig. 6.2 State transition of DA-CMAC

should be less than or equal to L_r. When the distance between two CHs is detected to be less than or equal to a predetermined threshold, $D(D \leq L_r)$, the backup CH will take the position of CH. The CH will select a backup CH that has the next highest priority factor among all vehicles in the cluster other than itself. To increase the stability of the cluster's topology, the elected CH should be as stable as possible.

Each CH will calculate the future positions of all of its CMs after time t_n, based on received speeds as $P(t_n) = P(t_1) + s_1(t_n - t_1)$, where $P(t_1)$ is the position of a vehicle at t_1, t_n is the end of the next frame, and s_1 is the speed of the vehicle advertised at time t_1. If more than 75 % of the current CMs of the CH become out of the transmission range at t_n but are still within the backup CHs range, the current cluster will hand the responsibility to the BCH.

6.2.1 Cluster Head Election

Different factors are considered to elect the CH. Any vehicle should only consider the parameters of those vehicles that are one hop away from itself. Too many CHs around the same set of vehicles led to no MDS [2]. The information about neighbours helps in achieving MDS. CH calculates priorities for all registered vehicles based on the status message received earlier. We make various assumptions while designing the equation for calculating the priorities. We make two assumptions, no pair of vehicle can receive the same priority in its one hop neighbour or cluster. Moreover, no pair of neighbouring vehicles can have the same instant speed during their travel in the network. This means that the relative speed between two vehicles can never be zero.

After random election of CH and formation of CMs, each CH calculates the vehicle that has highest priority in the cluster based on the following conditions:

- The vehicle has the highest priority among those vehicles that are travelling in the same direction and in its one-hop neighbourhood.
- Each vehicle calculates the Connected Set (CS). The overall CS β, is the utmost number of vehicles that are one hop neighbour to vehicle i. This is represented as

$$\beta_m(t) = \sum_n C(i, j, t) \tag{6.1}$$

  where j is a potential one hop neighbouring vehicle. $C(i, j, t)$ is equal to 1 if both i and j are with in the transmission range of each other at time t. Additionally, $C(i, j, t)$ is equal to 0 if both i and j are not in the range of each other at time t. Moreover, we have identified the number of CS between a vehicle and all other neighbours that are travelling in the same direction.
- Each vehicle calculates the average distance between all neighbours and itself using the Eq. 6.4. This represents, how close are neighbours to one vehicle. Taking account of this parameter will decrease the packet delay and increase the lifetime of CH.

$$D_{xavg} = \left(\frac{D_{x1} + D_{x2} + D_{x3} + \cdots D_{xn}}{n} \right) \tag{6.2}$$

$$D_{yavg} = \left(\frac{D_{y1} + D_{y2} + D_{y3} + \cdots D_{yn}}{n} \right) \tag{6.3}$$

$$\triangle\, D_{i,ne} = \sqrt{|D_{xi} - D_{xavg}|^2 + |D_{yi} - D_{yavg}|^2} \tag{6.4}$$

- To take into account the mobility of VANETs. Each vehicle calculates the average difference of speed between one vehicle and all its CMs using the Eq. 6.6. This parameter is used to avoid elected CHs losing connectivity with their neighbours very soon, the eligibility of a vehicle should decrease quickly when its speed has a big difference from the average speed. Thus, a vehicle with large speed deviation is assigned lower priority.

$$S_{av} = \left(\frac{S_1 + S_2 + S_3 + \cdots S_n}{n} \right) \tag{6.5}$$

$$\triangle\, S_{i,ne} = |S_i - S_{av}| \tag{6.6}$$

- To increase the stability of the cluster topology, the elected CH should be as stable as possible. Each vehicle will calculate the expected positions of all of its one hop neighbours after time T_f, based on their advertised speeds as $x(T_f) = x(0) + vT_f$, where $x(0)$ is the current position of a vehicle. More priority is allocated to the vehicle if all of its one hop neighbours are still within its communication range after time T_f.

Overall, to avoid elected CHs losing connectivity with their neighbours very soon, the eligibility of a vehicle should decrease quickly when its velocity has a big difference from the average speed, the distance between each other is large and the number of connected neighbours is less. Thus, a vehicle with large speed deviation, less number of neighbours and it has the largest distance between all its neighbours, then it is assigned lower priority. Many possible solutions can be used to compute the priority of a vehicle while considering the aforementioned criteria. Therefore we define that the priority of vehicle p_i is given by Eq. 6.7:

$$p_i = Hash(P(t_n) \oplus i) \oplus E_i \tag{6.7}$$

A hash function is used to generate a unique priority for vehicle i according to the input of local ID, the future position of the vehicle and the eligibility function E_i, which is defined by Eq. 6.8:

$$E_i = \beta_i e^{-0.2 S_{i,ne} D_{i,ne}} \tag{6.8}$$

where $\beta_i \in (0, \beta_i^{max})$ is the number of possible connected neighbours, $S_{i,ne} \in [0, 120]$ is the speed deviation and $D_{i,ne}$ is speed deviation with its one hop neighbours. The units of $D_{i,ne}$ and $S_{i,ne}$ are meters and miles/hour, respectively.

6.2.2 Scheduling at DA-CMAC Protocol

After the creation of the cluster, election of CHs, organization of CMs and gateway vehicles to different clusters, the CH has to assign slots to its CMs for channel access. In DA-CMAC, it is assumed that every vehicle in the cluster is equipped with a single transceiver. However, the IEEE 1609 has allocated 7 channels for both safety and non-safety applications. DA-CMAC protocol is designed in such a way that IEEE 1609 interfaces demodulate one channel at a time.

In this protocol, it is considered that there are c service channels (SCHs) numbered from 0 to $c-1$ and one Control Channel (CCH). Similar to IEEE 1609.4 [8] protocol, we introduce the so called system cycle, which is divided into Control Channel Period (CCP) and Service Channel Period (SCP) sub periods and repeat every 100 ms. Each CCP period is a frame of CCH channel. CH utilizes SCP period and takes over the responsibility of intra-cluster management. This task includes: assigning time slots to all CMs, receiving newly arriving vehicles (undecided), processing and disseminating all received messages. The proposed protocol assumes a system cycle is shared between the SCP and CCP.

As shown in Fig. 6.3, the CCP consists of slots for CMs, GVs and CH itself. The time slots in each frame are divided into three disjoint sets L, R, and U. The set L

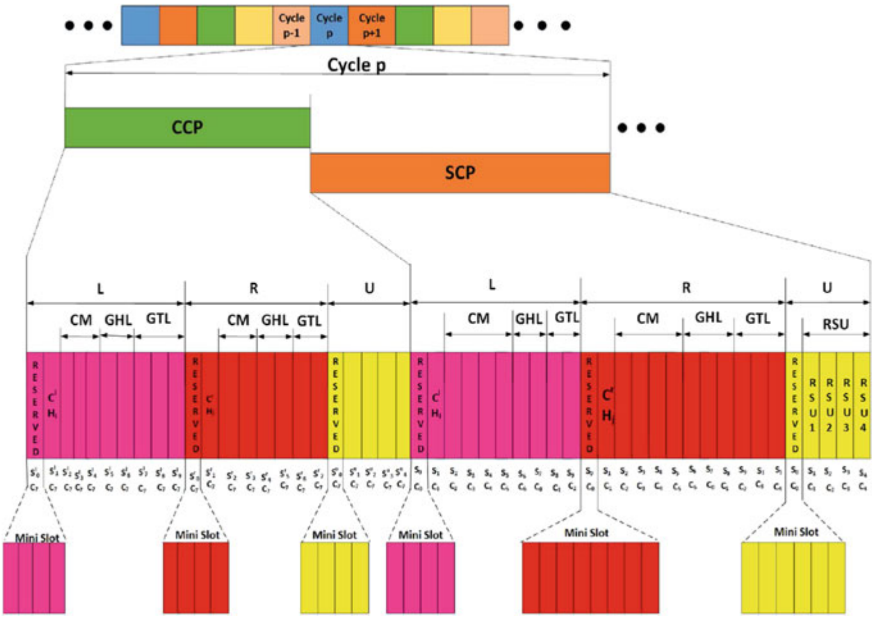


Fig. 6.3 DA-CMAC frames

is associated with the vehicles going on the left direction, R for vehicles travelling in opposite direction of vehicles in L, and set U for RSUs placed on the side of the road. The L disjoint set is further divided into three disjoint sub sets CMs, GFL, GTL. Vehicles travelling in left direction own only one slot in set L in CCP. At the beginning of each cycle, all vehicles switch to CCH channel. Each system cycle starts with a frame sent by the CH. Each vehicle is required to include how the channel is perceived using a field called Slot Information (SI), in order to realize the slot reservations. DA-CMAC extends the SI to contain the speed, direction and mode of vehicles rather than just the status of each slot. An undecided state vehicle discovers the unoccupied slots by sensing the channel continuously for a synchronization interval.

Each CCH interval contains $(N + 3)$ slots numbered from 0 to $N + 2$. $N = N^l + N^r + N^{rsu}$ is the sum of number of slots for vehicles travelling in the left direction $(N^l = N^l_{CM} + N^l_{CH} + N^l_{GHL} + N^l_{GTL})$, number of slots for vehicles travelling in the right direction $(N^r = N^r_{CM} + N^r_{CH} + N^r_{GHL} + N^r_{GTL})$ and road side units in the opposite directions $(N^{rsu} = N^l_{RSU} + N^r_{RSU})$. The total number of vehicles N may change dynamically, and the CH is responsible for updating N and for informing all vehicles in the cluster of the new value of N.

The CH^l_i is the cluster head of ith cluster travelling in the left direction. CH^l_i is like a master vehicle that is allocated with a local-ID 1, which has the responsibility of allocating time slots to CMs and GVs of the cluster. Local ID 0 is not used. CH^l_i transmit the SI in the first time slot of the L set and as mentioned earlier SI contains all time slots for CMs and GVs. All CMs and GVs listen to the SI and transmit in the time slot allocated. SCP period contains $(N + 3)$ number of slots from 0 to $N + 2$ with $\lfloor \frac{N}{c} \rfloor$ channel cycles in each SCP. All slots are the same size, and the slot size τ is known to all vehicles in the cluster. The CMs, GVs and CH travelling in one direction takes the local-ID equal to the slot number of the previously allocated slot in the CCP.

Time slots in the SCP are used by GVs for inter cluster communication and mini slot in SCP are used by newly arriving vehicles for cluster management (cluster joining). In addition, CCP and SCP are divided into equal size time slots. Moreover, first time slot of all sets L, R, U in both CCP and SCP interval are partitioned into c mini slots. Mini slots on the CCP and SCP are exploited by newly arriving vehicles and RSUs to disseminate status, and safety messages. It is supposed that vehicles ideally remain on the CCH until they are granted a time slot in the CCP period. It is assumed that vehicles are aware of the frame and slot boundaries.

In each time frame, t is used by the vehicle or RSU to specify the channel and the channel cycle according to the following rules, where $0 < t < \left(\lfloor \frac{N}{c} \rfloor + 1\right) \times c$:

- Use channel $t \, mod \, c$ during channel cycle $\lfloor \frac{t}{c} \rfloor$. The basic idea is that in each logical frame, while idle, vehicle or RSU t listens to channel $t \, mod \, c$ in channel cycle $\lfloor \frac{t}{c} \rfloor$ and sets the corresponding byte in the CCH in order for other vehicles to be aware. Notice that the Integer Division Theorem guarantees that if $t \neq m$ then either:
- $\lfloor \frac{t}{c} \rfloor \neq \lfloor \frac{m}{c} \rfloor$
- $t \, mod \, c \neq m \, mod \, c$

This confirms that no two vehicles own the same channel in same channel cycle. For an illustration, let N = 54 and c = 6. Consider the number of vehicles moving in L and R direction N^l = 20, N^r = 30. The number of RSUs registered in both direction is N^{rsu} = 4. The local-ID for vehicles travelling in L direction is from 1 to 20, for vehicles in R direction is from 1 to 30, and for RSUs is from 1 to 4. As shown in Table 6.1, vehicle travelling in right direction with local ID 15 owns channel (15 *mod* 6) = 3 during channel cycle $\lfloor \frac{15}{6} \rfloor = 2$.

In the Table 6.1, the slots for vehicles travelling in L and R direction are given magenta and red colour, the slots for RSUs in both direction are given yellow colour. We note that for any given $N = N^l + N^r + N^{rsu}$, the number of unused slots in each

Table 6.1 Logical frames in DA-CMAC for N = 54 and c = 6

Channel/ Cycle	0	1	2	3	0	1	2	3	4	0
5	5	11	17	Unused	5	11	17	23	**30**	Unused
4	4	10	16	Unused	4	10	16	22	29	**4**
3	3	9	15	Unused	3	9	**15**	21	28	3
2	2	8	14	**20**	2	8	14	20	27	2
1	1	7	13	19	1	7	13	19	26	1
0	Reserved	6	12	18	Reserved	6	12	18	24	Reserved

SCP period is given by

$$\begin{aligned}(\left\lfloor \tfrac{N^l}{c} \right\rfloor + 1) \times c - 1 - N^l &= \left\lfloor \tfrac{N^l}{c} \right\rfloor \times c + c - 1 - N^l \\ &= N^l + c - 1 - (N^l \, mod \, c) - N^l \\ &= c - 1 - (N^l \, mod \, c), \quad (N^l \, mod \, c \neq 0)\end{aligned} \tag{6.9}$$

For $N^l \, mod \, c = 0$, the number of unused slots is 0. By using Eq. 6.9, we can calculate the number of unused slots in set L, R and U. The number of unused slots in set L from Eq. 6.9 is $6 - 1 - (20 \, mod \, 6) = 5 - 2 = 3$. Moreover, the number of slots in set R for $N^r = 30$. The value of $30 \, mod \, 6 = 0$, so the number of unused slot is zero. Lastly, for $N^{rsu} = 4$ is $6 - 1 - (4 \, mod \, 6) = 5 - 4 = 1$. The total number of unused slots is 4. These unused slots in channel cycles will be put to work in various ways that depend on the specific clustering regimen under investigation.

6.2.3 Transition from Undecided State to Cluster Member

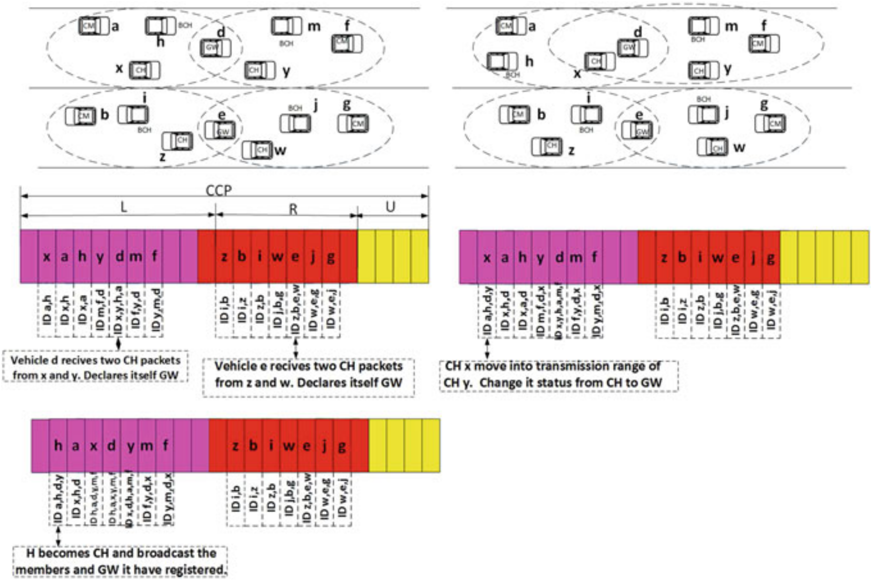
Those vehicles in the undecided state in both direction listen to the CCH channel c always during the initial approach in order to attain a slot in the CCP and SCP cycle. The protocol reduces transmission collisions when a vehicle in the undecided state try to join the nearest cluster. This scenario can occur when vehicles joining a highway from non highway road. The transmission collision causing due to vehicles travelling in both directions can be avoided by assigning disjoint sets of time slots to vehicles travelling in opposite direction and cluster. Now, suppose vehicle x is just entered the highway from a section road with no V2I or V2V communication and needs to join the cluster and acquire a time slot as a CM or GV.

An access collision happens when two or more vehicles in the undecided state within the transmission range of each other tries to access the same available mini-time slot. In DA-CMAC, the slot 0 of all sets is reserved and these slots are divided into k mini slots. Undecided vehicle x is travelling in L direction that wish to join the cluster i and require a slot in the L sets of the CCP and SCP. Given N_x^{cm}, N_x^{gv}, and N_x^{rsu} are the set of occupied slots of CM, GV and RSU in the direction of L. $N_x = N_x^{cm} \cup N_x^{gv} \cup N_x^{rsu}$. x will listen to at least one SCP and CCP cycle and transmit in one of the mini slot in the L set. By listening to the CCH channel 7 for N successive time slots (not necessarily in the same frame), vehicle x can determine set N_x and the time slot(s) used by either GVs or CHs in N_x. Given N_x, vehicle x determines the set of available time slots, A_x, (to be discussed) and then attempts to access the mini slot and ask for any time slot in A_x, say time slot k. If x receives a slot in the SI frame of the corresponding CH, then the slot access of the x is a success. After the transmission of SI by the corresponding CH, all the other CMs (e.g.; w) add x to its CM list and to the occupied slot list N_w and record the global-ID used by vehicle x to access time slot k, denoted by ID_x^k. Moreover, if x does not receive a slot in

SI packet then an access collision have occurred and x need to access the mini-slot again in the next CCP cycle. Once vehicle x acquires a time slot, it keeps using the same slot in all subsequent frames unless a merging collision or transition to CH or GV occur.

6.2.4 Transition from Cluster Member to Gateway Vehicle

In a highway, vehicles travel at different speeds, lanes, and move rapidly and two clusters may share an overlapping area for a certain time. When two clusters overlap together, we assume one vehicle is in between two clusters. If CM vehicle receives SI from more than one CH, then it will change its mode from CM to Gateway Vehicle (GV) mode. From Fig. 6.4, the vehicle d is member of cluster y initially. Then d receives SI from CH vehicle x without having a slot for itself in the information. d adds x to its list of CHs and add the other CMs of the newly received cluster with all its members to its two hop neighbour list to inform its one hop neighbours to release if any of the slots that are occupied by one hop neighbours. Then compare the available time slots in $A_{gv}(x) \cap A_{gv}(y)$. It selects a common slot based on the synchronization and an ID is generated by itself based on the slot number. Another scenario can occur, when x receives a message from d without inclusion of CH x. Then CH x identifies a common free slot in $A_{gv}(x) \cap A_{gv}(y)$ and selects a slot and allocate a local-ID with respect to the slot number. If they cannot find a common free slot among $A_{gv}(x) \cap$



Vehicle d recives two CH packets from x and y. Declares itself GW

Vehicle e recives two CH packets from z and w. Declares itself GW

CH x move into transmission range of CH y. Change it status from CH to GW

H becomes CH and broadcast the members and GW it have registered.

Fig. 6.4 Gateway vehicle and time slot

$A_{gv}(y)$ then it tries to find a common slot among $(A_{cm}(x) \cup A_{gv}(x)) \cap A_{gv}(y)$. Moreover, if it cannot find a slot among those slots, it tries to find a slot among $(A_{cm}(x) \cup A_{gv}(x)) \cap (A_{gv}(y) \cup A_{cm}(y))$. If it cannot find a common slot again, it tries to find a slot among $(A_{cm}(x) \cup A_{gv}(x) \cup A_{rsu}(x)) \cap (A_{gv}(y) \cup A_{cm}(y))$ if not then it tries to find from $(A_{cm}(x) \cup A_{gv}(x) \cup A_{rsu}(x)) \cap (A_{gv}(y) \cup A_{cm}(y) \cup A_{rsu}(y))$.

6.2.5 Transition from Cluster Head to Cluster Member

When CH vehicle x moves to transmission range of neighbouring CH y, the vehicles x or y may receive the SI packet first. The CH with lower number of connected members loses the CH status and elect the BCH as next CH of the cluster. Firstly, if x receives the SI from y first and compares the number of connected neighbours is higher than y, then x continues the role as CH of the cluster and transmits the SI. When y receives the SI information from x then y compares the number of connected neighbours with x. If y have less connected neighbours then y release its slot and allocate the BCH to its slot in the next cluster cycle and takes a slot in the GHL or GTL based on its position compared with newly elected CH. Secondly, if both x and y are having equal number of connected neighbours then the CH will keep the status if the average speed of the neighbours is less than the other. Moreover the vehicle that loses the CH status will assign BCH to it status.

6.3 Simulation Results

The network simulator ns-3.18 was used for evaluation. The wireless channel assumes Nakagami highway propagation model with 6 Mbit/s data rate and 10 MHz bandwidth at 5.9 GHz for all communication. A realistic highway traffic scenario is used in the evaluation a 10 km highway with two lanes in each direction and 1800 vehicles randomly distributed on these four lanes. The mobility models assume vehicle travel with a velocity between 20 and 40 m/s in the free traffic flow. After 2 min a sudden two directional traffic jam in the middle of the equipped road segment forces the velocity to drop to 8–10 m/s. For the next 3 min, vehicles queue in each direction. Afterwards, the traffic jam dissolves slowly for the next 4 min, restoring the original velocity distribution. Vehicles receive safety/update messages from other vehicles in both direction, but only relevant safety/update messages are taken into consideration for the further process. The frame length for DA-CMAC is 100 and 50 ms for both CCP and SCP. Ideally, the vehicle in DA-CMAC will be tuned to the CCH during the time interval; unless its own SCH slot time on the SCHs. Additional simulation parameters are depicted in Table 6.2.

Stable clusters are important for an efficient and reliable information transfer. Stable clustering methods reduce communication load of re-clustering and led to an efficient use of available bandwidth. Cluster stability depends on the selection

Table 6.2 Simulation setup

Simulation parameter	Value
Data rate	6 Mbit/s
Frequency	5.9 GHz
Transmission power	15 dBm
Highway length	10 km
Number of vehicles in each direction	900
Speed of vehicles	20–40 m/s
Number of clusters in each direction	10
Cluster radius	300 m
Propagation model	Nakagami
Safety packet size	200 bytes

of a suitable CH and cluster formation to ensure greater cluster residence times by reducing cluster change events. If vehicles are changing their mode very frequently and remain only for a short period of time in the CH state, stability of CH is low. As expected, the performance of DA-CMAC has varied depending upon the cluster radius. The number of nodes in the connected set decreases when the cluster radius decreases. Moreover, Fig. 6.5 shows the number of CH changes decreases when the cluster radius is increased for the variable speed, where a traffic jam is created in the middle of the road for few minutes.

The lower CH changes increases the Packet delivery rate (PDR). PDR is the best parameter to measure the stability of the cluster. In this paper, we define PDR as the total number of packets successfully received in CH divided by the total number of packets generated in CMs.

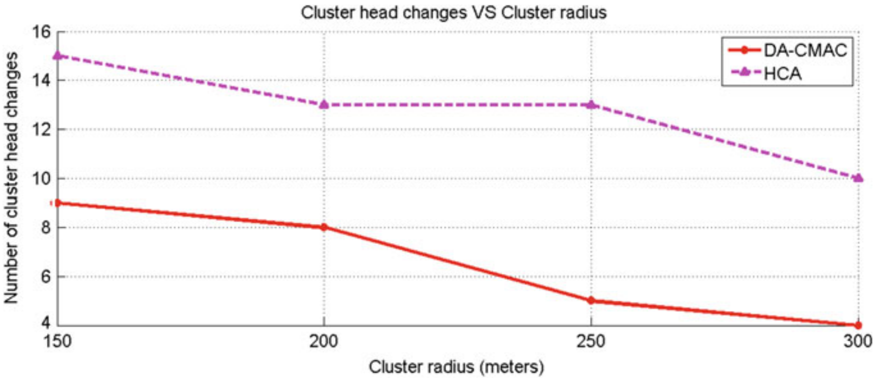


Fig. 6.5 Cluster head changes versus cluster radius

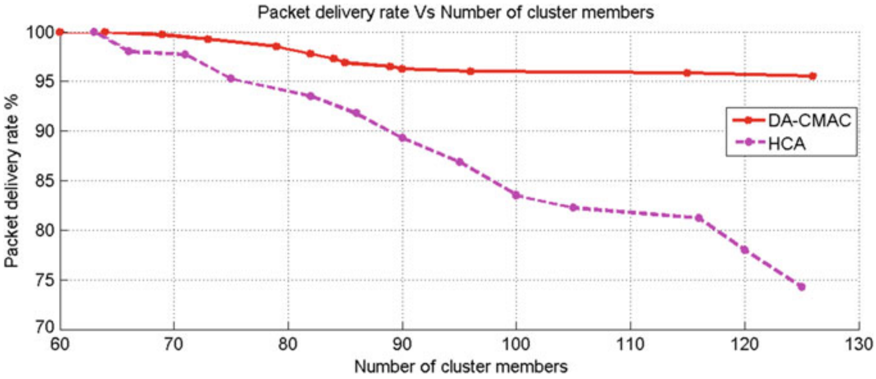


Fig. 6.6 Packet delivery rate versus number of CMs

From Fig. 6.6, the delivery rate decreases as the number of CMs increases. Moreover, the DA-CMAC and HCA [4] is almost ideal since it starts with a PDR which reaches about 100 % reception probability and presents sharp falls in the PDR for HCA. In case of DA-CMAC, the PDR is between 100 and 96 % and this may be due to the change of CHs. From Fig. 6.6, it can be seen that the performance of DA-CMAC exceeds the performance of HCA protocol. This is due to its feature of selecting a stable CH and a backup CH to take over the main CHs responsibilities when CHs value is higher than the threshold. The MAC of DA-CMAC protocol helps to maintain a high reliability and predictability compared to HCA, particularly in high-density networks.

The access collision rate is defined as the average number of access collisions that happen within a slot in one hop neighbour. The overall access collision rates of all the DA-CMAC and HCA protocols under different traffic densities are shown in Fig. 6.7. Access collisions in HCA increases with the increasing traffic density in

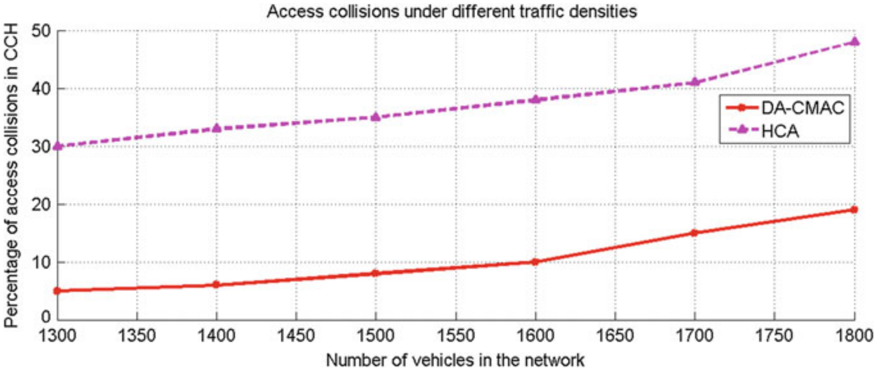


Fig. 6.7 Access collisions under different traffic densities

one hop neighbourhood. However, access collisions in DA-CMAC is not higher as compared to HCA due to the allocation of different sets of mini slots for vehicles travelling in opposite directions.

6.4 Conclusions

In this chapter, DA-CMAC protocol is presented in order to improve the reliability, predictability and scalability of VANETs. Clusters are formed based on the direction of travel and elect a stable CH based on the eligibility function. Moreover, the CH stability is further increased by assigning optimal dismiss threshold for cluster dismissal. Additionally, the CH checks the number of CMs in its transmission range in the next frame using the advertised speed of CM. Furthermore, this schedules the channel access by dividing time into cycles and each cycle is divided into CCP and SCP periods. CCP and SCP period is sub divided into three sets L, R, and U based on the direction of travel and characteristic of the node. CH manages the channel access and schedules transmission for all its CMs and GVs. The simulation results show that DA-CMAC has the lesser number of access collisions in the CCH and higher number of successful packets compared to HCA protocol. Simulations show that DA-CMAC has the lesser number of CH changes compared to HCA with different cluster radius.

Acknowledgments The authors would like to thank the EU Intelligent Cooperative Sensing for Improved traffic efficiency (ICSI) project (FP7-ICT-2011-8) for its support in the development of this work.

References

1. M.S. Almalag, S. Olariu, M.C. Weigle, TDMA cluster-based MAC for VANETs (TC-MAC), in *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2012, pp. 1–6. doi:10.1109/WoWMoM.2012.6263796
2. N. Alon, L. Babai, A. Itai, A fast and simple randomized parallel algorithm for the maximal independent set problem. Technical report, Chicago, IL, USA (1985)
3. F. Borgonovo et al., ADHOC: a new, flexible and reliable MAC architecture for ad-hoc networks, in *2003 IEEE Wireless Communications and Networking, 2003. WCNC 2003*, Mar 2003, vol. 2, pp. 965–970. doi:10.1109/WCNC.2003.1200502
4. E. Dror, C. Avin, Z. Lotker, Fast randomized algorithm for hierarchical clustering in vehicular ad-hoc networks, in *2011 The 10th IFIP Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, June 2011, pp. 1–8. doi:10.1109/Med-Hoc-Net.2011.5970488
5. F.V. Fomin et al., Combinatorial bounds via measure and conquer: bounding minimal dominating sets and applications. ACM Trans. Algorithms **5**(1), 9:1–9:17 (2008). ISSN: 1549-6325. doi:10.1145/1435375.1435384
6. Y. Gunter, B. Wiegel, H.P. Grossmann, Medium access concept for VANETs based on clustering, in *2007 IEEE 66th Vehicular Technology Conference, 2007. VTC-2007 Fall*, Sep 2007, pp. 2189–2193. doi:10.1109/VETECF.2007.459

7. H. Hartenstein, K.P. Laberteaux, A tutorial survey on vehicular ad hoc networks. IEEE Commun. Mag. **46**(6), 164–171 (2008). ISSN: 0163-6804. doi:10.1109/MCOM.2008.4539481
8. IEEE Guide for Wireless Access in Vehicular Environments (WAVE)—Architecture, inIEEE Std 1609.0-2013, Mar 2014, pp. 1–78. doi:10.1109/IEEESTD.2014.6755433
9. A.S.K. Mammu, U. Hernandez-Jayo, N. Sainz, Clusterbased MAC in VANETs for safety applications, in *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Aug 2013, pp. 1424–1429. doi:10.1109/ICACCI.2013.6637388
10. Y.-C. Lai et al., A region-based clustering mechanism for channel access in vehicular ad hoc networks. IEEE J. Sel. Areas Commun. **29**(1), 83–93 (2011). ISSN: 0733-8716. doi:10.1109/JSAC.2011.110109
11. N. Lu et al., A dedicated multi-channel MAC protocol design for vanet with adaptive broadcasting, in *2010 IEEE Wireless Communications and Networking Conference (WCNC)*, Apr 2010, pp. 1–6. doi:10.1109/WCNC.2010.5506242
12. H.A. Omar, W. Zhuang, L. Li, VeMAC: a novel multichannel MAC protocol for vehicular ad hoc networks, in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Apr 2011, pp. 413–418. doi:10.1109/INFCOMW.2011.5928848
13. Z.Y. Rawashdeh, S.M. Mahmud, Media access technique for cluster-based vehicular ad hoc networks, in *IEEE 68th Vehicular Technology Conference, 2008. VTC 2008-Fall*, Sep 2008, pp. 1–5. doi:10.1109/VETECF.2008.448
14. C. Shea, B. Hassanabadi, S. Valaee, Mobility-based clustering in VANETs using affinity propagation, in *IEEE Global Telecommunications Conference, 2009. GLOBECOM 2009*, Nov 2009, pp. 1–6. doi:10.1109/GLOCOM.2009.5425236
15. R. Uzcategui, G. Acosta-Marum, Wave: a tutorial. IEEE Commun. Mag. **47**(5), 126–133 (2009). ISSN: 0163-6804. doi:10.1109/MCOM.2009.4939288
16. X. Zhang, H. Su, H.-H. Chen, Cluster-based multi-channel communications protocols in vehicle ad hoc networks. IEEE Wirel. Commun. **13**(5), 44–51 (2006). ISSN: 1536-1284. doi:10.1109/WC-M.2006.250357

Chapter 7
Towards Predictable Vehicular Networks

Elad Michael Schiller

Abstract Communication primitives consider information delivery with different guarantees regarding their reliability. The provision of reliability and predictability needs to overcome a number of challenges with respect to failures and a number of known impossibility results. This chapter covers a number of these challenges in the context of vehicular systems and networks. We start by showing the medium access control (MAC) protocol for wireless mobile ad hoc networks can recover from timing failures and message collision and yet provide a predictable schedule in a time-division fashion without the need for external reference, such as commonly synchronized clock. We then consider the case of transport layer protocols and show how to deal with settings in which messages can be omitted, reordered and duplicated. We also consider how mobile ad hoc networks and vehicular networks can organize themselves for emulating virtual nodes as well as emulating replicated state-machines using group communication. In this context, we discuss the different alternatives for overcoming well-known impossibilities when considering cooperative vehicular applications. Finally, we exemplify applications and discuss their validation.

7.1 Introduction

Recent algorithmic developments provide enablers for designing and demonstrating cyber-physical vehicular systems for safety-critical applications that base their decisions on (uncertain) sensory information and yet function safely in presence of failures, such as (unbounded) communication delays. In this chapter, we review these recent developments and discuss their applications.

Cyber-physical vehicular systems require positioning information about the road layout, obstacles, hazards, nearby vehicles, road users to name a few. According to this information, the system cooperatively decides on its joint vehicular

E.M. Schiller (✉)
Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg, Sweden
e-mail: elad@chalmers.se

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_7

manoeuvres, which is often based on inter-vehicle exchange of sensory information and the combination between onboard and remote sources in order to reduce uncertainty, achieve completeness, provide dependability, as well as the integrate of perspectives and multi-dimensional viewpoints. The resulting (fused) sensory information is often better than what would be possible when merely using onboard sources. Recent developments in this field as well as sensory technology have opened the door for the automotive industry to consider using affordable sensors and inter-vehicle communications for cooperative safety-critical applications. Unfortunately, sensory information that originates from affordable onboard sensors increases significantly the position information uncertainty. Moreover, since inter-vehicle communications are prone to unbounded delays, their use implies arbitrary disconnection from remote information sources. Consequently, the current practice mostly relies on onboard sources rather than leveraging on remote sensory sources.

There are known methods for estimating the quality sensor information sources as well as their fused result. Based on these estimations, the cooperative vehicular systems are to choose, within a real-time constraint, exactly one functionality out of a set of (distributed) vehicular functionalities. We call this selected functionality the *operation level* of the (cooperative) vehicular system. Operational deadlines are imperative constraints of any safety-critical (cooperative) system. The functionality set could be, for example, the implementation of vehicular platooning. Here the system has to decide on the headings (or the inter-vehicle distances) and whether the vehicles are allowed to accelerate as well as to what extent so that a safe and comfort ride is guaranteed. The choice could be based, for instance, on the common uncertainty bound that all system vehicles can support. We call this bound the *information validity level* and point out that the selection of a single functionality can encode a complex formations, manoeuvres and behaviour in which each vehicle can anticipate the behaviour of nearby vehicles, with respect to the bounds that they use for estimating the uncertainty of the sensory information.

Some of the existing cooperative vehicular systems are based on *implicit communication* methods in which the vehicles use onboard sensors for anticipating the approaching vehicle intentions and by that aim at making sure that all vehicles indeed agree on who has the right to cross the intersection. Note that both implicit and explicit (inter-vehicle) communications are prone to interferences. However, the latter approach allows each vehicle to inform nearby vehicles about its (immediate) intensions. Virtual Traffic Lights (VTLs), for example, can use explicit communication for dynamically scheduling their green light phases. The challenges here include the need to facilitate (successive and) coherent decisions in the presence failures, such as (unbounded) communication delays. For instance, VTLs can only give a green light for vehicles coming from one direction after it gave red to all conflicting directions. Moreover, intermediate yellow periods are required between red and green periods. One of the key obstacles that we plan to overcome is the impossibility to decide uniformly in the presence of failures, such as (unbounded) communication delay [1, 2]. In other words, the problem to circumvent here is how not to allow communication failures to create a split brain phenomena in which, in the case of virtual traffic lights, different vehicles decides on different traffic light schedules.

One may seek the solution for such problems by using infrastructure-based services, such as group communication systems. These services can help to follow the presence of vehicles, i.e., location and membership. Note that such systems were also proposed for ad hoc networks [3]. In this chapter, we consider enables that follow both approaches.

We review recent results about new ways to circumvent the impossibility to choose uniformly a single value, i.e., the operation level of the cooperative system (Sect. 7.5). By that we can to significantly simplify the design of complex cooperative functionalities, for example, when scheduling lane changes, intersection crossing and going through roundabouts. The challenge referred to here is how to design (distributed) synchronization (control) mechanisms that can support successive cooperative decisions that are based on uncertain sensory information and perform coherently in presence failures, such as (unbounded) inter-vehicle communication delay.

7.1.1 The Self-stabilization Design Criteria

Large and dynamic networks are hard to control and it is challenging to provide network protocols with predictable behaviour. Very important design criteria for the implementation of communication services are their fault tolerance and robustness. However, the distributed algorithms that implement these communication protocols often assume a particular set of possible failures, such as crash failures, link failures, or message loss. The correctness of these implementing algorithms is proved by assuming a predefined initial state and considering every possible execution that involves the assumed possible set of failures. This abstraction, which limit the set of possible failures allows a more convenient correctness demonstration. However, it is also too restrictive. Communication protocols are often long-lived, on-line services for which it is hard to predict in advance the exact set of possible faults. Moreover, when communication delays are unbounded, and the chances for packet drop are high, it may be the case that due to the occurrence of an unexpected fault the system reaches a state that is not attainable from the initial state by steps of the algorithm and the occurrence of the assumed fault model. Therefore, self-stabilizing systems [4, 5] can be started in any arbitrary state, and they will exhibit the desired behavior after a convergence period.

Another important benefit of the self-stabilizing design criteria is the system ability to offer automatic recovery from unexpected transient failures, such as a temporary violation of the assumptions that were made by the system designers. As an illustrative example, let us consider the use of probabilistic error detection codes for ensuring that the arriving packets are identical to the ones sent. It can happen that the error detection eventually fails to detect a corrupted message. The system then regards a corrupted message as a legitimate one. This might bring the system to an arbitrary state for which, in the case of non-self-stabilizing systems, there are no guarantees for service functionality and availability (without human intervention).

7.1.2 Chapter Roadmap

This chapter covers a number of challenges in the context of vehicular systems and networks. We start by showing that medium access control (MAC) protocols for wireless mobile ad hoc networks can recover from timing failures and message collision and yet provide a predictable schedule in a time-division fashion without the need for external reference, such as commonly synchronized clock (Sect. 7.2). We then consider the case of transport layer protocols and show how to deal with networks in which messages can be omitted, reordered and duplicated (Sect. 7.3). We also consider how mobile ad hoc networks and vehicular networks can organize themselves for emulating virtual nodes as well as emulating replicated state-machines using group communication services (Sect. 7.4). In this context, we discuss the different alternatives for overcoming well-known impossibilities when considering cooperative vehicular applications (Sect. 7.5). Finally, we exemplify applications (Sect. 7.6) and discuss their validation (Sect. 7.7).

7.2 Self-stabilizing MAC for Wireless Ad Hoc Networks

One of the key enablers of predictive communication is having a media access control (MAC) layer, with predictable behaviour. Namely, after a convergence period and in the absence of external interferences, each node should be able to access the network within a bounded communication delay. We discuss several recent development that allows the system to increase their predictability degree when using wireless ad hoc networks.

Mustafa et al. [6], Leone et al. [7–9] and Petig et al. [10] suggest algorithmic designs for stabilizing MAC algorithms with an emphasis is on providing resilience and predictability. Such algorithms are required for ad hoc vehicular networks.

In the context of predictable vehicular ad hoc network (VANET), which have frequent topological changes, MAC protocols need to be self-stabilizing, have low communication delays and high bandwidth utilization. We propose a self-stabilizing MAC algorithm that guarantees to satisfy these severe timing requirements.

In the context of TDMA, the timing alignment of packet transmissions helps to avoid transmission interferences. Existing VANET implementations often assume the availability synchronized clocks, e.g., GPS signals. Mustafa et al. [6] consider autonomic design criteria, and present a (probabilistic) self-$\star$ broadcasting timeslot alignment for ad hoc wireless networks that follow the time division multiple access (TDMA) approaches. In these networks, the radio time is divided into timeslots, wherein each such timeslot a subset of the network node are allowed to transmit, such that the interference degree among of concurrent transmissions among the nodes in these subsets in kept low. The algorithm by Mustafa at al. [6] can make sure that these timeslots are well aligned.

Leone et al. [8] assumes such timeslot alignment and present a (probabilistic) self-stabilizing algorithm for scheduling transmissions among nodes, such that no two neighbouring nodes transmit concurrently. The authors prove a rapid stabilization, and by that, the algorithm allows greater degree of predictability, while maintaining high throughput and low communication delays.

During the stabilization period, several nodes can have the same assigned timeslot. The algorithm solves such timeslot allocation conflicts via a (listening/signalling) competition in which node p_i and node p_j participate before transmitting in their broadcasting timeslots. The competition requires p_i and p_j to select one out of n listening/signalling periods their timeslots. Note that among all the nodes that aim at broadcasting in a particular timeslot, the ones who win and access the communication media are the ones that select the earliest listening/signalling period. Prior to accessing their timeslots, the winners inform to their neighbourhoods about their win by broadcasting beacons during their selected signalling periods. Upon beacon reception, a node defer from transmitting during that timeslot, since it lost the competition. After a back-off period, the losing nodes compete on the next broadcasting round for their new timeslots by selecting them randomly.

Petig et al. [10] present protocols for dynamic wireless ad-hoc networks that allocates the timeslot while considering the transmission timing aspects of the problem. They show that the solution existence depends on the ratio, τ/δ, between the frame size τ (which is the number of timeslot in each TDMA frame) and the node degree δ in the communication graph. They prove that $\tau/\delta \geq 2$ is required for any (eventually) collision-free TDMA algorithms and present a (probabilistic) algorithm for the case of $\tau/\delta \leq 4$.

The exposure period of a packet is the period during which it may be co-transmitted with other packets from nearby nodes. In the absence of an external reference, the TDMA algorithm has to concurrently align timeslots while allocating them. Petig et al. [10] show that $\tau/\delta \leq 4$ is sufficient for guaranteeing zero exposure period with respect to a single timeslot, s, and a single receiver, rather than all neighbouring transmitters. The algorithm considers nodes that transmit data and control packets. Data packets are sent by active nodes during their data packet timeslots while passive nodes listen to the actives ones and do not transmit. Both active and passive nodes use control packets, which include frame information about the recently received packets. The algorithm uses this information for avoiding collisions, acknowledging packet transmission and resolving hidden node problems.

7.3 Self-stabilizing End-to-End Protocols

End-to-end communication protocols is an important reliable communication enabler for any type of network, including (vehicular) mobile ad hoc networks. The end-to-end protocol handles the exchange of packets between pair of network nodes, which do not necessary communicate directly and thus need the assistant of relay nodes. Where as the relay nodes perform the packet forwarding operations, it is up to the nodes at the forwarding path ends to make sure that the packet are delivered to their destination in exactly the same order in which the source has sent them.

Network protocols use techniques, such as retransmissions and multi-path routing, to increase their robustness and over packet omissions. These techniques can create anomalies, such as packet reordering and duplication. Dolev at el. [11] present end-to-end algorithms for dynamic networks that makes sure that the receiver at the destination node delivers the same sequence of (high level) messages. The algorithm can be applied to networks for which there is a bound on the number of packets that can reside in them their capacity that omit, duplicate and reorder packets.

We outline the algorithm's basic ideas while sketching an algorithm with a large message overhead. The detailed algorithm uses error correction codes and has a smaller overhead, see [11] for details and [12] for a self-stabilizing end-to-end protocol in the presence of Byzantine nodes.

7.3.1 The Algorithm Sketch

Let us consider a sender, p_s, and receiver p_r nodes. Node p_s needs to fetch messages and send them to p_r, which in turn needs to deliver m the order in which it was sent. Once p_s fetches m, it starts transmitting $2 \cdot capacity + 1$ copies of it, and p_r acknowledges them, where $capacity$ is the network capacity. These transmissions use labels that are distinct from each copy. The sender does not stop retransmitting till it receives from p_r $(capacity + 1)$ different labeled acknowledgments, where the majority of them are copies of m. Namely, p_s maintains an alternating index, $AltIndex \in [0, 2]$, which is a three state counter that is incremented upon fetching a new message. Moreover, p_s transmits a set of packets, $\langle ai, lbl, dat \rangle$, where $ai = AltIndex$, and lbl are packet labels. This transmission ends once p_r receives a packet set, $\{\langle 0, \ell, dat \rangle\}_{\ell \in [1, 2 \cdot capacity+1]}$, that is distinctly labeled by ℓ with respect to the alternating index. After recovering from any transient failure, the set of received packets includes a majority of packets that have the same value of dat. When that happens, p_r delivers m and updates the value of the last delivered alternating index.

The correct packet transmission depends on the synchrony of m's alternating index at the sending-side, and $LastDeliveredIndex$ on the receiver side, as well as the packets that p_r accumulates in $packet\_set_r$. Node p_s repeatedly sends its packet set until it receives $(capacity + 1)$ distinctly labeled acknowledgment packets, $\langle ldai, lbl \rangle$, for which it holds that $ldai = AltIndex$. Node p_r acknowledges each incoming packet, $\langle ai, lbl, dat \rangle$, using acknowledgment packet $\langle ldai, lbl \rangle$, where $ldai$ refers to the value of the last alternating index, $LastDeliveredIndex$. That that the receiver also delivers this packet.

On the other-side, node p_r delivers $m = \langle dat \rangle$, from one of the $(capacity + 1)$ distinctly labeled packets that have identical dat and ai values. Then, p_r assigns ai to $LastDeliveredIndex$, empties its packet set and restarts accumulating packets, $\langle ai', lbl', dat' \rangle$, for which $LastDeliveredIndex \neq ai'$.

7.4 Self-stabilizing Group Communication

Group communication systems provide high level communication primitives that enable nodes that share a collective interest, to identify themselves as a single logical communication endpoint. Each such endpoint is named a group, and each group has a unique group identifier. Nodes may join a leave the group and it is up to the membership service to provide the current group view that is uniquely identified. The view, thus, includes the group membership set and the view identifier. The group multicast service can the multicast messages the group and collect its acknowledgement after its delivery to all of the view members. After reviewing the relevant self-stabilizing literature on group communication, we present solutions that are dedicated to mobile ad hoc (vehicular) networks and vehicular networks that also include communication infrastructure.

7.4.1 Infrastructure-Based Approaches

The first algorithmic design of self-stabilizing group communication system includes the ones for undirected [13] and directed networks [14]. The proof of correctness demonstrates convergence after the last transient fault, such as a crash failure or any other topological change to the network graph. In the presence of failures, such as fail-stop crashes and unbounded communication delays, it is not possible to guarantee message delivery to all members of the sending-view. The property of virtual synchrony [15] allows all system events, view changes and multicast messages, to be delivered in the same order. The virtual synchrony property requires that any two nodes that are members of two consecutive views of communicating groups shall deliver the set of system events, e.g., multicast messages. This property makes it easier to vehicular applications that, for example, are based on state-machine replication [16–18].

Recently, it was shown how to design a self-stabilizing group communication system and how that system can emulate state-machine replication [19] using a self-stabilizing emulator of multi-reader multi-writer registers over message passing systems, similar to the ones by [20, 21]. The emulation of a replicated state-machine is a way to let the nodes to periodically exchange their current state and their current input by sending multicast messages. Once all multicast messages and their acknowledgements are delivered, the nodes can verify that they all share the same state, apply the new input to the current state and by that get the new state.

7.4.2 Infrastructure-Less Approaches

Self-stabilizing state-machines replication is also in the heart of high-level communication primitives, such as virtual (mobile) nodes [22–25]. The idea is to emulate replicates in geographic regions, say, by tilling the area, such that each tile has its own replicated stationary automata [26]. Any nodes that enters a tile starts emulating the tile's automata in the manner of state machines replication. Virtual mobile nodes consider the case in which the tiles are moving according to a deterministic function that is based on time, as in [24, 25] or also the environment input [22, 23]. The design of this virtual infrastructure has inspired the idea about virtual traffic light in a junction as well as other applications, which we discuss in Sect. 7.6.

We note that the first algorithmic design for (self-stabilizing) group communication systems for ad hoc networks [27] was mobile agents, collecting and distributing information, during their many random walks. They eventually elect a single agent that is the basis of the group membership and multicast services, i.e., it collects and distributes information. Several systems were developed based on this approach, such as RaWMS [28] and Pilot [29].

7.5 Vehicular Coordination in the Presence of Failures

Vehicular networks are the basis for providing high-performmance cooperative vehicular systems while assuring safety standards. The vehicles determine their maneuver strategies according remote sensory information together with its quality, i.e., information validity. Since radio communications are prone to failure, it is unclear how to assume a joint awareness of timely message reception. Using such joint awareness, the planning of conflict-free trajectories becomes earlier. Morales et al. [30, 31] present a timed deterministic communication protocol that facilitate cooperative vehicular functionality in the presence of failures. After reviewing their protocol and its related properties, in this paper we discuss the protocol application for cooperative (vehicular) systems. Such applications, can facilitate the development of automated driving system, which have to work around the (communication) uncertainties that failure-prone communications bring in.

7.5.1 Problem Description

Vehicle-to-vehicle communications are the way to allow automated driving system to become affordable by raising the confidence level on the sensory information. This information confederate is at the heart of advanced cooperative (vehicular) functionalities, such as lane changing and intersection crossing, as well as busting the road capacity [32]. It is imperative that such cooperative system are able to deal with communication failures while maintaining safety in all hazardous situations.

Morales et al. [30, 31] consider a communication protocol for exchanging messages among vehicles. When planning the vehicle trajectories, a control algorithm uses this communication protocol together with the remote and onboard sensory information. The process of trajectory planning becomes simple when all vehicles can use the (in general vectorial) variable LoS (level of service). The correctness of the cooperation algorithm depends on this common information resource, LoS. Let us consider an example in which vehicular platooning, LoS might include the maximum acceleration, which is due to the vehicle's braking limits [33].

The problem of distributed (uniform) consensus is a related problem to that one studied by Morales et al. [30, 31]. Both problems consider a set of values proposed by the nodes, i.e., the system vehicles, and the uniform selection of a single value from such sets. Since vehicular systems is a safety critical one, this section must terminate within a time constraint that is more severe than the bounded achievable for different versions of the uniform consensus problem. The safety analysis is greatly simplified when using exact and deterministic solutions, in contrast to the approximate consensus [34].

It is well-known that unbounded communication delays, say, due to packet drop, can defer reaching a uniform consensus about LoS's value, and the literature includes a several related negative results [2, 35, 36]. Lynch [37] shows that communication failures can prevent the system from reaching consensus deterministically. Moreover, any probabilistic algorithm that takes no more than r rounds reaches a non-uniform decision with a probability of at least $\frac{1}{r+1}$. Thus, there are no assurance to reach uniform consensus within a deadline, because radio communications are prone to failures. Therefore, we cannot hope for achieving a solution that is based on protocols that provide uniform consensus. Thus, when there is a need to establish a new value of s, it is imperative not to let the communication network, which may present non-uniform values of S to different vehicles, to lead the cooperative vehicular system an unsafe operation.

Uniform consensus algorithms with real-time constraints often assume timed and reliable communication [38, 39]. Morales et al. [30, 31] do not assume reliable communication. They present the problem of minimum longest uncertainty period. The problem considers deterministic solutions for eventually deciding on the LoS. Unlike the uniform consensus problem, the LoS needs to repeatedly decided on; once in every synchronous round. Moreover, there could be a period, called the uncertainty period, during which different nodes output different LoS values. This problem asks what is the longest period in which the different vehicles can consider different performmance levels. Note that by Lynch [37], this bound cannot be zero. Namely, the vehicles are at risk to disagree when, for example, some nodes miss receiving the required information by the deadline of a synchronous round. However, at the end of the uncertainty period, all nodes need to agree on the same value.

7.5.1.1 The Solution Approach and Key Concepts

Morales et al. [30, 31] present a communication protocol that collects LoS proposals from all system vehicles. When a vehicle receives proposals from all vehicles in the system, the protocol can decide deterministically on a single proposal. The protocol identifies the risky periods due to transient failures, i.e., a period in which not all LoS proposals were received in due time by every node in the system. Upon risk identification, the protocol triggers a strategy that deals with possible disagreement about the LoS value.

The system settings assume the availability of a membership service that returns the set of all system nodes (vehicles), $P = \{p_i\}_{i \in \{1,\ldots n\}}$, such as [13, 14, 19, 27]. They assume the availability of a common clock and the division of the execution to communication rounds. They also assume the availability of an unreliable dissemination protocol, such as [40, 41], that its messages can reside in the network for a bounded duration.

The system also considers vehicular applications that are based on a set of cooperative functionalities out of which the protocol is to select one, which is the LoS. The assumption here is that this set of cooperative functionalities always include a baseline functionality that is always safe. For example, in the case of Adaptive Cruise Control (ACC) and Vehicular Platooning, the application has to adjust the inter-vehicular distance by controlling the velocity of the different vehicles. ACC uses its sensory information for keeping that distance while considering on the vehicles that are in its direct line-of-sight. A Vehicular platooning application can be seen as an advanced (cooperative) ACC that is based not only on providing sensory information from remote sensors but also having a joint control strategy that allows shorter inter-vehicle distance, as long as communication is available. Morales et al. [30, 31] show how to deal with communication failures and assumes the existence of a baseline application such as ACC in the above set of cooperative functionalities.

The Morales et al. [30, 31] protocol let the system nodes to gossip their LoS proposals until the deadline at the end of each communication round k. Once all nodes receive by all node, they can locally and deterministically select a single LoS proposal for round k. In case of a communication failure during round k, each node p_i that experience a failure, outputs the baseline application as its LoS for round k and reports about the failure during the next communication round, $k + 1$. During that round each other node, p_j, either receives p_i's report about the failure during round k or experience a failure during round k. In both cases, p_j outputs the baseline application as its LoS for round $k + 1$. The correctness proof demonstrates that, in the presence of at least one communication round, there could be at most one disagreement round in which different vehicles follow different LoS values. Moreover, once the network become stable and all communication rounds allow any pair of nodes to exchange messages in a timely manner, the system returns to select an LoS value of the sent proposals.

7.6 Example Applications

We discussed a number of functionalities that are based on the protocol by Morales et al. [30, 31]. We consider both functionalities that support key system enables and applications of vehicular coordination algorithms.

7.6.1 Supporting Functionalities

Casimiro et al. [42, 43] present the *safety kernel* as an architectural concept that allows a cooperative vehicular system to decide on a common performmance level based on the information validity level of the different vehicles. The selection of the performmance level is based on a common service level value, LoS. The protocol by Morales et al. [30, 31] can be the basis of the joint LoS selection and by that implement an architectural component that is called *cooperative evaluator of service level* [42, 43]. At each round, each vehicle proposes the maximum service level that it can support. The cooperative evaluator of service level uses the protocol for collecting and selecting a joint LoS value. This value is the basis for selecting a common performmance level during the next communication round. Note that the possibility to disagree on the joint LoS value (and hence the performmance level) could last for a bounded period of time, which is just one round in the common case of a single hop network.

Once can imagine an infrastructure-based support for the safety kernel on the network level. Stations for intelligent transportation system (ITS-stations) and their global and local dynamic maps [44, 45] can be the providers of critical information that is needed for cooperative vehicular systems. In particular, one can consider an extension of the ETSI standard for ITS stations that will also include the position of nearby vehicles. Moreover, using concepts similar to the ones of safety kernel and cooperative evaluator of service level the ITS station would be able to facilitate a joint choice of the operation level in a safe manner.

7.6.2 Cooperative Vehicular Functionalities

Adaptive Cruise Control and Vehicle Platooning are two applications that are part of a set of applications in which vehicles control their headway (inter-vehicle distance) by adjusting their velocity according to their join performmance level. Casimiro et al. [42, 43] propose how to base the choice of the performmance level using a safety kernel architecture [46–48]. Namely, the safety kernel indirectly decides on the headway and chooses the performmance level that all vehicles can support. For example, the safety kernel creates a longer larger inter-vehicle distance whenever at least one vehicles cannot support the current performmance level. In contrast, whenever the performmance level that all vehicles can support recovery to a higher level, the safety kernel shorten the headway.

Casimiro et al. [42, 43] also propose how to use the safety kernel for coordinating intersection crossing. In this application, vehicles from conflicting direction need to schedule their exact arrival time to the intersection boarder so that they can safely enter and leave the intersection. Here, at the performmance highest level, the vehicles cross the intersection with minimal waiting time while in the lowest level the vehicles take caution by, say, always give way to the vehicle coming from the right. Note that there is no need for explicit agreement on an ad hoc schedule, but rather to base the coordination on the needed safe headway and some predefined rules, such as direction priority, although both approaches can work.

Casimiro et al. [42, 43] also propose the application of coordinated lane-change. The coordination algorithm adjusts the inter-vehicle distances at the target lane using the safety kernel. Whenever the performmance level is high, the application performers the maneuver while keeping a shorter inter-vehicle distance than it maintains in lower performmance levels.

7.7 Conclusions

We have revised some of the recent advances in the area of distributed vehicular systems. The review considers different communication later, design criteria, such as self-stabilization, as well as infrastructure and ad hoc approaches. These vehicular system and networks require extensive validation. Pahlavan et al. [49] and Berger et al. [50] propose combining digital simulation together with the use of a cyber-physical platform that include scaled vehicles. This way, the system designer can gradually demonstrate the system properties. In particular, in can demonstrate the system in a relevant environment before requiring the more costly demonstration using full-scale vehicles in the representative environment.

References

1. N.A. Lynch, *Distributed Computing* (Morgan Kaufmann Publishers, 1996). ISBN: 1-55860-348-4
2. M.J. Fischer, N.A. Lynch, and M.Paterson (ed.), Impossibility of distributed consensus with one faulty process. J. ACM **32**(2), 374–382 (1985)
3. Friedhelm Meyer auf der Heide, C.A. Phillips (eds.), *Best-Effort Group Service in Dynamic Networks*, in *SPAA 2010: Proceedings of the 22nd Annual ACM Symposium on Parallelism in Algorithms and Architectures, Thira, Santorini, Greece, June13-15, 2010*, (ACM, 2010), pp. 233–242. ISBN: 978-1-4503-0079-7
4. E.W. Dijkstra, Self-stabilizing systems in spite of distributed control. ACM Commun. **17**(11), 643–644 (1974). doi:10.1145/361179.361202
5. S. Dolev, *Self-Stabilization* (MIT Press, Cambridge, 2000)
6. *Autonomous TDMA Alignment for VANETs*, in *Proceedings of the 76th IEEE Vehicular Technology Conference, VTC Fall 2012, Quebec City, QC, Canada, September 3-6, 2012*, (IEEE, 2012), pp. 1–5. ISBN: 978-1-4673-1880-8. doi:10.1109/VTCFall..6399373
7. P. Leone, M. Papatriantafilou and E.M. Schiller, *Relocation Analysis of Stabilizing MAC Algorithms for Large-Scale Mobile Ad Hoc Networks*, Lecture Notes in Computer Science, vol. 5804 (Springer, 2009), pp. 203–217. ISBN: 978-3-642-05433-4. doi:10.1007/978-3-642-05434-1_21
8. P. Leone, E. Schiller, Self-stabilizing TDMA algorithms for dynamic wireless ad hoc networks. Int. J. Distrib. Sens. Netw. **2013** (2013). doi:10.1155/2013/639761
9. P. Leone, M. Papatriantafilou, E. M. Schiller, and G. Zhu (eds.), *Chameleon-MAC: Adaptive and Self-* Algorithms for Media Access Control in Mobile Ad Hoc Networks*, in *Stabilization,*

Safety, and Security of Distributed Systems - 12th International Symposium, SSS 2010, New York, NY, USA, September 20-22, 2010. Proceedings, Lecture Notes in Computer Science, vol. 6366 (Springer, 2010), pp. 468–488. ISBN: 978-3-642-16022-6. doi:10.1007/978-3-642-16023-3_37

10. T. Petig, E. Schiller, and P. Tsigas *Self-stabilizing TDMA Algorithms for Wireless Ad-hoc Networks without External Reference*, in *13th Annual Mediterranean Ad Hoc Networking Workshop, MED-HOC-NET 2014, Piran, Slovenia, June 2-4, 2014*, (IEEE, 2014), pp. 87–94. ISBN: 978-1-4799-5258-8. doi:10.1109/MedHocNet.2014.6849109
11. S. Dolev, A. Hanemann, E. M. Schiller, and S. Sharma (eds.), *Self-stabilizing End-to-End Communication in (bounded capacity, omitting, duplicating and non-fifo) Dynamic Networks*, in *Stabilization, Safety, and Security of Distributed Systems - 14th International Symposium, SSS 2012, Toronto, Canada, October 1-4, 2012. Proceedings*, Lecture Notes in Computer Science, vol. 7596 (Springer, 2012), pp. 133–147. ISBN: 978-3-642-33535-8. doi:10.1007/978-3-642-33536-5_14
12. S. Dolev, O. Liba, and E. M. Schiller (eds.), *Self-stabilizing Byzantine Resilient Topology Discovery and Message Delivery*, in *Networked Systems - First International Conference, NETYS 2013, Marrakech, Morocco, May 2-4, 2013, Revised Selected Papers*, Lecture Notes in Computer Science, vol. 7853 (Springer, 2013), pp. 42–57. ISBN: 978-3-642-40147-3. doi:10.1007/978-3-642-40148-0_4
13. S. Dolev, E. Schiller, Communication adaptive self-stabilizing group membership service. IEEE Trans. Parallel Distrib. Syst. **14**(7), 709–720 (2003). doi:10.1109/TPDS.2003.1214322
14. S. Dolev, E. Schiller, Self-stabilizing group communication in directed networks. Acta Informatica **40**(9), 609–636 (2004). doi:10.1007/s00236-004-0143-1
15. K.P. Birman, R. van Renesse et al., *Reliable Distributed Computing with the Isis Toolkit*, vol. 85 (IEEE Computer Society Press, Los Alamitos, 1994)
16. A. Bartoli, Implementing a replicated service with group communication. J. Syst. Architect. **50**(8), 493–519 (2004). doi:10.1016/j.sysarc.2003.11.003
17. K. Birman (ed.), *A History of the Virtual Synchrony Replication Model*, in *Replication: Theory and Practice*, Lecture Notes in Computer Science, vol. 5959 (Springer, 2010), pp. 91–120. ISBN: 978-3-642-11293-5. doi:10.1007/978-3-642-11294-2_6
18. R. Khazan, A. Fekete, and N. A. Lynch (eds.), *Multicast Group Communication as a Base for a Load-Balancing Replicated Data Service*, in *Distributed Computing, 12th International Symposium, DISC '98, Andros, Greece, September 24-26, 1998, Proceedings*, Lecture Notes in Computer Science, vol. 1499 (Springer, 1998), pp. 258–272. ISBN: 3-540-65066-0. doi:10.1007/BFb0056488
19. S. Dolev, C. Georgiou, I. Marcoullis, and E. M. Schiller (eds.), *Practically Stabilizing Virtual Synchrony*, in *Stabilization, Safety, and Security of DistributedSystems - 17th International Symposium, SSS 2015, Edmonton, Canada, August 18-21, 2015. Proceedings*, Lecture Notes in Computer Science (Springer, 2015)
20. S. Dolev, T. Petig, and E. M. Schiller (eds.), *Brief Announcement: Robust and Private Distributed Shared Atomic Memory in Message Passing Networks*, in *Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing, PODC 2015, Donostia-San Sebastián, Spain, July 21 - 23, 2015*, (ACM, 2015), pp. 311–313. ISBN: 978-1-4503-3617-8. doi:10.1145/2767386.2767450
21. R. Fan and N. A. Lynch (eds.), *Efficient Replication of Large Data Objects*, in *Distributed Computing, 17th International Conference, DISC 2003, Sorrento, Italy, October 1-3, 2003,Proceedings*, Lecture Notes in Computer Science, vol. 2848 (Springer, 2003), pp. 75–91. ISBN: 3-540-20184-X. doi:10.1007/978-3-540-39989-6_6
22. S. Dolev, S. Gilbert, E. Schiller, A. A. Shvartsman, and J. L. Welch (eds.), Autonomous Virtual Mobile Nodes, in *Joint Workshop on Foundations of Mobile Computing (DIALM-POMC)*, (2005), pp. 62–69. ISBN: 1-58113-986-1. doi:10.1145/1073970.1074004
23. S. Dolev, S. Gilbert, E. Schiller, A. A. Shvartsman, and J. L. Welch (eds.), Autonomous Virtual Mobile Nodes, in *SPAA 2005: Proceedings of the 17th Annual ACM Symposium on Parallelism*

in Algorithms and Architectures, July 18-20, 2005, Las Vegas,Nevada, USA (ACM, 2005), p. 215

24. S. Dolev, S. Gilbert, N. A. Lynch, E. Schiller, A. A. Shvartsman, and J. L. Welch (eds.), *Brief Announcement: Virtual Mobile Nodes for Mobile Ad Hoc Networks*, in *Proceedings of the Twenty-Third Annual ACM Symposium on Principles of Distributed Computing, PODC 2004, St. John's, Newfoundland, Canada, July 25-28, 2004*, (ACM, 2004), p. 385
25. S. Dolev, S. Gilbert, N. A. Lynch, E. Schiller, A. A. Shvartsman, and J. L. Welch (eds.), *Virtual Mobile Nodes for Mobile Ad Hoc Networks*, in *Distributed Computing, 18th International Conference, DISC 2004, Amsterdam, The Netherlands, October 4-7, 2004, Proceedings*, Lecture Notes in Computer Science, vol. 3274 (Springer, 2004), pp. 230–244. ISBN: 3-540-23306-7. doi:10.1007/978-3-540-30186-8_17
26. S. Dolev, S. Gilbert, L. Lahiani, N. A. Lynch, and T. Nolte (eds.), *Timed Virtual Stationary Automata for Mobile Networks*, in *Principles of Distributed Systems, 9th International Conference, OPODIS 2005, Pisa, Italy, December 12-14, 2005, RevisedSelected Papers*, Lecture Notes in Computer Science, vol. 3974 (Springer, 2005), pp. 130–145. ISBN: 3-540-36321-1. doi:10.1007/11795490_12
27. S. Dolev, E. Schiller, J.L. Welch, Random walk for self-stabilizing group communication in ad hoc networks. IEEE Trans. Mob. Comput. **5**(7), 893–905 (2006). doi:10.1109/TMC.2006.104
28. Z. Bar-Yossef, R. Friedman, G. Kliot, RaWMS—random walk based lightweight membership service for wireless ad hoc networks. ACM Trans. Comput. Syst. **26**(2) (2008). doi:10.1145/1365815.1365817
29. J. Luo, P.T. Eugster, J.-P. Hubaux, Pilot: probabilistic lightweight group communication system for ad hoc networks. IEEE Trans. Mob. Comput. **3**(2), 164–179 (2004) doi:10.1109/TMC.2004.12
30. O. Morales-Ponce, E. M. Schiller, and P. Falcone (eds.), Cooperation with Disagreement Correction in the Presence of Communication Failures, in (IEEE, 2014), *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pp. 1105–1110
31. O. Morales Ponce, E.M. Schiller, P. Falcone, Cooperation with Disagreement Correction in the Presence of Communication Failures. In: CoRR abs/1408.7035 (2014). arXiv:1408.7035
32. B. Kulcsar, O. Morales-Ponce, M. Papatriantafilou, E.M. Schiller and P. Tsigas (ed.), Cooperative Driving for Best Road Network Capacity, *Nationella Konferens i Transportforskning* (2013)
33. R. Kianfar, P. Falcone, J. Fredriksson, Safety verification of automated driving systems. IEEE Intell. Transp. Syst. Mag. **5**(4), 73–86 (2013). doi:10.1109/MITS.2013.2278405
34. J.H. Lala, R.E. Harper, S. Alger, A design approach for utrareliable real-time systems. IEEE Comput. **24**(5), 12–22 (1991). doi:10.1109/2.76283
35. A. Fekete et al., The impossibility of implementing reliable communication in the face of crashes. J. ACM **40**(5), 1087–1107 (1993)
36. M.J. Fischer, N.A. Lynch, M. Merritt, Easy impossibility proofs for distributed consensus problems. Distrib. Comput. **1**(1), 26–39 (1986)
37. N.A. Lynch, *Distributed Algorithms* (Morgan Kaufmann Publishers, 1996). ISBN:1-55860-348-4
38. J.-F. Hermant, G. Le Lann, Fast asynchronous uniform consensus in real-time distributed systems. IEEE Trans. Comput. **51**(8), 931–944 (2002)
39. M. K. Aguilera, G. L. Lann, and S. Toueg (eds.), *On the Impact of Fast Failure Detectors on Real-Time Fault-Tolerant Systems*, in *Distributed Computing, 16th International Conference, DISC 2002, Toulouse, France, October 28-30, 2002 Proceedings*, Lecture Notes in Computer Science, vol. 2508 (Springer, 2002), pp. 354–370. ISBN: 3-540-00073-9
40. S.P. Boyd et al., Randomized gossip algorithms. IEEE Trans. Inf. Theory. **52**(6), 2508–2530 (2006)
41. C. Georgiou, S. Gilbert, D.R. Kowalski, Meeting the bdeadline: on the complexity of fault-tolerant continuous gossip. Distrib. Comput. **24**(5), 223–244 (2011)
42. A. Casimiro et al., A Kernel-Based Architecture for Safe Cooperative Vehicular Functions, *Industrial Embedded Systems (SIES), 2014 9th IEEE International Symposium on*, June 2014, pp. 228–237. doi:10.1109/SIES.2014.6871208

43. A. Casimiro, O. Morales Ponce, T. Petig, and E.M. Schiller (ed.), Vehicular Coordination via a Safety Kernel in the Gulliver Test-Bed, in *Distributed Computing Systems Workshops (ICDCSW), 2014 IEEE 34th International Conference on*, June 2014, pp. 167–176. doi:10.1109/ICDCSW.2014.25

44. C. Berger, O. M. Ponce, T. Petig, and E. M. Schiller (eds.), *Driving with Confidence: Local Dynamic Maps that Provide LoS for the Gulliver Test-Bed*, in *3rd Workshop on Architecting Safety in Collaborative Mobile Systems (ASCoMS), Florence, Italy, September 8-9, 2014. Proceedings*, Lecture Notes in Computer Science, vol. 8696 (Springer, 2014), pp. 36–45. ISBN: 978-3-319-10556-7. doi:10.1007/978-3-319-10557-4_6

45. J. Ibanez-Guzman, S. Lefevre, A. Mokkadem, and S. Rodhaim (ed.), Vehicle to Vehicle Communications Applied to Road Intersection Safety, Field Results, in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on* Sep 2010, pp. 192–197. doi:10.1109/ITSC.2010.5625246

46. A. Casimiro et al. (eds.), *KARYON: Towards Safety Kernels for Cooperative Vehicular Systems*, in *Stabilization, Safety, and Security of Distributed Systems - 14th International Symposium, SSS 2012, Toronto, Canada, October 1-4, 2012. Proceedings*, Lecture Notes in Computer Science, vol. 7596 (Springer, 2012), pp. 232–235. ISBN: 978-3-642-33535-8. doi:10.1007/978-3-642-33536-5_22

47. P.N.D. Costa, J. Craveiro, A. Casimiro, and J. Rufino (eds.), *Safety Kernel for Cooperative Sensor-Based Systems*, in *SAFECOMP 2013 - Workshop ASCoMS (Architecting Safety in Collaborative Mobile Systems) of the 32nd International Conference onComputer Safety, Reliability and Security, Toulouse, France, 2013*, (HAL, 2013). http://hal.archives-ouvertes.fr/SAFECOMP2013-ASCOMS/hal-00847903

48. E. Vial and A. Casimiro (eds.), *Evaluation of Safety Rules in a Safety Kernel-Based Architecture*, in *SAFECOMP 2014 - Workshop ASCoMS (Architecting Safety inCollaborative Mobile Systems) of the 33rd International Conference on Computer Safety, Reliability and Security, Florence, Italy, September 8-9, 2014. Proceedings*, Lecture Notes in Computer Science, vol. 8696 (Springer, 2014), pp. 27–35. ISBN: 978-3-319-10556-7. doi:10.1007/978-3-319-10557-4_5

49. M. Pahlavan, M. Papatriantafilou, and E. M. Schiller (ed.), Gulliver: A Test-Bed for Developing, Demonstrating and Prototyping Vehicular Systems, in *Proceedings of the 75th IEEE Vehicular Technology Conference, VTC Spring 2012, Yokohama, Japan, May 6-9, 2012*, (IEEE, 2012), pp. 1–2. ISBN: 978-1-4673-0989-9. doi:10.1109/VETECS.2012.6239951

50. C. Berger et al., Bridging physical and digital traffic system simulations with the gulliver test-bed, in *Proceedings of 5th International Workshop on Communication Technologies for Vehicles, Nets4Cars/Nets4Trains 2013*, Villeneuve d'Ascq, France, 14–15 May 2013, ed. by Marion Berbineau et al., vol. 7865. Lecture Notes in Computer Science (Springer, 2013), pp. 169–184. ISBN: 978-3-642-37973-4. doi:10.1007/978-3-642-37974-1_14

Chapter 8
Fault Tolerant Architecture for Infrastructure based Vehicular Networks

João Almeida, Joaquim Ferreira and Arnaldo S.R. Oliveira

Abstract Wireless vehicular communications have been a trending topic in the last few years, leading to the development of a complete set of new standards and the emergence of innovative vehicular applications. Despite the obvious benefits of vehicular networks, it has been a challenging issue to design dependable vehicular communication systems. This is mainly due to the high speed mobility scenarios that are involved and the open nature of these networks. As a consequence, there are scalability problems with the proposed medium access control (MAC) methods under highly dense traffic environments. This results in large values for the end-to-end delay and for the probability of packet drops, compromising the reliability of vehicular communications. Besides that, there are few strategies to enhance fault-tolerance in vehicular systems, whose operation strongly depends on the dynamic topology of the network and on the real-time guarantees provided by the communications protocol. Based on these arguments, this chapter presents a fault-tolerant architecture to improve the dependability of infrastructure-based vehicular networks. The presence of road-side units (RSUs) and a backhauling network adds a degree of determinism that is useful to enforce real-time and dependability, both by providing global knowledge and supporting the operation of collision-free deterministic MAC protocols. One of such protocols is V-FTT, for which the proposed architecture was designed as a case study. Notice, however that this architecture is protocol independent and can be adapted to any wireless communications system. The chapter's final sections specially focus on the design of fail silent RSUs, by presenting the proposed implementation and the obtained experimental results.

J. Almeida (✉) · A.S.R. Oliveira
Instituto de Telecomunicações, DETI - Universidade de Aveiro, Aveiro, Portugal
e-mail: jmpa@ua.pt

A.S.R. Oliveira
e-mail: arnaldo.oliveira@ua.pt

J. Ferreira
Instituto de Telecomunicações, ESTGA - Universidade de Aveiro, Aveiro, Portugal
e-mail: jjcf@ua.pt

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_8

8.1 Introduction

Wireless vehicular networks, as a fundamental area of research in modern Intelligent Transportation Systems (ITS), aim to improve vehicle and road safety, passenger's comfort, efficiency of traffic management and road monitoring. Vehicular communications rely on the recent IEEE 802.11-2012, IEEE 1609 and ETSI ITS-G5 family of standards, in which there are still some open problems concerning the timeliness and dependability of the exchanged messages [12]. In order to ensure correct operation even under dense traffic conditions, vehicular communication nodes and protocols should be developed by taking into consideration the questions that commonly arise in the design of dependable real-time systems [1], namely deterministic operation, timely behaviour, safety, reliability, availability, among others. The need for these design concerns is even more evident, given the scenario described next.

During a transitory market penetration period, vehicular communication systems will be regarded as an auxiliary technology that could aid driver to take more informed decisions and warn him about dangerous situations. However, after this initial phase, and with the advent of autonomous vehicles, road traffic systems will completely rely on the information provided by this and other technologies (such as cameras or radars). At that stage of development, the human judgement, which is very prone to error, will likely be replaced by computer-driven decisions. In this scenario, dependability will be an essential aspect in road traffic safety systems, since their operation will strongly depend on the correct service provided by vehicular communication devices. Based on these arguments, vehicular networks must be analyzed as distributed computer control systems (DCCS) that interact together to achieve the common goal of guaranteeing traffic safety.

In the last few decades, DCCS have been widely used in many application fields, such as robotics, industrial process control, avionics and automotive systems. A large number of these applications pose strict requirements, which if not fulfilled may cause important economic and environmental losses or even put human life in danger [14]. These systems must exhibit a high probability to provide continuous correct service. Beyond that, many of them comprise real-time activities that must be performed within stringent time bounds. Therefore, in safety-critical DCCS with real-time constraints such dependability attributes are of uttermost importance, since its distributed nature requires a timely and reliable exchange of data among the several nodes, in order to achieve the envisaged control over the operating environment.

Due to the nondeterministic behaviour of the environment where each specific distributed computer control system operates, the use of the best design's practices does not fully guarantee the absence of faults. Thus, fault-tolerance methods need to be included in the system to avoid that possible faults can cause its failure. By combining these kind of mechanisms with system's real-time requirements, dependable DCCS can be developed for operating in safety-critical scenarios, such as the ones existing in vehicular environments.

Several types of faults must be considered in vehicular scenarios. For example, the wireless channel is regularly affected by transient faults in the communication link due to the constantly changing atmospheric and road traffic conditions. This effect is much larger than the one observed for instance in wired or indoor wireless environments. Furthermore, channel permanent faults could also occur, due to unregulated interference, which can typically be considered as malicious faults, since the spectrum band assigned for wireless vehicular communications is reserved by law. Not only the wireless channel, which is a single point of failure in vehicular systems, but also the nodes of the network should be regarded as a possible source of problems. For instance, hardware and software faults can affect the operation of either the road-side units (RSUs) that constitute the network infrastructure or the on-board units (OBUs) placed inside each vehicle. For the given reasons, a careful design work must be performed in the deployment of vehicular communications systems, by taking into consideration the general dependability aspects of safety-critical applications, as well as the specific issues that arise in the vehicular context.

8.2 MAC Issues in Vehicular Networks

The medium access control (MAC) layer defined in IEEE 802.11 adopts a carrier sense multiple access with collision avoidance, in which collisions may occur indefinitely, due to the non-determinism of the back-off mechanism. Therefore, native IEEE 802.11 alone does not support real-time communications. However, this property is crucial for safety applications, where warning messages have to be always delivered with a bounded delay.

The probability of collisions occurring may be reduced if the load of the network is kept low, which can be accomplished by using an adaptive and distributed message-rate control algorithm, as described in [2]. Although this type of solution can reduce the probability of collisions, it does not provide strict real-time guarantees. Collision-free MAC protocols are considered deterministic as data collisions do not occur and a worst-case delay from packet generation to channel access can be calculated. This can only be achieved if the protocol restricts and controls the medium access to provide a deterministic behaviour.

In the design of a deterministic MAC protocol for vehicular communications, there are basically two main possible choices. The protocol could rely on the road side infrastructure [5, 18, 20] or it could be based on ad-hoc networks [11, 17, 23]. A hybrid approach that takes advantage of both models could also be implemented. Strict real-time behaviour and safety guarantees are typically difficult to attain in ad-hoc networks, but they are even harder to achieve in high speed mobility scenarios, where the response time of distributed consensus algorithms, e.g. for cluster formation and leader election, may not be compatible with the dynamics of the system. Therefore, the presence of the infrastructure, e.g. road-side units (RSUs) and the backbone cabled network, adds a degree of determinism that is useful to enforce real-time and dependability at the wireless end of the network.

In addition to this, if the road-side infrastructure can be made more predictable than the other parts of the network, by executing certain tasks faster or in a more reliable way, the system could be seen as an instantiation of the *wormhole* metaphor [29].

In this scheme, the uncertainty is neither uniform nor permanent across all system components, and the road-side units which are connected through the backhauling network, can be seen as *wormholes*, since they are more reliable and predictable than the mobile nodes of the network (the on-board units—OBUs).

8.2.1 Overview of V-FTT Protocol

Based on the above arguments, a deterministic medium access control (MAC) protocol was proposed [13, 26], taking advantage of the road-side infrastructure. This protocol, entitled Vehicular Flexible Time-Triggered (V-FTT), adopts a multi-master multi-slave spatial time division multiple access (STDMA). The road-side units (masters) are responsible for registering the on-board units (slaves) with the infrastructure and for scheduling their transmission slots. These masters are synchronized through GPS receivers placed in each one of the nodes. They also share common knowledge of the road traffic system, which is attained by exchanging update messages holding the current state of each individual RSU's database. These messages are transmitted through the backhauling network (e.g. fiber optics), enabling RSUs with a global vision of the entire vehicular network.

The protocol is divided into periodic elementary cycles (ECs) with 100 ms duration and, as can be seen in Fig. 8.1, each EC is further divided into three different parts. It starts with an Infrastructure Window, that is used by the RSUs to transmit two types

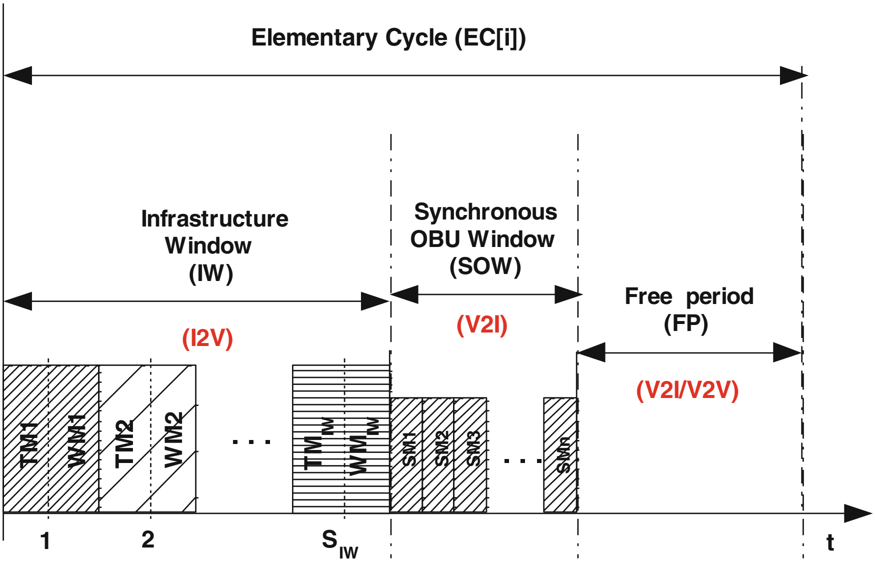


Fig. 8.1 Elementary cycle of Vehicular Flexible Time-Triggered protocol [26]

of messages: trigger messages (TM) with the schedule of the registered OBUs; and warning messages (WM) with cross-validated information regarding safety events that occurred in the road. In vehicular networks, these messages sent by the RSUs to the OBUs are commonly referred as Infrastructure-to-Vehicle (I2V) communication (Fig. 8.1). The second part corresponds to the Synchronous OBU Window, where each OBU has a fixed size slot to send information to RSUs (Vehicle-to-Infrastructure or V2I communication), either regular data (like vehicle's speed and heading) or a safety event. The Synchronous OBU Window duration is variable, depending on the number of OBUs (denoted as n in Fig. 8.1) scheduled in that particular elementary cycle. At the end of the EC, there is a free period, during which non V-FTT enabled OBUs can communicate (Vehicle-to-Vehicle or V2V communications), and the V-FTT nodes (both RSUs and OBUs) can exchange non-safety messages.

In order to increase the probability of successful reception of a TM or a WM by an OBU, RSUs' coverage areas can partially overlap, causing the same OBU to receive TMs/WMs from different RSUs. This redundancy level (S variable in Fig. 8.1) can be dynamically configured, depending on the number of RSUs in the same region and on the transmission power and sensitivity levels of the nodes of the network. However, some coordination is required among RSUs in order to ensure that for a given OBU, the transmission slot allocated to it in the Synchronous OBU Window (SOW) is the same in all TMs transmitted by the distinct RSUs. This is based on the fact that each OBU will only transmit a single message per EC. In a similar way, the OBUs' messages can be listened by several different RSUs during the SOW period.

From the above description, one can conclude that in the proposed protocol, RSUs play an extremely important role, since they are responsible for all traffic scheduling and admission control mechanisms. If an RSU stops working properly, all communications in its coverage area can be severely compromised. There are no guarantees anymore regarding the timeliness and deterministic properties of the protocol. This leads to the development of fault-tolerance mechanisms that are able to improve the dependability of V-FTT and other deterministic protocols with the same characteristics.

8.3 Fault-Tolerance Techniques

As it was referred in Sect. 8.1, the design of dependable systems is becoming pervasive in many domains, e.g. in automotive vehicles, in avionics, in nuclear plants, in factory automation, etc. These systems are usually distributed and rely on a communications network to interconnect sensors, actuators and controllers in a reliable and timely way. Although dependability aspects are traditionally studied in fieldbus technologies and wired networks, the same fundamental concepts apply to wireless systems. Since vehicular communications are expected to provide safety-critical road services with a high level of reliability, this work aims to develop methods to enhance dependability in vehicular networks. A real-time communication protocol, such as the one previously discussed, can attain an higher probability of deliver correct safety services if dependability attributes are considered and mechanisms to improve them are implemented. For that purpose, fault-tolerant techniques need to be included in vehicular communication systems, in order to cope with the presence of unpredicted operating problems.

In the specific case of real-time communications, the mechanisms to achieve fault-tolerance are not just concentrated in the network nodes through, for instance, voting schemes or hardware redundancy. The faults propagated in the communication channel may also compromise the operation of the distributed system, since erroneous information can be shared and the ability to achieve common knowledge could be destroyed. A faulty message transmitted by a malfunctioning node may be propagated to all other nodes causing the whole system to collapse. Notable examples of these latter faults are timing or value failures in a node or replica non-determinism. Therefore, methods to ensure the validity of the transmitted messages should be employed in the design of a real-time communications system.

The rest of this section presents a survey of some relevant topics of dependability in the context of real-time communications, such as replication and fail silence failure mode. These techniques will later (Sect. 8.4) be employed in the design and definition of a fault-tolerant architecture for infrastructure based vehicular networks.

8.3.1 Replication

One of the most common methods to build fault-tolerant distributed systems is to replicate subsystems that fail in an independent way. The goal of this approach is to give other subsystems the idea that the delivered service is provided by a single entity. In order to enforce consistency among the replicas, there are two main categories of replication schemes: active and passive replication.

Active replication, also called state machine approach [24], is characterized by having all the replicas receiving and processing the same sequence of requests in parallel, and producing the same output result at the end. In order to ensure that the state of these replicas is kept consistent, i.e., they will produce the same output, it has to be guaranteed that all of them are fed with the same inputs in the same order, typically by employing an atomic broadcast protocol to disseminate the commands [32]. These requests are handled independently but must be processed in a deterministic way.

In active replication, there is no need for an explicit recovery procedure when one of the replicas fails, since the other ones will continue to provide correct service. Therefore, this technique is simple and transparent to the clients in case of node failure. However there are some disadvantages with this approach. For example, the determinism constraint may be difficult to enforce (e.g. in a multithreaded node) and it is resource demanding, because the operation of each replica requires a full set of resources.

Passive replication, also known as primary backup [7], is more economic than the active scheme, because only one replica, the primary one, processes the commands and produces the output results. The remaining replicas, called secondary or backup, remain in idle state and only interact with the primary to update and log all commands. When the primary system fails, the output result can not be delivered immediately with this technique. Instead, one of the backup replicas is selected as the new primary and it will resume operation from the last well known state of the system with the aid of the information available in the log repository. In this case, the client will time-out and resend the request after detecting the new primary replica. This significantly increases the response time of the system in case of failure, making this method unsuitable for some time sensitive applications. In passive replication, no determinism constraint is necessary but special care must be put on the mechanisms that enforce agreement between primary and backups.

Semi-active and semi-passive replication are two variants of the replication schemes referred above. The idea behind semi-active replication [21], also called leader-follower, is that the replicas do not need to process requests in a deterministic way. Only one replica (the leader) is responsible for making non-deterministic decisions and inform the followers of these choices. In semi-passive replication [9], client requests are received by all replicas and every replica delivers its responses back to the client. This way, the clients do not need to know the identity of the primary and there is no need for time-outs to detect the crash of the primary, being system failures completely masked to the client. Semi-passive replication is fully based on failure detectors and thus it does not require an agreement method (e.g. membership service) to select the primary replica. This reduces the response time in case of crash of the primary, when compared with the passive replication approach.

The earlier replication strategies proposed in the literature were mostly focused on applications for which real-time behaviour was not an essential requisite. However, real-time applications usually operate under stringent timing and dependability constraints and therefore, several replication strategies had been developed [15, 19, 33], in order to solve the additional challenges posed by safety-critical environments.

An important concept related with the design of replication protocols is the fail silence failure mode that will be described next.

8.3.2 Fail Silence Failure Mode

A faulty node of a distributed system that sends unsolicited messages at arbitrary points in time (babbling idiot failure mode [14]) without respecting the media access rules can disable nodes with legitimate messages to access the network. However, this failure mode can only occur if a node fails in an uncontrolled way. Network topologies that support the operation of fail uncontrolled nodes are costly [21]. Thus a node should only exhibit simple failure modes and ideally it should have just a single failure mode, the fail silent failure mode [27], i.e., it produces correct results or no results at all. In this matter, a node can be fail-silent in the time domain, i.e., transmissions occur at the right instants, only, or in the value domain, i.e., messages contain correct values, only. With fail silence behaviour, an error inside a node cannot affect other nodes and thus each node becomes a different fault confinement region [27] (a specific part of the system where the immediate impact of a fault is limited only to that defined region). Furthermore, if k failures of a functional unit in a system must be tolerated, then $k + 1$ replicas of that unit are needed as long as they are fail silent. If the replicas are fail uncontrolled, then $3k + 1$ will be required [16]. Thus, the use of fail silent nodes also reduces the complexity of designing fault-tolerant systems.

The alternatives to enforce fail silent behaviour may be generically divided in two main groups. The ones that result from adding redundancy to each node and ones that rely on behavioural error detection techniques [27].

Using replicated processing within a node with output comparison or voting calls for the use of mechanisms to keep the replicas perfectly synchronized and to avoid replicas to diverge due, e.g. to asynchronous events. Synchronization at processor instruction level is the most obvious way to achieve replica synchronism, driving identical processors with the same clock source and evaluating their outputs (either comparing or voting) at critical instants, e.g. every bus access. Special care must be taken with asynchronous events that must be delivered to the processors so that all perceive the same event at the same point of their instruction streams.

Over the years, many systems were designed based in double-processor fail silent nodes such as Sequoia [4] and Stratus [30]. However, these systems have some drawbacks [6]. First of all, the processors must exhibit the same deterministic behaviour every clock cycle and don't care states are not allowed so that they produce identical outputs. Secondly, the use of special purpose hardware as comparators or voters, reliable clock sources and asynchronous event handlers greatly increases the design complexity. Finally, due to their operation in lock step, a transient fault could affect both processors in the same way, making the node susceptible to common mode failures. An alternative approach to eliminate the hardware level complexity of the solutions referred above is to transfer the replica synchronism to a higher level using software protocols over a set of standard processors operating independently of each other in a node. Task synchronization approaches were used in SIFT [31] and Voltan [25].

Behavioural error detection mechanisms, either in software or in hardware, are another alternative for enforcing fail-silence behaviour. Mechanisms such as checksums, watchdog timers and processor monitoring, are usually implemented using commercial-off-the-shelf (COTS) components. Error detection latency is the major bottleneck of these systems since the error detection mechanisms are only able to detect errors a relatively long time after they occur, possibly forcing other nodes to put in place some sort of error recovery policy.

8.3.2.1 Bus Guardians

Bus guardians (Fig. 8.2), which are autonomous devices with respect to the node network controller and host processor, also implement behavioural error detection mechanisms. Usually used in wired networks to enforce fail silence, bus guardians

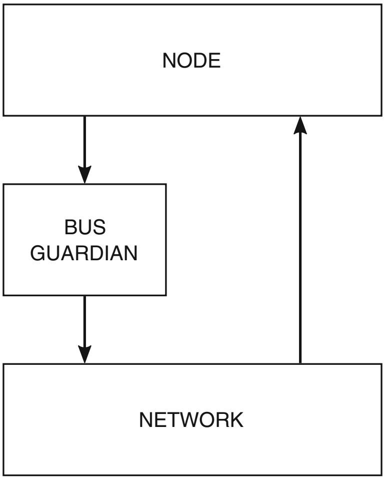


Fig. 8.2 Generic bus guardian scheme

act as failure mode converters, i.e., the failure modes of the component are, at the interface to other components, replaced by the failure modes of the guardian.

In order to be fail-independent with respect to the interface it monitors, the bus guardian must belong to a separate fault confinement region. A guardian would be of no use if it failed whenever the node that it is guarding also failed. Some potential sources of common mode failures are: clocks, CPU/hardware, power supply, protocol implementation, operating system, etc. Designing a bus guardian with independent hardware, with no common components and design diversity can help to avoid common failure modes. Despite the possible design compromises made between independence, fault coverage and simplicity/cost in any bus guardian architecture it is mandatory for the guardian to have some a priori knowledge of the timing behaviour of the node it is policing. In time-triggered (TDMA) networks this implies that each bus guardian needs to have its own copy of the schedule and an independent knowledge of the time.

Almost all the work developed in the area of fail silent systems is for wired industrial, or automotive systems [3, 28]. Although the principles are similar, nodes of wireless networks pose some additional problems implementing fail silent behaviour. For instance, the open nature of such networks, in contrast with the closed wired environments found in industrial and automotive systems, raises several issues to the identification of which and how many nodes are in the radio range of a given network at each moment in time. For the same reason, a number of other problems, like the hidden node, arise and need to be addressed. Besides, wireless communications systems are inherently half-duplex, i.e. a node is not able to transmit and receive at the same time, an important characteristic that must also be considered. Furthermore, the use of centralized bus guardians as in star topologies is not possible. However, in wireless communications, one can think in medium guardians as devices that protect non-faulty wireless network nodes from erroneous ones.

8.4 Achieving Fault-Tolerance in V-FTT

In master-slave networks, as V-FTT, the most obvious issue that must be dealt with, when addressing fault-tolerance and dependability, is the single point of failure formed by the master holding the traffic scheduler and vehicles registration database. In fact, if a master node (RSU) fails to transmit trigger messages with the EC-schedules, transmit them out of time or with erroneous contents, then all network activity could be seriously compromised or even disrupted.

This can be handled using replication, with one or more similar nodes acting as backup masters. In this way, as soon as a missing trigger message is detected, a backup master comes into the foreground and transmits it, maintaining the communication without any discontinuity of the traffic schedule. However, this is only possible if all master replicas are synchronized, with respect to value and time.

To ease the task of designing mechanisms that enforce masters replication, it is considered that nodes are fail-silent, i.e., nodes can only fail by not issuing any message to the network. This, however, must also be enforced by using adequate components as nodes can fail uncontrollably. These additional elements should guarantee that the messages sent to the medium by the RSUs are correct both in time and value domain.

After guaranteeing that all active RSUs exhibit fail-silence behaviour, an active replication scheme can be used to ensure that even in the presence of a failure in the primary RSU, the trigger messages with the EC-schedules will still be delivered to the OBU nodes. In this active scheme, the backup RSUs receive and process the exact same sequence of messages in parallel with the active one, and produce the same trigger messages in each EC. However, the packet transmission operation in the backup RSUs is deliberately delayed by a small amount of time, in the order of few μs, relatively to the primary one. This is done with the objective of facilitating the recovery procedure in case of RSU failure, by making it completely automatic and transparent to the slave nodes. This way, if the active RSU is able to transmit the trigger message in the planned instant, the replica will sense the wireless medium as occupied and will conclude that the primary system is free of error. Hence, it will not issue any message to the air, avoiding any overlap with the active node. On the other hand, if at that moment the medium is perceived as free, the replica will continue the transmission of its message and will replace the operation of the previously active RSU. As a result, the trigger messages will still be transmitted on time, since only a small delay is introduced at the beginning of the EC.

OBUs (slaves) should also exhibit fail-silence behaviour and although one could adopt the same mechanism used in master nodes, that would be expensive. The cost of that solution probably could not be supported by the vehicles' owners. Thus, slave nodes fail-silence enforcement both in time and value domain should only be adopted in special cases where the slave node information (value and timing) is absolutely essential, e.g. for police and emergency vehicles. For regular vehicles, a more inexpensive solution must be considered. Given the fact that slave nodes are not responsible for any type of network coordination, limiting OBUs ability to transmit uncontrollably will suffice. This corresponds to enforce fail-silence behaviour in the time domain only. From the OBU's perspective, a schedule is valid only within the scope of an elementary cycle, thus an entity policing the node only needs to be aware of the node schedule in a EC by EC basis. This entity can be regarded as a medium guardian that avoids node's transmissions outside the time slot assigned to it by the masters of the network—the RSUs. For that, this medium guardian just needs to decode every trigger message contents and block any unscheduled transmission from the node.

8.4.1 Fault Hypothesis

Figure 8.3 presents the overall architecture of the network based on road side infrastructure. The fault-tolerance mechanisms are also depicted, giving rise to the following fault hypothesis:

- **Node faults**—Master nodes (RSUs) are assumed to exhibit fail-silence failure semantics, which is guaranteed by their internal redundancy and a fail silence enforcement entity that validates the agreement both in value and time domains. This mechanism will be explained in more detail in the following sections. Besides that, RSUs are also replicated (Backup RSUs), in order to undertake a failure in the active nodes (Active RSUs). In OBUs (slave nodes), medium guardians are used, to enforce fail silence only in time domain.
  In this scheme, the fail silence enforcement entity (as well as the medium guardian when considering OBUs) belongs to a fault containment region that is assumed independent of the one constituted by the remaining components of the node.

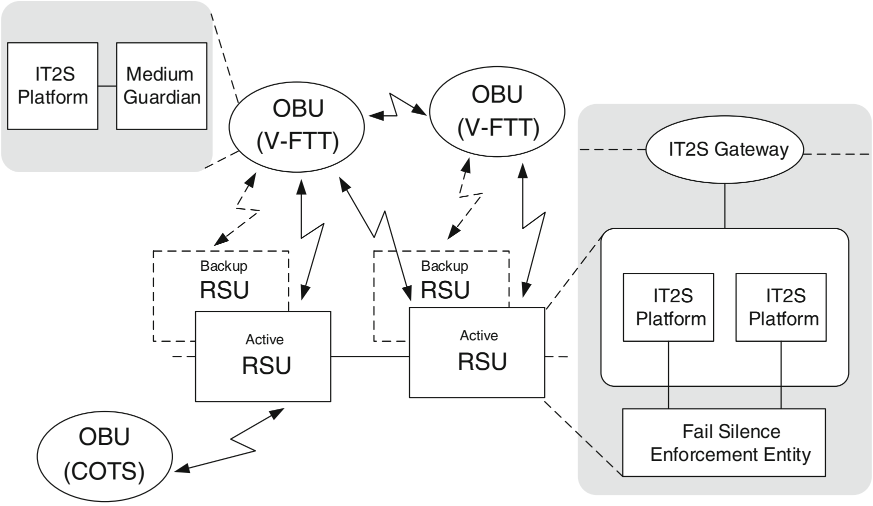


Fig. 8.3 Fault-tolerance mechanisms in V-FTT network

The covered faults within each node include hardware faults, both transient and permanent, and software faults. However, as it will be shown, faults in the analog part of the physical layer are not considered. Byzantine faults, notably intrusions, are not totally covered, since such kind of faults need to be handled also at the IT2S Gateway level, a component that provides communication with other RSUs through the backhauling network.

- **Channel transient faults**—Vehicular communications are regularly affected by transient faults, since the wireless channel conditions may vary, depending on atmospheric and traffic conditions. This effect is much larger than the one observed in communication protocols for wired environments. A way to circumvent the higher packet error rate is by introducing time and spatial redundancy in the Trigger Message transmission by the RSUs, as depicted in Fig. 8.1. This can be achieved by deploying a more dense RSU distribution along the road, leading to a partial overlap of the RSUs' radio coverage areas. This way, a single OBU transmission can be scheduled by a configurable number of adjacent RSUs, typically 3, for the same transmission slot. In this case, time and space diversity are employed together, since several RSUs placed in distinct locations transmit the same EC-schedule at different time instants. The required RSUs coordination for the execution of this redundancy mechanism is guaranteed through the backhauling network. It is thus assumed that every OBU receives at least one Trigger Message every elementary cycle, i.e., channel transient faults do not impact the trigger message transmission. In this RSU redundant scheme, space diversity is also attained for the transmission of OBU messages, since several RSUs can receive the same packet. Moreover, critical nodes could be assigned with several time slots to re-transmit their messages.
- **Channel permanent faults**—The transmission medium is a single point of failure for vehicular networks. Permanent faults may occur in the wireless medium, e.g. due to unregulated interference, however, such faults are not considered in this work. Depending on the severity of channel interference, medium redundancy could be included, by detecting the disturbance and commuting the operation of V-FTT protocol to another channel in the available frequency spectrum. As the band assigned for wireless vehicular communications is reserved by law, unregulated interference can be considered as malicious faults.
- **Synchrony assumptions**—Nodes synchronization, both masters and slaves, is ensured by a GPS receiver located at each one of the nodes. The accuracy provided by this system is typically below 333 μs (the maximum value used to determine if a device is synchronized to UTC or not [10]) and it must be sufficient to cope with the synchronization requirements of V-FTT. However, if by some reason, the GPS signal is not available or is not sufficiently accurate at some instant in time, another strategy could be used. The alternative relies on the fact that the trigger message transmitted by the master node, besides conveying the scheduling information, can also act as a synchronization mark to all network nodes. This way, master nodes are also time masters, and it is assumed that in between two consecutive trigger messages (typically 100 ms), the clock counters of each node do not diverge more than a negligible amount of time.

Based on this network architecture and associated fault hypothesis, the next sections discuss the design and implementation of one of the proposed mechanisms to increase dependability in infrastructure based vehicular networks: the fail silence behaviour of RSUs. A rationale with several possible design choices for the fail-silent RSU is presented, together with its implementation in dedicated hardware and some obtained experimental results.

8.5 Enforcing Fail-Silence in V-FTT

8.5.1 IT2S Platform: A Brief Description

The proposed system architecture takes advantage of a flexible implementation of an IEEE-WAVE/ETSI-ITS-G5 controller, the IT2S platform. This platform (Fig. 8.4) has been developed from scratch at Telecommunications Institute (Aveiro site) and it could either operate as an RSU or as an OBU. The IT2S platform is essentially constituted by three main modules: the Smartphone, the Single Board Computer and the IT2S board. The latter one implements the recent IEEE 802.11p standard, focused on the MAC and physical layers of the protocol stack. However, since the implementation of the MAC layer resides in a hardware/software partition co-design, only the low level functionalities are executed in the IT2S board by the FPGA. This sub-layer that comprises the time-critical and deterministic operations of the MAC scheme, is designated as Lower MAC (LMAC).

The physical layer is completely implemented in the IT2S board, being divided in two main parts: the Analog PHY and the Digital PHY. The Analog PHY is responsible for the signal processing operations in the analog domain, such as the up and down conversion from base-band to RF and vice-versa, respectively. On the other hand,

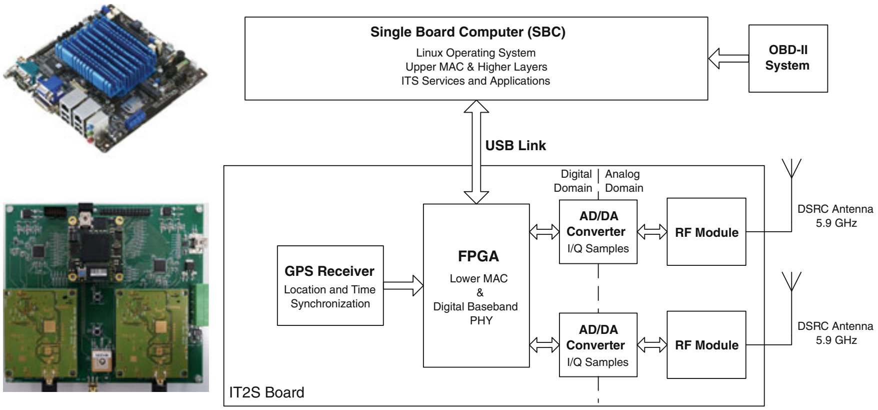


Fig. 8.4 Main blocks of the IT2S platform

the Digital PHY deals with signals in the digital domain, implementing the OFDM transmission and reception chains, converting bytes from a MAC frame into baseband In-phase and Quadrature (I/Q) samples and the reverse operation.

In order to cope with the simultaneous multi-channel operation requirement specified in the most recent versions of the standards [8], the board includes two complete sets of hardware units (2 DSRC antennas and RF modules, 2 AD/DA processors and 2 digital PHY and LMAC modules inside the FGPA) for the implementation of the IEEE 802.11p standard in both radios. The IT2S board also incorporates a GPS receiver for location and synchronization purposes. The interconnection with the SBC is established through an USB link and it is based on a time multiplexing scheme, allowing the co-existence of several independent channels for accessing each radio unit separately, retrieving information from the GPS device, performing updates on the FPGA bitstream, etc.

The main function of the Single Board Computer (SBC) is to execute the higher layers of the WAVE/ITS-G5 protocol stack, namely from the high level functionalities of the MAC layer, called Upper MAC (UMAC), to the Application layer. The SBC is a COTS embedded PC that runs a Linux-based operating system, providing high degree of flexibility and more control over the system's operation. When operating as an OBU, the SBC can also interact with the (On-Board Diagnostics) OBD-II system available in all recent vehicles. This way, it can access detailed information about vehicle's status and performance.

The Smartphone is responsible for implementing the graphical user interface that will provide more traffic related information to vehicle's driver and passengers. For instance, it will be able to display warnings in case of road accidents or traffic congestions. Another important advantage of the Smartphone is its capability to provide connectivity between the IT2S platform and a 3G or a 4G network. This feature allows the remote diagnostics and access to the information available in the platform, as well as a possible upgrade of the software and the reconfiguration of the bitstream in the FPGA. Finally, it could also enable the implementation of the eCall service, which basically consists in an automatic call to the 112 emergency number in the event of a serious road accident. As already referred, the IT2S platform can operate either as an RSU or an OBU. Obviously when the platform is working as an RSU, there is no need for a graphical user interface, and therefore the Smartphone is not included in the overall system's architecture.

8.5.2 Fail-Silent RSU Design

In order to implement a fail-silent RSU, all the possible device failure modes must be converted in fail-silent failure mode, enforced by a simpler, thus less prone to failures, component. The complexity of this fail silence enforcement entity can be greatly reduced if it is implemented at the lower layers of the OSI stack, in which the number of possible defects of system's design decrease and are easily identified [14, 22]. The design of the fail silence mechanism at the lower layers of the protocol stack is simplified given the white box access to a flexible vehicular research platform. Implementing a similar solution in a COTS platform would probably imply higher latency, caused by the API.

As briefly discussed in Sect. 8.4, the operation of the proposed fail silence mechanism consists in an internal redundancy scheme, based on the replication of the IT2S platform (right side of Fig. 8.3). Two complete sets of single board computers and IT2S Boards are used to produce messages, whose purpose is to be disseminated through the air medium. The fail silence enforcement entity then compares the values and the timing of the messages produced by these two sets and, if everything is working correctly, it validates the frame and allows its transmission by one of the platforms. On the other hand, if the frames differ or are significantly out of phase, the entity silences the system until a restart signal is received.

Notwithstanding the decision to implement the fail silence mechanism at the lower layers of the protocol stack, there are still several design choices that should be taken when choosing the ideal place and method to perform the comparison of the samples produced by both platforms. As shown in Fig. 8.5, there are three main possible checkpoints where this verification can be executed, if one considers MAC and PHY as the appropriate layers to implement this mechanism. The rationale to choose only these lower levels is, as already explained, based on the observation that moving it higher on the protocol stack, will increase the design complexity and will potentiate design defects [22]. The three different checkpoints, numbered as 1, 2 and 3, correspond to the output interfaces of MAC, digital and analog PHY layers, respectively.

Output comparison at checkpoint number 1 could be attained using a voting scheme between both MAC frames produced at the output of Lower MAC sublayer

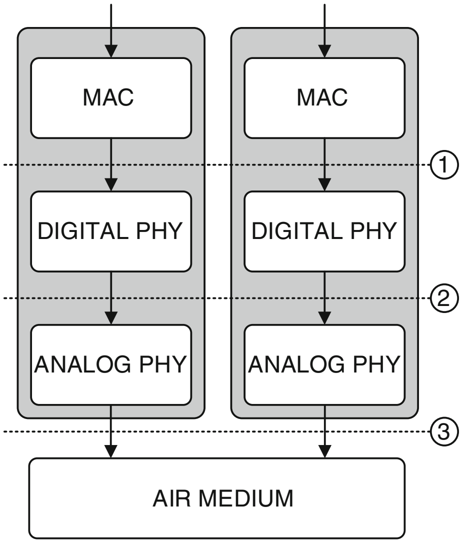


Fig. 8.5 Possible checkpoints for the fail silence mechanism

in the FPGA. The verification at this point, however, will not include possible faults in the operation of the digital physical layer, which basically comprises the baseband processing of the OFDM transmission chain. This could be attained if the fail silence mechanism is implemented at checkpoint 2, in the interface between the FPGA and the AD/DA processor, by comparing the I/Q samples that constitute the OFDM modulated frame. A verification at this stage already encompasses any discrepancy at higher levels of the IT2S platform, validating all the processing done at the SBC and at the FPGA. If, for instance, a software fault occurs at the higher layers, leading the SBCs to attempt transmitting different frames, e.g. due to distinct views of the network or inconsistent scheduling computation, the outputs at this checkpoint will differ and that fault can be detected. Nevertheless, in an ideal scenario, checkpoint number 3 would be the best place to verify the correctness of the outputs produced by both transmission chains, since it would also include the operation of the analog part of the IT2S platform. However, implementing an online verification algorithm of high frequency analog signals is a very complicated task, since signal comparison would probably require a downconversion to a lower frequency, which will add an excessive delay and will demand for resources in the analog part that may not be available. Furthermore, both transmission chains could produce correct signals although slightly different, due to minor mismatches on the digital to analog conversion and on the radio frequency amplification processes.

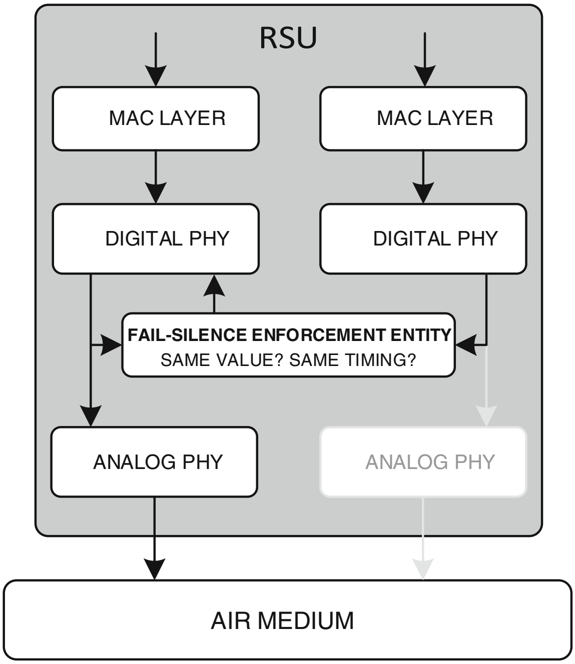


Fig. 8.6 Basic fail silent master RSU scheme

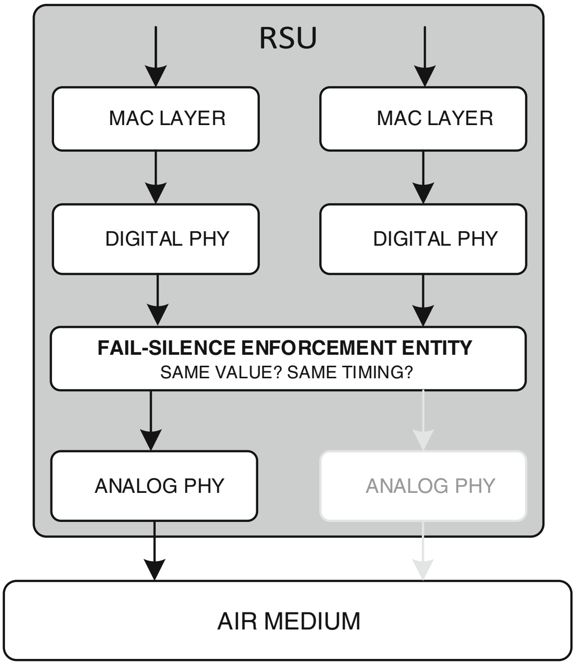


Fig. 8.7 Improved fail silent master RSU scheme

From the previous analysis, one can conclude that checkpoint 2 is the most appropriate place to implement the fail silence mechanism. A basic implementation, presented in Fig. 8.6, would be to perform a runtime comparison of the output produced by both digital PHYs and signal an error that would abort the ongoing transmission whenever a mismatch is detected. This method, however, would not prevent the medium from being occupied during at least part of an incorrect frame transmission, since a small tolerance must be allowed at the moment the results are produced. This time tolerance is needed to cope with slight variations between the internal clocks of both platforms that are synchronized through independent GPS receivers. As a consequence, the described solution would not result then in a true fail silent enforcement entity.

A possible improvement on the previous scheme is presented in Fig. 8.7, where each of the samples produced by the digital PHY is only allowed to proceed to the analog PHY after successful validation. This solution, although providing protection against frames that are completely different or out of time, still does not provide true fail silence behaviour, because a single error occurring at the middle of a message would invalidate the complete transmission. The medium could therefore be occupied with samples that although correct when analyzed independently, were invalid in the context of a full frame. Beyond that, given the fact that in the proposed real-time protocol, the RSU coordinates all the communications in the wireless channel, if an RSU transmission is interrupted, a complete elementary cycle will be wasted.

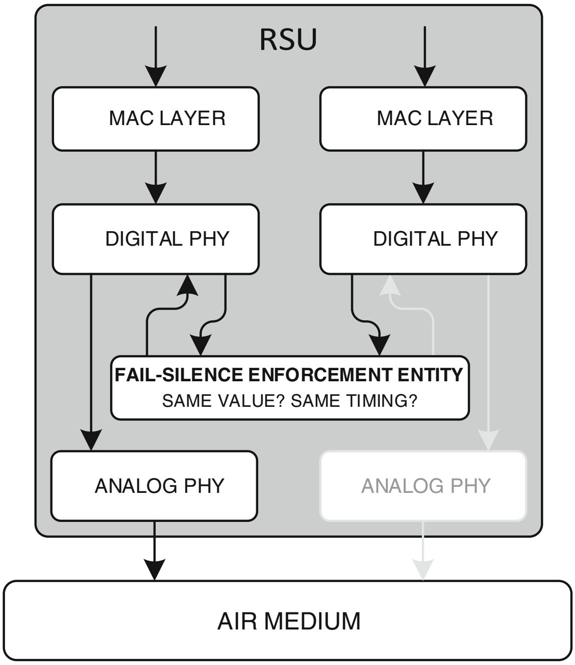


Fig. 8.8 Final fail silent master RSU scheme

A possible solution for this problem is to modify the transmission chain to produce the samples that are to be sent over the air in advance, relative to the moment when they are supposed to be sent. If enough advance is provided, the samples of an entire frame can be compared and validated before that frame transmission has even started. In this scheme, the validated samples are then fed back to the digital PHY that will, using a very simple mechanism, apply them at the correct moment to the analog PHY for transmission. If any difference is observed at any part of the frame or if the measured delay between the instants when samples are provided is greater than a certain fixed tolerance, the fail silent entity will not allow any of two units to transmit. In this case, the medium will not be occupied inadequately, contrarily to the previous proposals.

Based on these arguments, Fig. 8.8 presents the proposed scheme for the fail silent RSU and the final location of the fail silence enforcement entity inside the protocol stack. This entity was placed in the closest point possible to the antenna, at the end of the digital physical layer and before the analog part. This way, it is possible to validate the operation of the entire system and protocol, except the analog side of the physical layer. Ideally, the analog path should also be included, however and as already mentioned, implementing a runtime verification algorithm of high frequency analog signals is a very difficult task. Moreover, in order to guarantee that the fail silence enforcement entity belongs to a different fault containment region, with no common failure mode, it was developed in a external PCB with separated power supply and clock source.

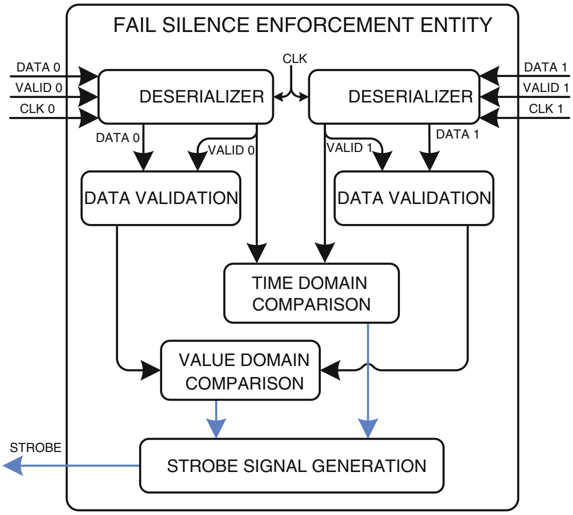


Fig. 8.9 Block diagram of the fail silence enforcement entity

In addition to this and as already referred, to truly implement a fail silent system, both transmission chains have to produce and send the frames to the fail silence enforcement entity in advance, so that the entire message could be verified before a single bit is transmitted through the antenna. More details regarding the internal architecture of the fail silence entity will be provided next.

8.5.3 *Fail Silence Enforcement Entity*

Figure 8.9 depicts the interfaces and the internal structure of the fail silence enforcement entity, which was implemented in a Xilinx FPGA Spartan-6. It receives the data generated by the digital transmission chains of the two IT2S Boards (10 bits of In-phase + 10 bits of Quadrature samples @10 Msps), which corresponds to signals DATA 0 and DATA 1. For each of these signals, there is a valid signal (VALID 0 or VALID 1) to indicate whether a frame is being transmitted or not in the data bus. After deserializing and removing the invalid bits, data from both sources of information are compared in the Value Domain Comparison module. Furthermore, the fail-silence entity also verifies if the two platforms are synchronized, i.e., producing information at the same time. It only allows a small time offset, to cope with the fact that the platforms may not be precisely aligned in time. However, this time tolerance should be small enough to guarantee the timely behaviour of the communications protocol (e.g. V-FTT). This operation is performed in the Time Domain Comparison module, based on the analysis of phase offset between the two valid signals.

For a given frame, if the results produced by these two comparison modules are both positive, a strobe signal is generated and sent to the primary platform (the one

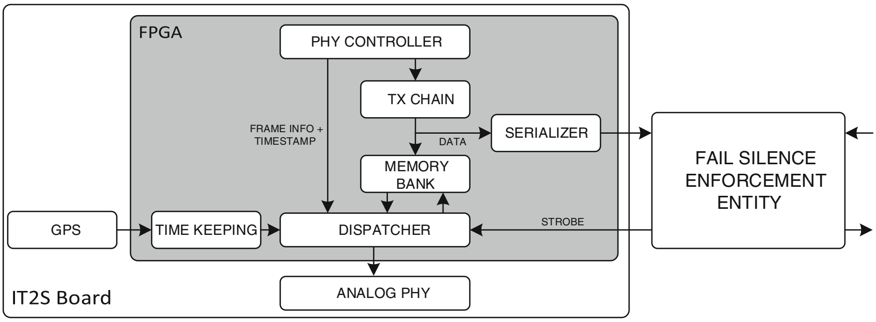


Fig. 8.10 Digital PHY scheme with fail silence behaviour

that will effectively send the message to the air). Otherwise, if at least one of the values is negative, no signal is generated and the entire RSU operation will stop until a restart signal is received in the fail silence enforcement entity.

The interaction of the IT2S Board with the fail silence enforcement entity is presented in Fig. 8.10. It depicts the architecture of the digital PHY in transmission mode and its integration with the fail silent scheme. A frame coming from the LMAC sublayer is handled in the PHY Controller module, which forwards its content to the digital transmission chain (OFDM modulator). Then, the resulting I/Q data samples are sent to the Serializer, in order to be verified by the fail silence enforcement entity, which will compare them with the ones provided by the other IT2S platform. The same samples are stored in a Memory Bank, waiting for the strobe signal to be sent to the wireless medium.

In case of successful frame validation, the Dispatcher module will receive the strobe indication and will start to prepare the transmission of the frame. Hence, it will compare the timestamp provided by the PHY Controller module, specifying in which moment the message should be sent, with the current time of the system, available from the GPS receiver. When both time values are equal, the Dispatcher reads the samples stored in the Memory Bank and send them to the Analog PHY. On the other hand, if the verification process performed by the fail silence entity fails, no strobe signal is received by the Dispatcher and consequently, no message will be issued to the wireless medium.

8.6 Experimental Evaluation and Results

The proposed architecture (Fig. 8.9) was successfully implemented in a Spartan-6 LX150 FPGA using the Trenz TE0300 carrier board. To validate the proposal, both comparison modules were tested. When two equal frames were sent with a phase offset below the maximum time tolerance limit, the strobe signal was successfully generated. However, when at least one bit error was introduced in one of the frames

Fig. 8.11 Fault detection in value domain

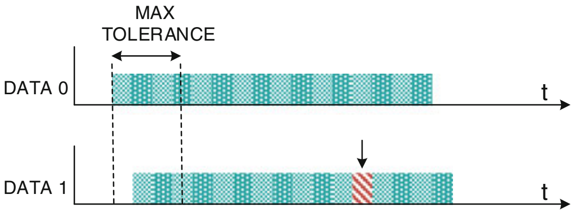
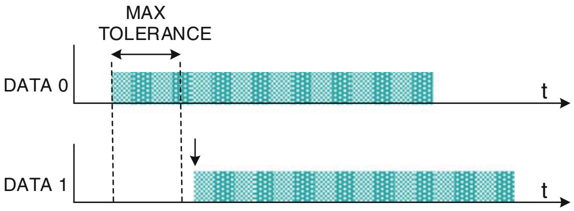


Fig. 8.12 Fault detection in time domain



(Fig. 8.11), the value domain comparison module was able to detect it, identifying a fault in the normal operation of the system. In this case, no strobe signal was generated and the fail silence entity entered into an idle state until a reset signal was received. This fault injection mechanism successfully tested the operation of the fail silence entity when the frames sent to validation were completely different as well as when there was only one error bit randomly inserted in a sample of one of the frames (as illustrated by the red striped mark in Fig. 8.11). The same result was achieved when a delay greater than the maximum tolerance allowed was introduced in the transmission of one of the frames (Fig. 8.12). In this situation, the time domain comparison module detected the excessive time difference between the beginning of the two frames and, similarly to the previous case, it signalled a fault to the strobe signal generation module.

As a proof of concept, the fail silence enforcement entity was developed in the same FPGA model that was used to implement the Lower MAC and the physical layer of the IT2S board. However, when comparing the resource usage of both projects, it can be concluded that the fail silence enforcement entity occupies much less resources (Table 8.1), around 1 % of the total available, than the project for the IT2S board (Table 8.2).

Table 8.1 Fail silence enforcement entity—resource usage on Spartan-6 FPGA

Logic resources	Used	Total	Percentage used (%)
Flip flops	314	184304	1
LUTs	284	92152	1
RAMB16BWERs	4	268	1
RAMB8BWERs	4	536	1
DSP48A1s	0	180	0
Logic max. frequency—160.805 MHz			

Table 8.2 IT2S board—resource usage on Spartan-6 FPGA

Logic resources	Used	Total	Percentage used (%)
Flip flops	39292	184304	21
LUTs	38111	92152	41
RAMB16BWERs	196	268	73
RAMB8BWERs	48	536	8
DSP48A1s	80	180	44
Logic max. frequency—158.188 MHz			

This result proves that the fail silence entity is much simpler than the entire vehicular communications platform and therefore, it is less prone to design errors and possible faults that can occur during system's operation. The simplicity of this unit is an extremely important characteristic that can be used to achieve a higher level of reliability, thus improving the dependability of the road-side infrastructure. Furthermore, in a future iteration of the fail silence enforcement entity, a smaller, less expensive FPGA, should be considered in order to reduce the implementation cost of this mechanism.

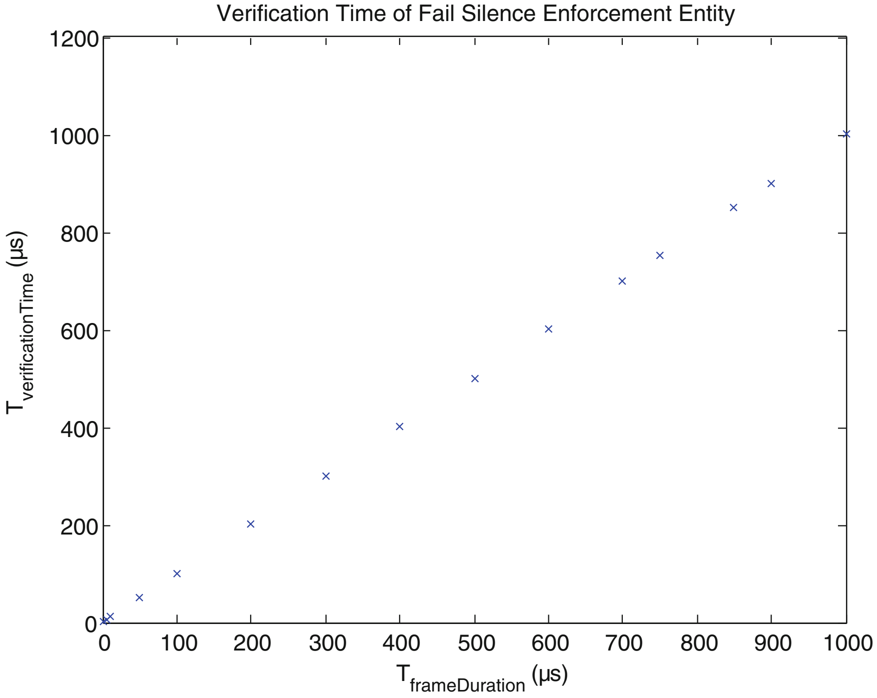


Fig. 8.13 Verification time as a function of frame duration

The frame verification time by the fail silence entity is another important aspect that should be considered when this mechanism is used to validate the operation of a deterministic wireless protocol, such as V-FTT. This verification time adds a delay to the beginning of the frame's transmission in the wireless medium, which should be taken into account by the real-time MAC protocol in use. For instance, this effect can be compensated if the frame is sent for validation with sufficient time in advance. Figure 8.13 shows the total delay introduced by the operation of the fail silence enforcement entity (FSEE) for various frame durations, from 1 μs to 1 ms. This verification time ($T_{verificationTime}$) is essentially constituted by two components (Eq. 8.1): the time that takes to transmit the entire message to the fail silence entity ($T_{frameDuration}$) and the delay introduced by the blocks inside the FPGA (T_{FSEE}). The latter one is constant and is approximately equal to 1.05 μs, while the first one is equal to the frame duration. Thus, the delay introduced inside the fail silence entity becomes negligible when the frame size increases, and in that case the frame verification time is approximately equal to the frame duration.

$$T_{verificationTime} = T_{frame\ Duration} + T_{FSEE} \tag{8.1}$$

Therefore, the samples should start to be sent to the fail silence enforcement entity, with a time advance ($T_{advance}$) greater than the one given by Eq. 8.2, so the message could be send to the air at the exact planned instant. The maximum tolerance allowed also contributes to this guard interval, since the verification time was measured when both platforms transmitted the same message at the same time, not considering a possible misalignment in the absolute clock sources.

$$T_{advance} \geq T_{verificationTime} + T_{maxTolerance} \tag{8.2}$$

This $T_{advance}$ time added to the normal operation of the system during the transmission of a message is completely acceptable, since the RSUs can start to prepare the packets for transmission with a large time advance. Given the fact that RSUs have a global vision of the road trough the backhauling network, they can easily compute the schedule of the next elementary cycle at the beginning of the current one. This represents an advance of approximately 100 ms (the total EC duration), which is more than enough to allow the inclusion of the fail silence mechanism in the operation of the system.

8.7 Conclusions

This chapter presented a fault tolerant architecture for infrastructure based vehicular networks. The rationale for designing dependable vehicular communication systems was explained together with the advantages of deploying networks that rely on the support provided by the road-side units. In addition to this, a brief overview of the MAC issues in dense vehicular scenarios was given, in order to explain the need for real-time behaviour in the operation of the communications protocol. V-FTT protocol was presented as a possible solution to overcome the issues found in the previous proposals, and it was chosen as a case of study. As a result, the dependable vehicular network architecture was designed, by taking V-FTT protocol as an example of a deterministic medium access control scheme. Nevertheless, the proposed architecture is protocol independent and thus can be applied to any system based on the same principles.

Then, some techniques typically employed in the design of fault-tolerant systems were surveyed, leading to the definition of a fault-tolerant architecture and a fault hypothesis for V-FTT networks. As a consequence, several mechanisms were proposed namely, the replication of the road-side infrastructure, a fail silence enforcement entity for RSUs and medium guardians for OBUs. In the rest of the chapter, special attention was paid to the design of fail-silent RSUs, since fail silence behaviour in the master nodes of the network is an essential property that should be attained, in order to provide dependable network operation. The design choices behind the proposed fail silence enforcement entity were presented together with its final structure and respective integration with the operation of a custom vehicular communication station, the IT2S platform.

In summary, the fail-silence mechanism compares the output messages produced by two vehicular communications platforms, both in value and time domain. Ideally, these two systems should produce the same outputs based on completely different hardware and software implementations. This should be done to avoid common mode failures. However, as a proof of concept, two identical IT2S platforms were used. The obtained results show that the developed mechanism successfully detects system faults both in time and value domain. The mechanism can be implemented in an FPGA with few resources, but it should belong to a separate fault confinement region with different power supply and clock source. Moreover, the delay introduced by the operation of the fail silence entity could be significant for large frame sizes, so it should be taking into consideration when analyzing the whole performance of the wireless communications protocol.

Acknowledgments This work is funded by National Funds through FCT—Fundação para a Ciência e a Tecnologia under the Ph.D. scholarship ref. SFRH/BD/52591/2014, by the European Union's Seventh Framework Programme (FP7) under grant agreement n. 3176711 and by BRISA, under research contract with Instituto de Telecomunicações—Aveiro.

References

1. A. Avizienis et al., Basic concepts and taxonomy of dependable and secure computing. IEEE Trans. Dependable Secure Comput. **1**(1), 11–33 (2004)
2. G. Bansal, J.B. Kenney, Controlling congestion in safety-message transmissions: a philosophy for vehicular DSRC systems. IEEE Veh. Technol. Mag. **8**(4), 20–26 (2013). ISSN: 1556–6072. doi:10.1109/MVT.2013.2281675
3. R. Belschner et al., FlexRay requirements specification, version 2.0.2. FlexRay Consortium (2002). http://www.flexray-group.com

4. P. Bernstein, Sequoia: a fault-tolerant tightly coupled multiprocessor for transaction processing. IEEE Comput. **21**(2), 37–45 (1988)
5. A. Böhm, M. Jonsson, *Real time communications support for cooperative, infrastructure-based traffic safety applications* (Int. J. Veh, Technol, 2011)
6. F.V. Brasileiro et al., Implementing fail-silent nodes for distributed systems. IEEE Trans. Comput. **45**(11), 1226–1238 (1996). ISSN: 0018–9340. doi:10.1109/12.544479. http://dx.doi.org/10.1109/12.544479
7. N. Budhiraja, K. Marzullo, F.B. Schneider, S. Toueg, in Distributed Systems, 2nd edn., ed. by S. Mullender. The Primary-backup Approach (ACM Press/Addison-Wesley Publishing Co., New York, 1993) pp. 199–216
8. C. Campolo, A. Molinaro, Multichannel communications in vehicular ad hoc networks: a survey. IEEE Commun. Mag. **51**(5) 158–169 (2013). ISSN: 0163–6804. doi:10.1109/MCOM.2013.6515061
9. X. Défago, A. Schiper, N. Sergent, Semi-passive replication, in Proceedings of the The 17th IEEE Symposium on Reliable Distributed Systems, SRDS '98 (IEEE Computer Society, Washington, 1998) pp. 43–50
10. IEEE Standard for Wireless Access in Vehicular Environments (WAVE), Multi-channel Operation, in IEEE Std 1609.4-2010 (Revision of IEEE Std 1609.4-2006) (2011), pp. 1–89. doi:10.1109/IEEESTD.2011.5712769
11. Intelligent Transport Systems (ITS), Performance evaluation of self-organizing TDMA as medium access control method applied to ITS; access layer part, in ETSI TR 102 862 V1.1.1, Dec 2011, pp. 1–51
12. J.B. Kenney, Dedicated short-range communications (DSRC) standards in the United States, in Proceedings of the IEEE 99.7, July 2011, pp. 1162–1182. ISSN: 0018–9219. doi:10.1109/JPROC.2011.2132790
13. S. Khan, P. Pedreiras, J. Ferreira, Improved real-time communication infrastructure for ITS, in INForum. Sept **2014**, 430–445 (2014)
14. H. Kopetz, Real-Time Systems: Design Principles for Distributed Embedded Applications. (Kluwer Academic Press, 1997)
15. H. Kopetz, G. Grunsteidl, TTP-a protocol for fault-tolerant realtime systems. Computer **27**(1), 14–23 (1994)
16. L. Lamport, R. Shostak, M. Pease, The byzantine generals problems. ACM Trans. Program. Lang. Syst. **4**(3), 382–401 (1982)
17. N. Lu et al., A distributed reliable multi-channel MAC protocol for vehicular ad hoc networks, in Intelligent Vehicles Symposium, 2009 IEEE, June 2009, pp. 1078–1082. doi:10.1109/IVS.2009.5164431
18. T.K. Mak, K.P. Laberteaux, R. Sengupta, A multi-channel VANET providing concurrent safety and commercial services, in Proceedings of the 2Nd ACM International Workshop on Vehicular Ad Hoc Networks, VANET '05 (ACM, Cologne, 2005), pp. 1–9. doi:10.1145/1080754.1080756. http://doi.acm.org/10.1145/1080754.1080756
19. A. Mehra, J. Rexford, F. Jahanian, Design and evaluation of a windowconsistent replication service. IEEE Trans. Comput. **46**(9), 986–996 (1997)
20. V. Milanes et al., An intelligent V2I-based traffic management system. Intell. IEEE Trans. Transp. Syst. **13**(1), 49–58 (2012). ISSN: 1524–9050. doi:10.1109/TITS.2011.2178839
21. D. Powell, Delta-4–A generic Architecture for Dependable Distributed Computing (ESPRIT Research Reports, 1991)
22. J. Proenza, J. Miro-Julia, *MajorCAN: a modification to the controller area network to achieve atomic broadcast, in IEEE International Workshop on Group Communication and Computations* (Taipei, Taiwan, 2000)
23. J. Rezgui, S. Cherkaoui, O. Chakroun, Deterministic access for DSRC/802.11p vehicular safety communication, in Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International, July 2011, pp. 595–600
24. F.B. Schneider, Implementing fault-tolerant services using the state machine approach: a tutorial. ACM Comput. Surv. **22**(4), 299–319 (1990)

25. S. Shrivastava et al., Principal features of the voltan family of reliable node architectures for distributed systems, in IEEE Trans. Compute. (Special Issue on Fault-Tolerant Computing) **41**(5), 542–549 (1992)
26. T. Meireles, J. Fonseca, J. Ferreira, The case for wireless vehicular communications supported by roadside infrastructure, in Intelligent Transportation Systems Technologies and Applications (John Wiley and Sons, 2014)
27. C. Temple, Avoiding the babbling-idiot failure in a time-triggered communication system, in Fault Tolerant Computing Symposium (IEEE Computer Society, 1998), pp. 218–227
28. TTTech, Time-Triggered Protocol TTP/C High-Level Specification Document, 1.0 edn. (2002). http://www.ttagroup.org
29. P. Veríssimo, Uncertainty and predictability: can they be reconciled?, in Future Directions in Distributed Computing LNCS 2584 (Springer-Verlag, May 2003)
30. S. Webber, J. Beirne, The stratus architecture, in Digest of Papers FTCS-21 (1991), pp. 79–85
31. J. Wensley et al., SIFT: design and analysis of a fault tolerant computer for aircraft control. Proceedings of IEEE **66**(10), 1240–1255 (1978)
32. M. Wiesmann et al., Understanding replication in databases and distributed systems, in 20th International Conference on Distributed Computing Systems. Proceedings, (2000), pp. 464–474
33. H. Zou, F. Jahanian, A real-time primary-backup replication service. IEEE Trans. Parall. Distrib. Syst. **10**(6), 533–548 (1999)

Chapter 9
Exploring Seamless Connectivity and Proactive Handover Techniques in VANET Systems

Glenford Mapp, Arindam Gosh, Vishnu Vardhan Paranthaman, Victor Otite Iniovosa, Jonathan Loo and Alexey Vinel

Abstract In order to provide Dependable Vehicular Communications for Improved Road Safety, it is necessary to have reliable Vehicular-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V) communication. Such requirements demand that the handover process as vehicles move between adjacent Roadside Units (RSUs) be examined in detail to understand how seamless communication can be achieved. Since the use of beacons is a key part of VANETs, it is necessary to investigate how the beaconing process affects the opportunities to effect handovers. A framework is needed to be able to calculate the regions of overlap in adjacent RSU coverage ranges to guarantee ubiquitous connectivity. A highly mobile environment, therefore, makes this a serious challenge and points to the need to look at proactive handover techniques. This chapter, therefore, explores the development of the proactive handover mechanisms required to provide seamless connectivity and dependable communication in VANET environments.

G. Mapp (✉) · A. Gosh · V.V. Paranthaman · V.O. Iniovosa · J. Loo
School of Science and Technology, Middlesex University, London, UK
e-mail: G.Mapp@mdx.ac.uk

A. Gosh
e-mail: A.Ghosh@mdx.ac.uk

V.V. Paranthaman
e-mail: V.Paranthaman@mdx.ac.uk

V.O. Iniovosa
e-mail: V.Iniovosa@mdx.ac.uk

J. Loo
e-mail: J.Loo@mdx.ac.uk

A. Vinel
School of Information Technology, Halmstad University, Halmstad, Sweden
e-mail: alexey.vinel@hh.se

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_9

9.1 Introduction

The development of Smart Cities will play an important part in the development of sustainable living and can therefore positively influence the well-being of the entire planet. This requires the implementation of Intelligent Transport Systems (ITS) using Vehicular Ad-hoc Networks (VANETs). This, in turn, will allow several new applications to be available for road-safety, traffic efficiency, and infotainments (i.e., information and entertainment applications). This optimised transportation system should lead to better overall energy use, reduced driving times, less accidents, better traffic management and more efficient road maintenance. However, in order to implement these applications, it is necessary to have seamless communication between the Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V). In this regard, it is, therefore, essential to carefully examine how mechanisms for seamless communication can be developed [3]. This requires a detailed understanding of the communication mechanisms within the VANET framework.

In VANETs, beacons are used to discover and maintain neighbour relationships. The European ITS VANET Protocol (EVIP) defines Beacons as Cooperative Awareness Messages (CAMs). Beacon messages are generated and issued periodically between the cars and the Road-Side Unit (RSU) (V2I). In the context of handovers, beacons are therefore used to indicate the presence of new networks as the vehicle moves around and hence beacons can be used for developing reliable handover. Beaconing mechanisms will, therefore, affect the overall performance of the handover and this chapter investigates the interaction between beaconing, the mobile environment including the velocity of the vehicles as well as the location of RSUs to achieve seamless handover.

This chapter addresses these issues in detail. It first focuses on the classification of handover mechanisms. It then uses key parameters of this analysis and applies them to the vehicular environment by comparing theoretical and measured values from simulation. An investigation is then performed to explain the differences obtained. This leads to a new model of the handover process based on cumulative probability, which is explored using an analytical model showing how communication changes as the vehicle approaches a new RSU. An approximate model is then developed to examine these issues further. Finally, a prototype VANET Testbed to test the simulation and analytical models is discussed.

9.2 Understanding Handover in Detail

9.2.1 General Characteristics of Handover

Handover is defined as the changing of the Point of Attachment (PoA) of a Mobile Node (MN) to a network. Handovers may generally be categorised as follows:

- Horizontal versus Vertical: In horizontal handovers, the point of next attachment is of the same technology as the previous PoA. For example, 3G to 3G or WiFi to WiFi. By contrast, in vertical handovers, the new PoA is of different technology compared with previous PoA. Hence, vertical handovers are challenging because when they occur they can be accompanied by huge changes in the Quality of Service (QoS) of the two networks involved in the handover. The management of the different QoS is an important part of providing seamless communication.
- Hard versus Soft: In hard handovers, the connection to the previous PoA is broken before the connection to the new PoA is established (i.e., break before make). By contrast, in soft handovers, the connection to the new PoA is established before the connection to the previous PoA is broken (i.e., make before break). Compared to hard handovers, soft handovers, therefore, result in less disruption.
- Upward versus Downward: In upward handover, the communication on the MN is moving from a network of small coverage to a network of larger coverage (e.g., going from a WiFi network to a LTE/3G network). By contrast, in downward handovers, the MN is going from a network of large coverage to a network of smaller coverage (i.e., going from a LTE/3G network to a WiFi network).
- Network-based versus Client-based: In network-based handover, the network is responsible for executing the handover, while in a client-based handover the client is responsible for executing the handover. This means that for client-based handovers, the MN must acquire all the relevant network resources to achieve handover.

9.2.2 Advanced Classification of Handover

Y-Comm is an architecture (shown in Fig. 9.1) that has been designed to build future mobile networks by integrating communications, mobility, QoS and security. It accomplishes this by dividing the Future Internet into two frameworks: Core and Peripheral Frameworks. The researchers of Y-Comm have made major contributions in the areas of proactive handover to provide seamless communication, QoS, as well as security [13].

An advanced classification of handover has been proposed by the Y-Comm Project [12] and is shown in Fig. 9.2. Handovers can also be divided into two advanced types. Imperative handovers occur due to technological reasons only. Hence the MN changes its network attachment because it has determined by technical analysis that it is good to do so. This could be based on parameters such as signal strength, coverage, and the quality-of-service offered by the new network. These handovers are imperative because there may be a severe loss of performance or loss of connection if they are not performed. In contrast, alternative handovers occur due to reasons other than technical issues [11]. The factors for performing an alternative handover include a preference for a given network based on price or incentives. User preferences based on features or promotions as well as contextual issues might also cause handover.

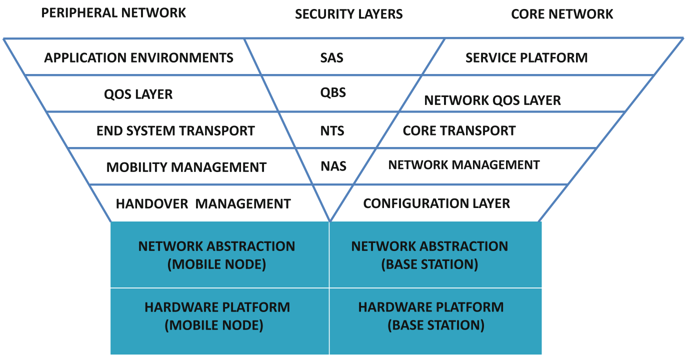


Fig. 9.1 YComm architecture

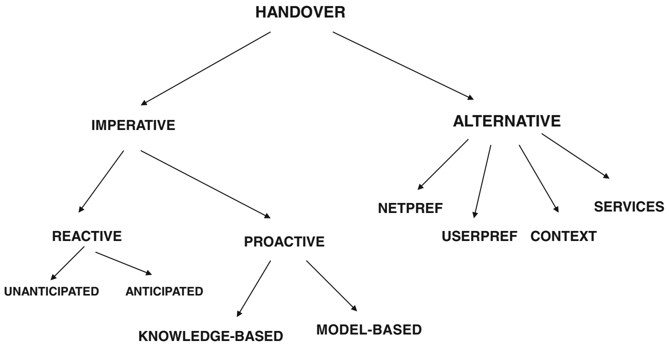


Fig. 9.2 Handover classification

Imperative handovers are, in turn, divided into two types. The first is called reactive handover. This responds to changes in the low-level wireless interfaces as to the availability of certain networks. Reactive handovers can be further divided into anticipated and unanticipated handovers [11]. Anticipated handovers are therefore soft handovers that describe the situation where there are alternative base-stations to which the mobile node may handover [2]. With unanticipated handover, the mobile node is heading out of range of the current PoA and there is no other base-station to which to handover. The other type of imperative handover is called proactive handover. These handovers use soft handover techniques.

Presently, two types of proactive handovers are being developed. The first is knowledge-based, where the MN attempts to know, by measuring beforehand, the signal strengths of available wireless networks over a given area such as a city. This most likely will involve physically driving around and taking these readings. The second proactive policy is based on a mathematical model which calculates the point when handover should occur and the time that the mobile would take to reach that point based on its velocity and direction. The accuracy of this approach is dependent on various factors including location technology, the propagation model

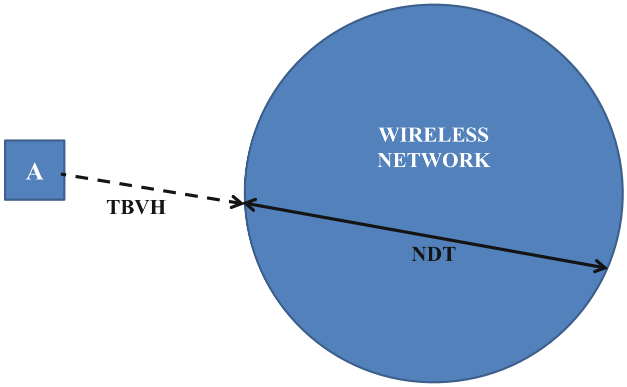


Fig. 9.3 Illustrating Time Before Vertical Handover & Network Dwell Time

used, network topology, and specific environments, for example, whether the MN is indoor or outdoor as well as the quality of the receiver [11].

Proactive handover policies attempt to know the condition of the various networks at a specific location before the MN reaches that location [1]. Two key parameters are used to develop algorithms for proactive handover: Time Before Vertical Handover (TBVH) which is the time after which the handover should occur, and Network Dwell Time (NDT) which is the time the MN spends in the coverage of the new network as shown in Fig. 9.3. According to the article [11], with an accurate measurement of the handover radius, it is possible to accurately estimate TBVH as well as NDT. By using these mechanisms, it is possible to minimize packet loss and service disruption as an impending handover can be signalled to the higher layers of the network protocol stack.

9.3 Applications of NDT and TBVH to Highly Mobile Environments

In order to develop mechanisms for seamless handover for highly mobile environments such as VANETs, it is therefore necessary to have a very accurate estimate of the NDT i.e., how long is the vehicle able to communicate in a given network. Handover in mobile environments can be depicted as shown in Fig. 9.4. There is a Hard Handover Threshold circle depicted by the hard barrier and there is a dotted circle within the hard barrier representing the Exit Threshold. The Exit Threshold circle is the boundary to start handover and in order to finish the handover before reaching the hard barrier, which is needed for a successful soft handover. If the handover is not successful before the hard barrier is reached, there is a break in the communication which leads to a hard handover.

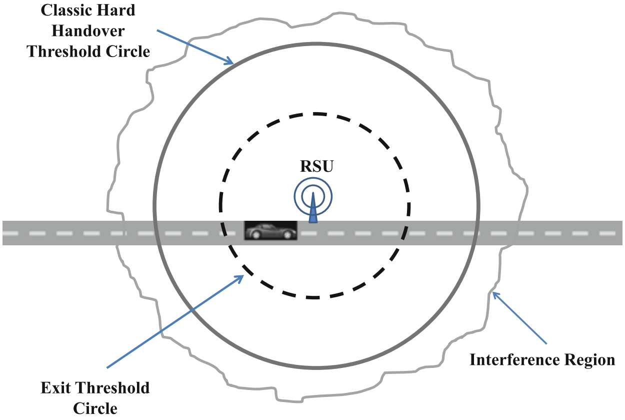


Fig. 9.4 Handover approach

9.4 Comparing Idealised NDT to Measured NDT

NDT is the time a vehicle spends in a RSU's network range. If this time can be estimated even before a vehicle enters the communication range, then the resources can be used in an efficient way and proactive handover can be made possible which ensures ubiquitous communication. NDT in a wireless network is given by the reciprocal of the mobility leave rate. According to [20], for a two-dimentional fluid flow model, the average outgoing rate μ_{dwell} of a mobile unit within a cell is given by

$$\mu_{dwell} = E[V]L/(\pi A) \tag{9.1}$$

where E[V] is the average velocity, L is the length of the perimeter of a cell with arbitrary shape and A is the area of the cell.

Using this approach it is possible to estimate quite accurately the NDT as shown in Fig. 9.5 based on the handover model. For a straight road this is approximately equal to 2R, as it is assumed that the RSU is placed along a straight road.

$$NDT = 1/\mu_{ml} = (\pi \times R_H)/V_{max} \tag{9.2}$$

where, $\mu_{ml} \rightarrow$ Mobility leave rate from equation(EquationMobilityLeaveRate)
$R_H \rightarrow$ Handover radius
$V_{max} \rightarrow$ Maximum velocity of the vehicle

$$NDT = 2R/E[V] \tag{9.3}$$

In motor way context, the distance between two travelling points can be directly calculated. Hence NDT is given as shown below

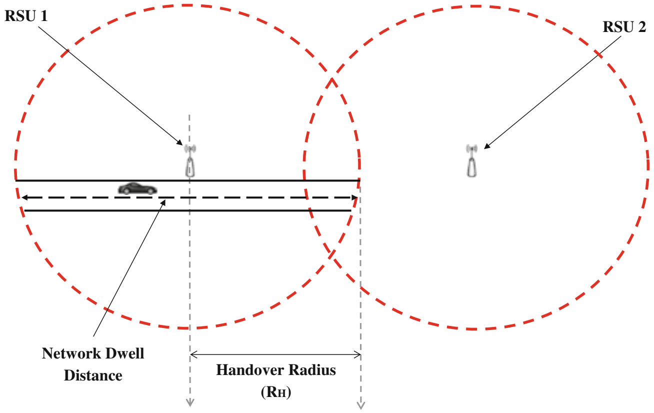


Fig. 9.5 Handover radius

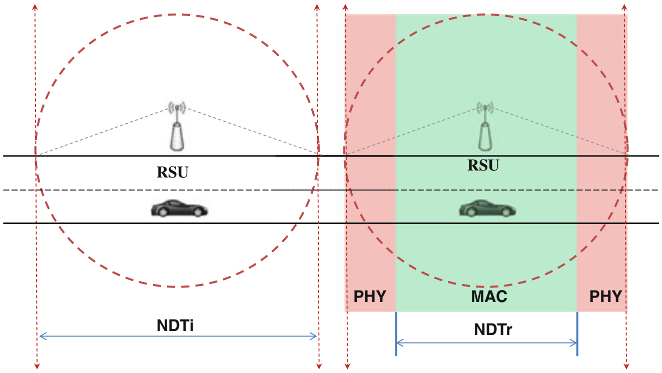


Fig. 9.6 NDTi versus NDTr

$$NDT = NDD/E_{vel} \tag{9.4}$$

where, NDD is Network Dwell Distance travelled along a motorway that is in coverage of a given network [11]. The exact distance between two points on a motorway can be calculated using Global Positioning System (GPS). For our study in VANETs, we assume that the RSU is alongside a straight road hence NDD is approximately equal to 2R where R is the radius of coverage. In our calculations, this is called ideal NDT denoted as (NDTi). It is assumed that the communication starts as soon as the vehicle hits the edge of the coverage of a communication range. However, in real-time, the measured definition of NDT, NDTr, can be defined as the time between the first and the last beacon reaching the MAC layer without being dropped in the PHY layer due to bit error as shown in Fig. 9.6. In VANETs, beaconing is one of the core communication modes, which is designed to advertise the presence of a car to its neighborhood [18].

Fig. 9.7 NDTr with different beacon sizes (30 m/s)

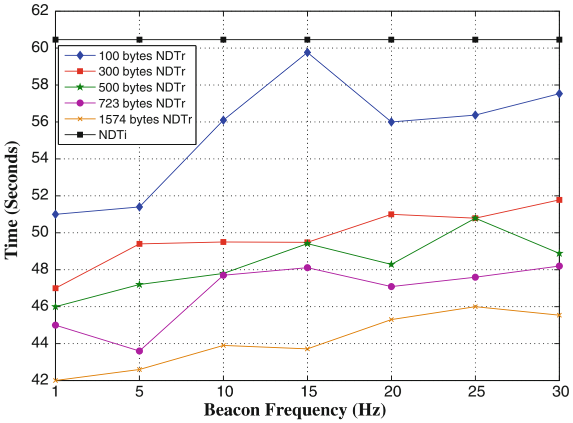


Table 9.1 Comparison of Network Dwell Time from simulation with theoretical calculation

Speed	NDTi	NDTr				
		$\lambda = 1\,\text{Hz}$	$\lambda = 5\,\text{Hz}$	$\lambda = 10\,\text{Hz}$	$\lambda = 20\,\text{Hz}$	$\lambda = 40\,\text{Hz}$
0–30 m/s	60 s	54 s	55 s	57 s	57 s	57 s
40 m/s	45 s	37 s	39 s	43 s	43 s	43 s
50 m/s	36 s	30 s	31 s	34 s	34 s	34 s

The graph in Fig. 9.7 shows the NDTr for different size of beacon broadcasted to the vehicle moving at a constant speed (30 m/s) with different beacon generation frequencies (λ). The NDTi is also plotted in this graph. It shows that as the beacon size increases the NDTr is reduced i.e., the communication time is reduced. This clearly shows that the size of the packet is an important factor in determining the NDTr. The graphs also clearly shows that there is no peak increase in NDTr after 10 Hz and it is also evident that some beacons are being dropped which causes this difference between NDTr and NDTi.

Our results showed that the difference between idealised and measure NDT was dependent on the frequency of the beacon, the size of the beacon and the velocity of the vehicle.

Table 9.1 shows the NDT values from simulation experiments (i.e., NDTr) with different λ from the RSU and NDT using the formula in Eq. (9.4) (i.e., NDTi) to calculate the upper bound. This upper bound does not consider any factors like contention, it assumes the medium or channel is ideal and that the only loss is due to propagation [8]. The reason and the way the beacons are dropped by the simulation in the PHY layer will be explained in the following section.

9.5 Veins Framework

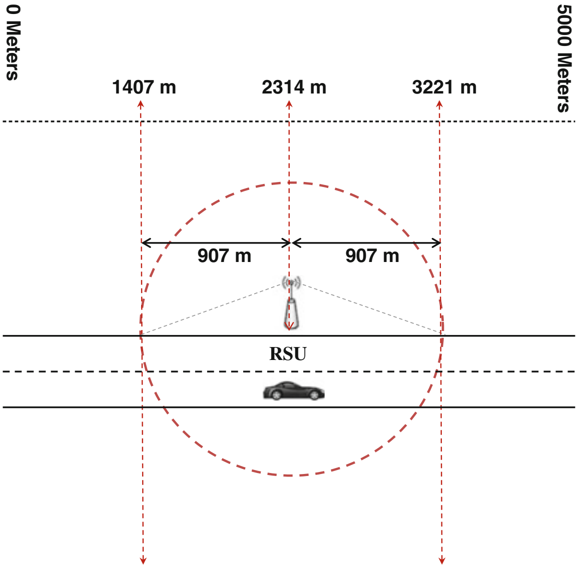
For the simulation experiments, a discrete event simulation environment OMNeT++ [17] is used in conjunction with the Veins framework [15, 16]. This is a mobility simulation framework for wireless and mobile networks. A beaconing model using IEEE 802.11p was implemented in Veins framework by [15]. All the PHY and MAC properties used in the IEEE 802.11p simulation model conform to [9, 10].

9.5.1 Simulation Scenario

A stationary node (i.e., RSU) is placed as shown in Fig. 9.8. Another mobile node (i.e., vehicle) is made to run over the range of the RSU for collecting various values for our study with a fixed velocity of 30 m/s. To understand and model a concept like NDT, which no other work has ever considered, we first have to start with a simple scenario. We have considered a very basic setup where there is no interference or other noises, effects of buildings and no traffic density issues in order to concentrate on the effect of beaconing, size of beacons and velocity of the vehicle on NDT. This will allow us to understand the key factors before studying more complex scenarios.

During simulation, the RSU broadcasts the beacon with different beacon generation rates and with different beacon sizes. Note that the simulation parameters contain the EDCA default values [5]. The remaining parameters are set according to the default values used by the Veins Framework [15].

Fig. 9.8 Simulation scenario



Beacon sizes of 100, 300, 500 and 723 bytes have been used in [14] for 6 Mbps packet error ratio modelling. This result was used in the development of Veins Framework in 6 Mbps packet error rate modelling. A beacon size of 1574 bytes has also been used in an experimental study [4]. Further, this result was used in the development of Veins Framework in 18 Mbps packet error rate modelling. Hence, these sizes are used in our simulation experiment.

9.5.2 Calculation of Detection Range in Simulation

Calculation of the Detection Range (DR) [15, 17] was based on transmitter power, wavelength, path loss coefficient and a threshold for minimal receive power for communication to take place is shown below.

$$DR = ((\lambda^2 \times pMax)/(16.0 \times \pi^2 \times minRecvPow))^{1/\alpha} \tag{9.5}$$

where, minRecvPow = $10^{sat/10}$
$\lambda \rightarrow$ Wavelength = (speedOfLight/carrierFrequency)
pMax → Maximum Transmission Power Possible
$\alpha \rightarrow$ Minimum path loss coefficient
sat → Minimum signal attenuation threshold
MinRecvPow → Minimum power level to be able to physically receive a signal.

Based on the simulation parameters as shown in Tables 9.2 and 9.3, the detection range is calculated in the simulation. The outcome from the formula suggests 907.842567 m i.e., the radius (R) of the coverage as shown in Fig. 9.8. For this reason all the mathematical calculations in our work has considered 907 m as the radius of the coverage.

Table 9.2 RSU configuration parameters

Parameter	Values
Transmission power	20 mW
Bit rate	18 Mbps
Sensitivity	−94.0 dBm
Thermal noise	−110.0 dBm
Header length	11 bytes
Beacon length	100, 300, 500, 723, 1574 bytes
Send data	False

Table 9.3 OBU configuration parameters

Parameter	Values
Speed	10, 30 m/s (36, 108 Km/h)
Channel bandwidth	10 MHz
OBU receiver sensitivity	−94.0 dBm

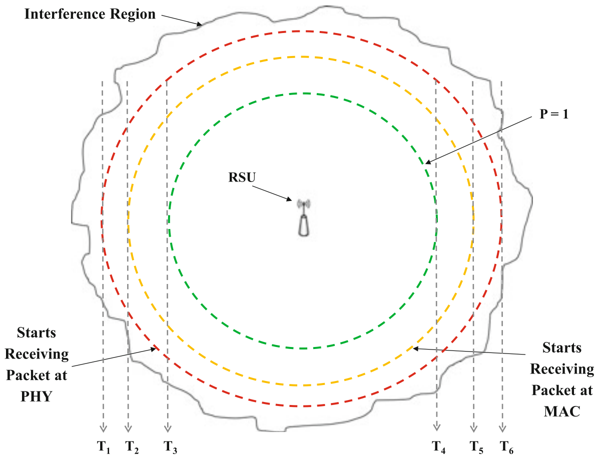

Interference Region
P = 1
RSU
Starts Receiving Packet at PHY
Starts Receiving Packet at MAC
T1
T2
T3
T4
T5
T6

Fig. 9.9 PHY and MAC segmentation

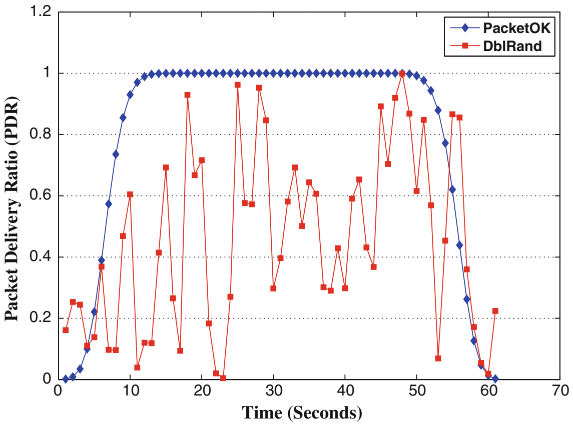
9.5.3 *Calculation of Successful Packet Reception in Simulation*

In Fig. 9.9, T1 and T2 is the time when the first packet at PHY and MAC layer are received respectively. Between T3 and T4 is the region where the packet is always successfully received i.e., where Probability (P) of successful packet reception is 1. T5 and T6 is the time when the last packet at MAC and PHY layer are received. All the packets between T1–T2 and T5–T6 are lost due to bit errors which show that the reliable communication starts only when the packet reaches the MAC layer. The reason and the way the packets are dropped by the simulation in the PHY layer are summarised below but explained in more detail in [7].

In the simulation, T2 is the time where the actual communication starts and we know that we receive packet at T1 but these received packets are discarded due to bit errors, hence the main issue to be addressed is: can this time, T2, be determined given that the vehicle receives the first packet in PHY layer at time T1? To analyse this effect, we further carefully investigate the calculation of the successful packet reception.

The graph, shown in Fig. 9.10, has been simulated in OMNeT++ using Veins Framework. The graph shows the PacketOk and Random double number. This simulation was carried out using only one RSU and one vehicle moving at speed of

Fig. 9.10 PacketOK versus DblRand



30 m/s. The RSU broadcasts beacons with a size of 956 bits at a beacon generation frequency of 1 Hz.

For each packet received at the PHY layer, a PacketOk number is computed which is the Probability (P) that the packet is received free of error. This PacketOk number is computed based on Bit Error Rate (BER) and length of packet. This computed double number is compared against a randomly generated double number ranging between 0 and 1. If this PacketOk (computed number) is less than the randomly generated number then that respective packet is dropped at the PHY layer, reason assumed that there is an error in the packet [7].

The lower (red) line is the randomly generated double number and the upper (blue) line is the computed PacketOk. If the PacketOk number is below the randomly generated number curve, it is assumed that there is an error in the packet, therefore that particular packet is dropped at the PHY layer itself. Packet Delivery Ratio in the Veins Framework for 18 Mbps bit rate is calculated using the below formula which has been modelled using [4].

$$PacketOk\ (P) = [1 - 1.5erfc(0.45\sqrt{SNR})]^L \tag{9.6}$$

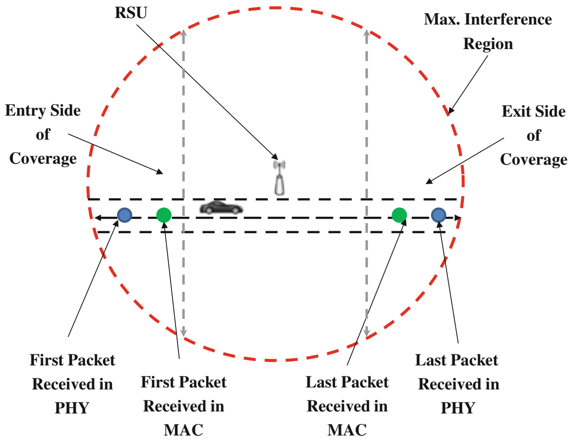
where,

$$SNR(SignaltoNoiseRatio) = 10^{SNR_{dB}/10} \tag{9.7}$$

and L → Length of the packet.

In Fig. 9.10, we can observe that as the vehicle is heading towards the RSU, P increases and at a point it reaches 1 which means there is no possibility of error in the packet. In other words we can say that the region where the P = 1 is a very reliable communication region. This is the time T3–T4 which has been shown in the Fig. 9.9.

Fig. 9.11 First & last beacon received at PHY & MAC layers


RSU
Max. Interference Region
Entry Side of Coverage
Exit Side of Coverage
First Packet Received in PHY
First Packet Received in MAC
Last Packet Received in MAC
Last Packet Received in PHY

9.5.4 *Further Investigation into PHY Layer in Relation to Beacon Size*

For further investigation, more simulations were carried out but this time monitoring the beacons received at the lower layer (i.e., the PHY layer). The illustration in Fig. 9.11 depicts the relation between first beacon and the last beacon that is received at the PHY and MAC layers in simulation. The simulation has been carried out for different sizes of beacons with different beacon frequency with the velocity of the vehicle being constant. It also shows the actual interference range or detection range calculated in the Veins Framework comparing with the beacons received.

The graphs in Figs. 9.12 and 9.13 show the first beacon reception at both PHY and MAC layers against simulation time during the entry by the vehicle in the coverage region. This clearly shows that the vehicle starts receiving the beacon at the PHY layer as soon as it enters the detection range. This detection range is the place where the minimum criteria for the communication to happen are met. The time delay between the PHY layer first beacon and the MAC layer first beacon is due to the loss of those beacons. The beacons are received by the PHY layer but with bit error in the beacon and hence dropped at the PHY layer. We can also see that when there is an increase in size of the beacon there is a delay in reception of the beacon at MAC layer i.e., more beacons are lost due to bit error at the PHY layer for beacons of larger sizes. We can conclude that the increase in size of beacon will reduce the value of NDT.

The graphs in Figs. 9.14 and 9.15 show the last beacon reception at both PHY and MAC layers against simulation time during the exit by the vehicle in the coverage region.

Fig. 9.12 Entry side of coverage area (10 m/s)

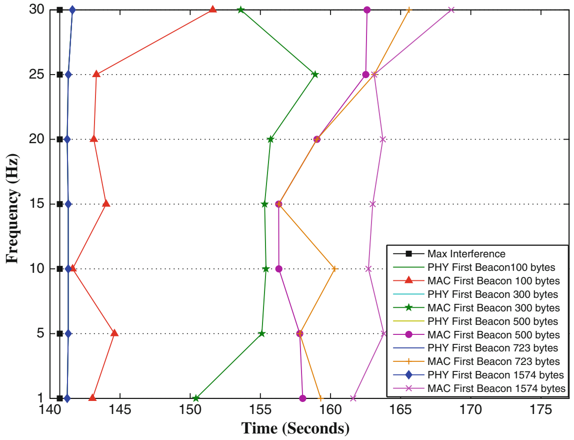


Fig. 9.13 Entry side of coverage area (30 m/s)

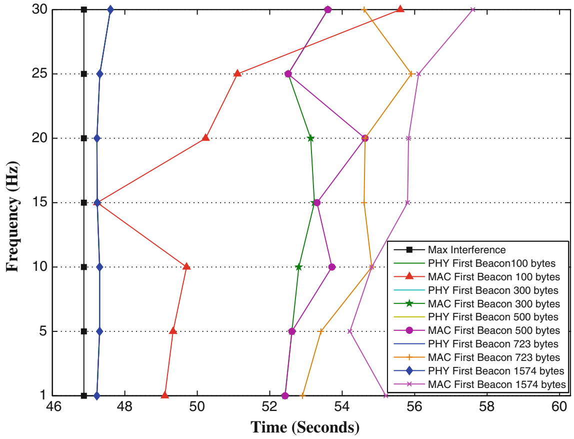
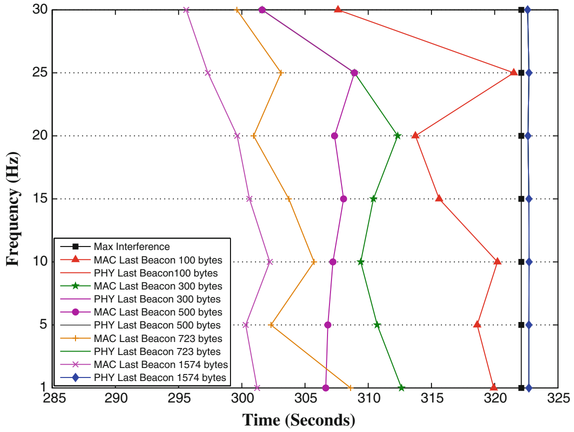


Fig. 9.14 Exit side of coverage area (10 m/s)



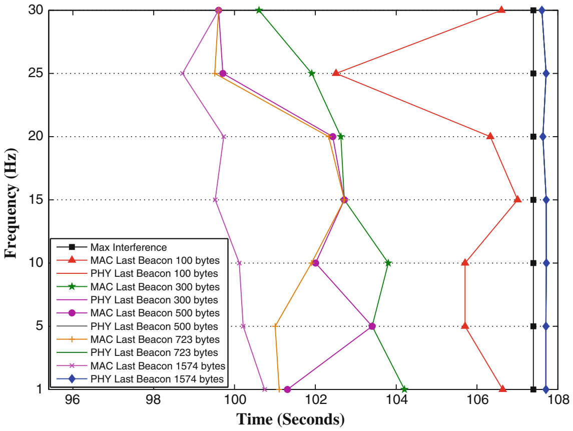


Fig. 9.15 Exit side of coverage area (30 m/s)

9.6 Further Investigation

9.6.1 Cumulative Probability Calculations

In order to investigate the effect of beacon frequency, we also need to look at the cumulative probability of a successful packet reception, in addition to calculating the probability of a successful packet reception for an individual packet at a given time ‘t’. Since we know the single packet reception probability using Eq. (9.8) from the simulation, the cumulative probability can be calculated.

Therefore, if P is the probability of a successful reception then the cumulative probability for a sequence of N receptions is given by:

$$P + (1 - P)P + (1 - P)^2 P + \cdots + (1 - P)^{N-1} P \tag{9.8}$$

In probability theory, P is constant and cumulative probability (CP) tends to 1 as N tends to infinity. In this case, it means that successful reception of the beacon is guaranteed once the CP reaches 1. But in this scenario because the vehicle is moving towards the RSU, P increases for every sequence. Therefore for N receptions the CP is:

$$CP = P_1 + (1 - P_1)P_2 + (1 - P_1)(1 - P_2)P_3 + \cdots \tag{9.9}$$

where, P_N is greater than $P_{N-1} \cdots = 1$

Since P is increasing because the vehicle is moving towards the RSU, hence the cumulative probability can reach 1 long before infinity and therefore affects the successful reception of the beacon. This analysis applies to when the vehicle enters the network.

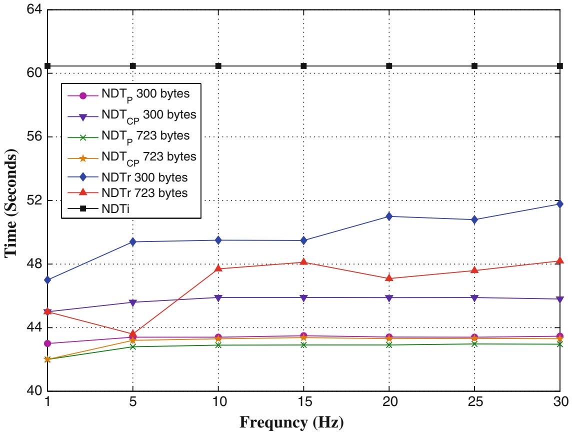


Fig. 9.16 Comparison of NDT_P versus NDT_{CP} and NDTr versus NDTi for two beacon sizes

For Exit times we consider the probability of not receiving the packet $P_n = 1 - P$ from the RSU as we drive away i.e., the negative cumulative probability. If P is the probability of successful reception the negative cumulative probability (CP_n) is given by:

$$CP_n = (1 - P_1) + P1(1 - P_2) + P1P2(1 - P_3) + \cdots \tag{9.10}$$

For the Exit scenario P the probability of the successful reception decreases as we move away from the RSU, hence $1 - P$ is increasing. Once the vehicle does not hear the beacon after the period T, the inverse of the beacon frequency, it immediately hands over to the next RSU. Our results consider the effect of the cumulative probability of entrance and exit region of RSU coverage.

The graph in Fig. 9.16 shows NDTr, NDTi, NDT_P and NDT_{CP} for two different size of beacon, where NDT_P is the NDT when P is equal to 1 and NDT_{CP} is the time between CP is equal to 1 and CP_n is equal to 1. It is clear that these values are affected by the size of the beacon. For relatively small beacon sizes NDT_{CP} is greater; but for much larger beacon sizes, the trend seems to be reversed. For beacon sizes around 723 bytes the NDT_{CP} and NDT_P are almost equal. This indicates that for handover where predictability is important, maximum beacon sizes around 600–800 bytes (approx.) could give the best chance for seamless communication [7].

9.6.2 Handover Policy Based on Cumulative Probability Approach

As indicated previously, P represents the probability of a successful reception of beacon at the Physical (PHY) layer. This probability can be calculated for each beacon with the knowledge of the SNR and the length of the beacon [4, 14]. In probability

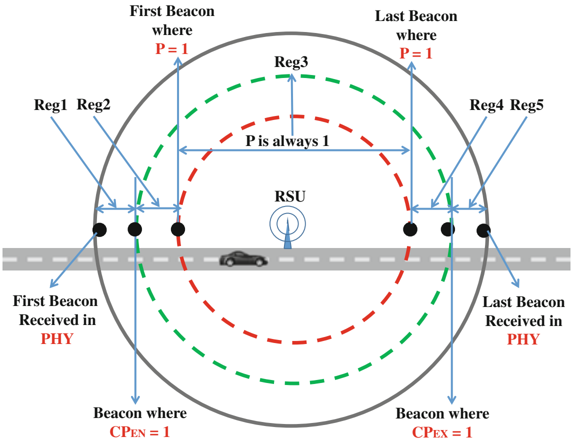


Fig. 9.17 Probabilistic segmentation

theory, P has a stationary distribution i.e., the possible outcomes are constant over time. Hence, we can define the Cumulative Probability as the probability of the event occurring—in this case, a successful beacon reception—before a given time or sequence number. In addition, when CP is 1, then we are sure that the event has occurred. If P is constant, then CP is normally 1 at infinity. In this case however, P does not have a stationary distribution because as the MN moves towards the RSU, P increases significantly and hence, CP may become 1 long before infinity and, in fact, may become 1 before P becomes 1. This shows that we can be certain of receiving a successful transmission before P become 1 due to CP. This means that it is necessary to use the CP approach to determine the regions of reliable communication. Therefore, we need to calculate CP for a sequence of N beacon receptions and compare it to when P is 1. This is illustrated in Fig. 9.17.

We define the CP as the vehicle enters a new network as the Cumulative Entrance Probability (CP_{EN}). For Exit scenarios, we consider the probability of not receiving the beacon P_n from the RSU as we drive away, i.e., the Exit Cumulative Probability, (CP_{EX}). For the Exit side, P, the probability of the successful reception decreases as we move away from the RSU. Hence $1 - P$ is increasing. Our results, therefore, consider the effect of the cumulative frequencies on entrance and exit regions of RSU coverage.

Figure 9.17 presents the communication time between the segments or regions named as Reg_1, Reg_2, Reg_3, Reg_4 and Reg_5. These regions are the communication times, i.e., the time duration when beacons are received by the vehicle in a particular segment of RSU coverage, which is also detailed in Table 9.4.

The simulation was carried out with one RSU and one vehicle moving along the road. The results in [6] also showed that for handover, a maximum beacon size between approximately 600 to 800 bytes could give the best chance for seamless communication. Hence, beacon sizes of 300, 500 and 723 bytes have been considered to conduct our study. In addition to this, the work in [6] also showed that an ideal

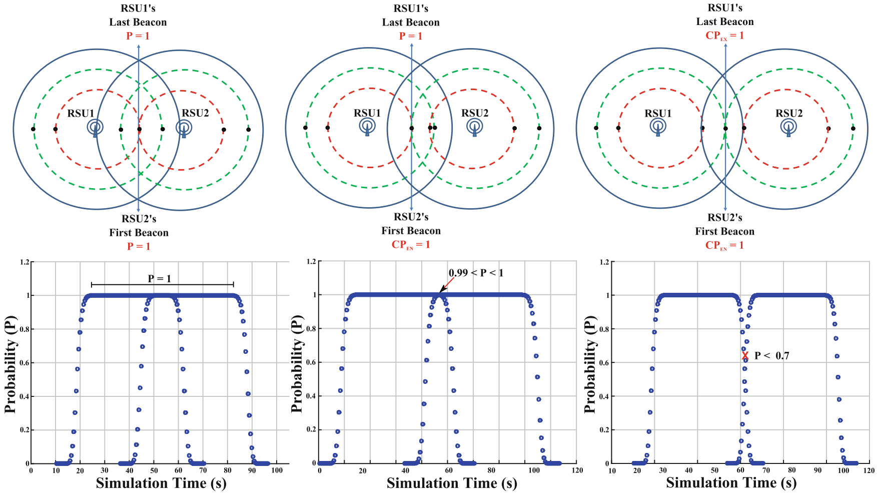
Table 9.4 Communication time in seconds between the segments

S.No	Beacon size (bytes)	Beacon frequency (Hz)	Reg_1 (s)	Reg_2 (s)	Reg_3 (s)	Reg_4 (s)	Reg_5 (s)
10 m/s							
1	300	10	41.8	8.9	80.5	46.1	4.6
2	300	15	38.46	12.73	81.26	43.73	7.4
3	300	20	35.85	14.75	80.5	41.5	9.1
4	500	10	43.3	7.9	79.5	43.2	8.0
5	500	15	40.13	11.53	80.33	41	10.6
6	500	20	37.6	13.5	79.5	38.95	12.15
7	723	10	44.2	7.5	78.5	41.4	10.3
8	723	15	41.26	10.93	79.26	39.4	12.73
9	723	20	38.75	12.85	78.5	37.45	14.15
30 m/s							
10	300	10	17.1	0	26.5	17.1	0
11	300	15	16.46	0.86	26.73	17.26	0
12	300	20	15.3	1.8	26.5	16.55	0.55
13	500	10	17.1	0	26.5	17.1	0
14	500	15	16.83	0.46	26.73	16.13	1.2
15	500	20	15.7	1.4	26.5	15.3	1.8
16	723	10	17.1	0	26.5	16.1	1.1
17	723	15	17.13	0.2	26.73	15.26	2.06
18	723	20	16	1.1	26.5	14.5	2.6

range of beacon frequency for vehicular communication is between 10 to 20 Hz. Hence, beacon frequencies of 10, 15 and 20 Hz are considered in this article. When there is an increase in beacon frequency, a considerable amount of communication time is achieved between $CP_{EN} = 1$ and P = 1 (i.e., Reg_2) and between $CP_{EX} = 1$ and P = 0 (i.e., Reg_5). This clearly indicates that a high beacon frequency should result in an increased NDT as the beacon is heard almost as soon the vehicle enters the coverage area.

9.6.3 Analysis of Overlapping Region

In order to verify our handover policy based on the CP approach, we have come up with three different scenarios of overlapping two RSUs as shown in Fig. 9.18. A mobile node (i.e., in our case a vehicle) is made to travel over the coverage range of these two RSUs with velocities of 10 and 30 m/s for collecting various values for our



Case (i)

Case (ii)

Case (iii)

Fig. 9.18 Overlapping scenarios

study. The same parameter settings were used as done for the one RSU simulation experiment setup for calculating CP.

Case (i) The two RSUs are overlapped such that RSU 1's last beacon received by the vehicle with P = 1 and RSU 2's first beacon with P = 1 are received one after another. The time difference between these two beacons is very small and hence Fig. 9.18 shows these two beacons at the same point.

Case (ii) The two RSUs are overlapped such that RSU 1's last beacon with P = 1 and RSU 2's first beacon reaching $CP_{EN} = 1$ are received one after another.

Case (iii) The two RSUs are overlapped such that RSU 1's beacon reaching $CP_{EX} = 1$ and RSU 2's beacon reaching $CP_{EN} = 1$ are received one after another. The simulation results for each case are illustrated as graphs in Fig. 9.18.

In Case (i), as mentioned earlier the overlapping of two RSUs are setup such that P is 1 for both RSUs at the overlapping region. Hence it is clearly evident from the graph that once the vehicle reaches the region where P = 1 of RSU 1, there is no drop in P till the vehicle exits the RSU2's P = 1 region, i.e., P is always 1 as shown in graph in Fig. 9.18. From this observation, it is clear that, this is the most reliable way of overlapping adjacent RSUs which ensures a seamless handover. But this reliability comes at the cost of more overlapping distance as shown in the graph in Fig. 9.18 and high interference issues as indicated in [5] as both RSUs are in communication range of each other.

In Case (ii), as the RSUs are setup such that of RSU 1's last beacon with P = 1 and CP_{EN} of RSU2 is 1 at the overlapping region. This way of overlapping yields us less overlapping distance as shown in Fig. 9.18 compared to case (i), however there is a very negligible amount of drop in P at the overlapping region i.e., $0.99 < P < 1$ as shown in Fig. 9.18. According to [19], P should be greater than 0.99 for the safety-related applications. Hence, case (ii) is equally reliable and also ensures seamless handover.

In Case (iii), the RSUs are setup considering CP_{EX} of RSU 1 and CP_{EN} of RSU2 for overlapping. This way of overlapping gives an advantage of a much smaller overlapping distance as compared to cases (i) and (ii). This also benefits the network with less interference as indicated in [5]. In the overlapping region, P reduces to less than 0.7 which is not suitable for safety critical applications but might be sufficient for entertainment applications which use reliable transport layer protocols such as Transmission Control Protocol (TCP).

As shown above Case (ii) performs equally as good as Case (i), therefore this approach can be adopted for a scenario where critical life-safety application are given higher priority. By contrast, the Case (iii) approach is more suitable for a scenario where optimal coverage is required and where non-safety applications are used.

9.6.4 The Change in Probability of Successful Beacon Reception (ΔP)

9.6.4.1 The Change (ΔP) at Entry

For the Entry Region the rate of change in P i.e., Probability of successful beacon reception is shown in the Eq. (9.11).

$$\Delta P_{ENTRY} = P_N - P_{N-1} \tag{9.11}$$

ΔP is significant because the SNR changes more rapidly with the increased velocity of the vehicle. Hence, ΔP increases significantly as the velocity of the vehicle increases. Where, P_N is the probability of packet reception of an individual packet ‘N’ and ‘N-1’ is the previous packet. ΔP is calculated until P reaches 1.

9.6.4.2 The Change (ΔP) at Exit

For the Exit Region the rate of change in P is as shown in the Eq. (9.12).

$$\Delta P_{EXIT} = P_N - P_{N+1} \tag{9.12}$$

where, P_N is the probability of packet reception of an individual packet 'N' and 'N+1' is the next packet. $\varDelta P$ is calculated until P reaches 0.

The graph is generated using the Eq. (9.13) which does not take into account the velocity of vehicle. We know that $\varDelta$P for second packet (i.e., N+1) with respect to first packet (i.e., N) can be calculated as,

$\Rightarrow \varDelta \text{P} = P_2 - P_1$

We know the formula for P i.e.,

Hence,

$$\varDelta P = [1 - 1.5erfc(0.45\sqrt{SNR_2})]^L - [1 - 1.5erfc(0.45\sqrt{SNR_1})]^L \quad (9.13)$$

The simulation experiments were conducted to analyse the change in P with respect to different velocities and different beacon frequencies. These results clearly show the effect of size of beacon, velocity of vehicle and frequency of beacon. If a formula is being modelled based on these results then for a given velocity of the vehicle, for a given beacon size and frequency; the rate of change of P can be calculated using the modelled formula. With this rate of change being known the P and CP at any point can be calculated, which in turn can be used to predict the NDTr more accurately.

Differentiation of P (Eq. 9.6) with respect to SNR (i.e., $\frac{dP}{dSNR}$) will yield us the $\varDelta$P for any given SNR which can be used to find the CP and when to Handover based on the prediction. The results are shown Fig. 9.19. However, from the Figure there is a fairly close match between the results measured using the simulation and those calculated using the equation. Hence, we can explore the situation further using the analytical approach.

$$\frac{dP}{dSNR} = L\frac{0.675}{\sqrt{\pi}}SNR^{-\frac{1}{2}}(1 - 1.5erfc(0.45\sqrt{SNR}))^{L-1}e^{-}((0.45)^2SNR) \quad (9.14)$$

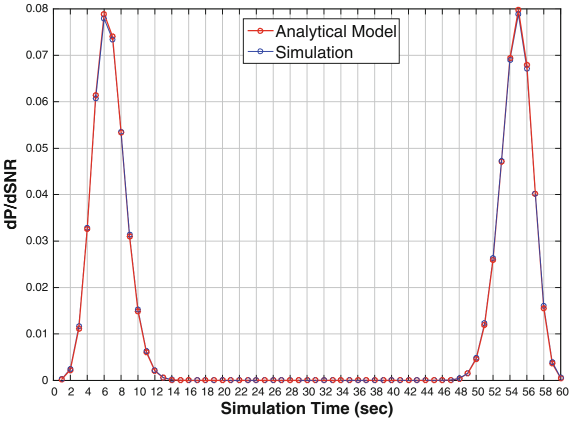


Fig. 9.19 Comparison of simulation versus analytical model

The results for the comparison of the simulation and the analytical model are shown in Fig. 9.19. It shows a very close match between the simulation and the analytical model and thus it is worth exploring the analytical model in more detail.

9.6.5 Further Observations

From both the simulation and calculated values of the $\frac{dP}{dSNR}$, we see that when P approaches 1, $\frac{dP}{dSNR}$ approaches 0. Furthermore, we know that as P approaches 1 i.e., $(1 - 1.5erfc(0.45\sqrt{SNR})$ goes to 1. This means that in the region of interest is,

$$\frac{dP}{dSNR} \approx L\frac{0.675}{\sqrt{\pi}} SNR^{-\frac{1}{2}} e^{-}((0.45)^2 SNR)) \approx 0 \tag{9.15}$$

where L is length of the packet;

This approximation is compared to the analytical model in Fig. 9.20. The results clearly indicate that the approximate equation captures the change of SNR as the vehicle approaches the RSU. We then use the approximate equation to compare the results for different packet lengths of 1556, 2856, 5456 bits (about 200, 325 and 752 bytes). The results are shown in Fig. 9.21.

The graph in Fig. 9.21 shows that the length of beacon does affect the rate of change of SNR but these values converge as the Packet OK approaches 1. However, when we plot $dP/dSNR$, approximate the rate of change of $dP/dSNR$ vs. Packet Length in the range being considered, we get a straight line that indicates that there is a linear relationship between the SNR and the length of the beacon. This line can be represented using a simple line equation (Eq. 9.16)

$$y = mx + b \tag{9.16}$$

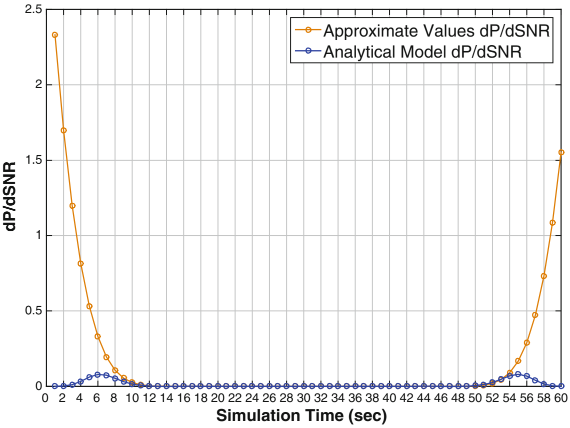


Fig. 9.20 Comparison of analytical model versus Approximation

Fig. 9.21 Approximate with different packet lengths

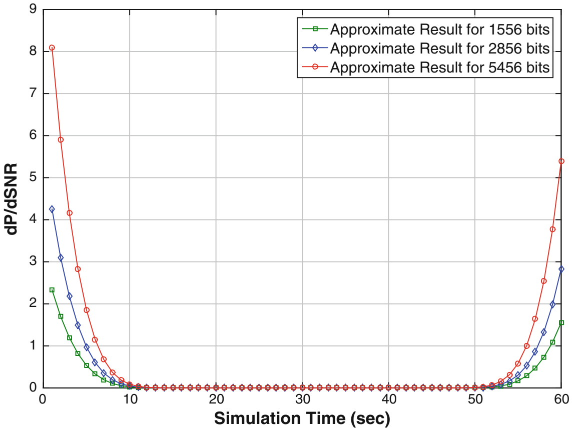
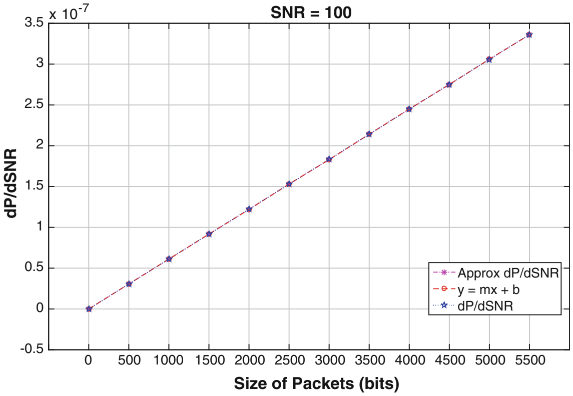


Fig. 9.22 Rate of change versus packet length



where, x and y are the coordinates of the line, m is the slope of the line and b is the y intercept.

Equation 9.16 can be represented in terms of length of beacon and $dP/dSNR$ as shown in Eq. 9.17.

$$dP/dSNR = mL + b \tag{9.17}$$

Here, L is the length of the beacon, m = 6.1172E-11 per bit and b = −1.00E-13 for SNR = 100.

The result in Fig. 9.22 where SNR = 100 shows that the $dP/dSNR$ of the approximation results are very close to the real results and hence the values of m, which is the change of $dP/dSNR$ per bit length of the beacon is an useful value to estimate the change of $dP/dSNR$ as the vehicle approaches the RSU.

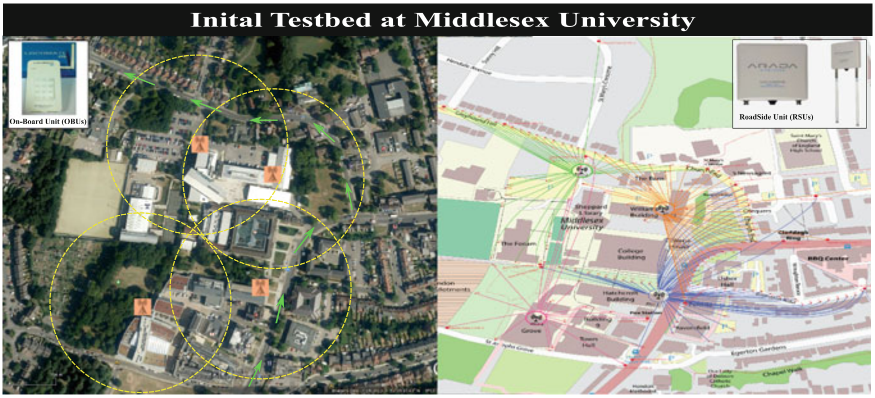


Fig. 9.23 MDX VANET Testbed

9.7 TestBed

In order to further explore the analytical and simulation models, a prototype VANET Testbed is being deployed at the Hendon Campus of Middlesex University as depicted in Fig. 9.23. This testbed will enable us to evaluate correctly these models thus giving us better insight into the deployment of real transport networks. In the long term, we are seeking to develop a comprehensive framework that includes types of modulation being used as well as traffic density in order to handle seamless handover in both urban and motorway contexts.

9.8 Summary

This chapter has looked at providing seamless communication in mobile environments in order to provide Dependable Vehicular Communication for Improved Road Safety. Results from simulation and analytical models have been presented, and a prototype VANET Tested is being deployed to further explore these issues.

References

1. M. Almulla et al., Design of a fast location-based handoff scheme for IEEE 802.11 vehicular networks. IEEE Trans. Veh. Technol. **63**(8), 3853–3866 (2014). ISSN: 0018-9545, doi:10.1109/TVT.2014.2309677
2. M. Augusto et al., MYHand: a novel architecture for improving handovers in NGNs (2014). http://www.thinkmind.org/index.php?view=article&articleid=aict_2013_9_40_10181

3. S. Bi et al., Proper handover between VANET and cellular network improves internet access, in *Vehicular Technology Conference (VTC Fall), 2014 IEEE 80th* September 2014, pp. 1–5. doi:10.1109/VTCFall.2014.6966137
4. P. Fuxjager et al., IEEE 802.11 p transmission using GNURadio, in *Proceedings of the IEEE 6th Karlsruhe Workshop on Software Radios (WSR)* (2010), pp. 83–86. http://userver.ftw.at/~valerio/files/wsr10.pdf
5. C. Ganan et al., Analysis of inter-RSU beaconing interference in VANETs, in *Multiple Access Communications*, Lecture Notes in Computer Science, ed. by B. Bellalta, et al. (Springer, Berlin, 2012), pp. 49–59. ISBN: 978-3-642-34975-1, http://dx.doi.org/10.1007/978-3-642-34976-8
6. A. Ghosh et al., Exploring efficient seamless handover in VANET systems using network dwell time. EURASIP J. Wirel. Commun. Netw. **2014**(1), 227 (2014). ISSN: 1687-1499, doi:10.1186/1687-1499-2014-227
7. A. Ghosh et al., Providing ubiquitous communication using handover techniques in VANET systems, in *Ad Hoc Networking Workshop (MED-HOCNET), 2014 13th Annual Mediterranean* June 2014, pp. 195–202. doi:10.1109/MedHocNet.2014.6849124
8. A. Ghosh et al., Providing ubiquitous communication using road-side units in VANET systems: Unveiling the challenges, in *13th International Conference on ITS Telecommunications (ITST), 2013* November 2013, pp. 74–79. doi:10.1109/ITST.2013.6685524
9. IEEE-Std, IEEE Draft Standard for Wireless Access in Vehicular Environments (WAVE)—Multi-Channel Operation (2010). http://ieeexplore.ieee.org/servlet/opac?punumber=5511462
10. IEEE-Std, IEEE Standard for Information technology-Local and metropolitan area networks-Specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications—Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements (2005). doi:10.1109/IEEESTD.2005.97890, http://dx.doi.org/10.1109/IEEESTD.2005.97890
11. G. Mapp et al., Exploiting location and contextual information to develop a comprehensive framework for proactive handover in heterogeneous environments. J. Comput. Netw. Commun. **2012** Article ID 748163, pp. 1–17 (2012). doi:10.1155/2012/748163
12. G. Mapp et al., Exploring efficient imperative handover mechanisms for heterogeneous wireless networks, in *International Conference on Network-Based Information Systems, 2009. NBIS '09* August 2009, pp. 286–291. doi:10.1109/NBiS.2009.95
13. G.E. Mapp et al., Y-Comm: a global architecture for heterogeneous networking, in *Proceedings of the 3rd International Conference on Wireless Internet. WICON '07*. Austin, Texas, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering) (2007) pp. 22:1–22:5. ISBN: 978-963-9799-12-7, http://dl.acm.org/citation.cfm?id=1460047.1460075
14. K. Sjoberg et al., Measuring and using the RSSI of IEEE 802.11p. eng. Busan, South Korea (2010)
15. C. Sommer, VEINS: vehicles in network simulation (2014). http://veins.car2x.org/
16. C. Sommer, R. German, F. Dressler, Bidirectionally coupled network and road traffic simulation for improved IVC analysis. IEEE Trans. Mob. Comput. **10**(1), 3–15 (2011). ISSN: 1536-1233, doi:10.1109/TMC.2010.133
17. A. Varga, OMNeT++: an extensible, modular, component-based C++ network simulation (2014). http://www.omnetpp.org/
18. A. Vinel et al., Estimation of a successful beacon reception probability in vehicular Ad-hoc networks, in *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*. IWCMC '09. (ACM, Leipzig, Germany, 2009) pp. 416–420. ISBN: 978-1-60558-569-7, doi:10.1145/1582379.1582470, http://doi.acm.org/10.1145/1582379.1582470
19. A. Vinel, D. Staehle, A. Turlikov, Study of beaconing for Car-to-Car communication in vehicular Ad-hoc networks, in *IEEE International Conference on Communications Workshops, 2009. ICC Workshops 2009* June 2009, pp. 1–5. doi:10.1109/ICCW.2009.5208066

20. J. Wang, Q.-A. Zeng, D.P. Agrawal, Performance analysis of a preemptive and priority reservation handoff scheme for integrated servicebased wireless mobile networks. IEEE Trans. Mob. Comput. **2**(1), 65–75 (2003). ISSN: 1536-1233, doi:10.1109/TMC.2003.1195152

Chapter 10
Modeling Vehicles Mobility for Connectivity Analysis in VANET

Tariq Umer, Muhammad Amjad, Nadir Shah and Zhiguo Ding

Abstract The availability of more realistic road conditions and dynamics provides sound ground to study the issues of Vehicular ad-hoc Network (VANET). In this chapter a new heterogeneous traffic flow based mathematical model is presented, to gain the time and space dynamics of vehicles. To achieve more accurate and realistic data about road conditions, microscopic parameters of varying safety distance between the vehicles and vehicular length are considered in the model. The density dynamics under different road scenarios are calculated under the influence of these constraints with the use of a defined mathematical model. The model is able to capture the impact of road constraints such as traffic lights and road incidents, on the traffic flow. The concept of Vehicular Ad-hoc Networks (VANET) has given mankind opportunities for secure and safe journeys on the roads. VANET is defined as a subclass of Mobile Ad-hoc Networks which holds the characteristics of ad-hoc networks. However due to the dynamic road conditions, traffic flow theory concepts, mobility constraints, human behaviours and vehicular characteristics VANET exhibits different dynamics. These factors have strong influences on the VANET architecture from physical to application layers. This highlights different areas of interest in VANET for researchers to investigate. This study aims to capture the impact of traffic flow theory constraints on the vehicular density under the heterogeneous traffic flow on the road. The microscopic and macroscopic characteristics of vehicles moving on the roads are utilized for the improvement of VANET connectivity dynamics.

T. Umer (✉) · M. Amjad · N. Shah
Comsats Institute of Information Technology, Wah Cantonment, Pakistan
e-mail: t_umer@yahoo.com

M. Amjad
e-mail: amjdbhutta0706@gmail.com

N. Shah
e-mail: nadirshah82@gmail.com

Z. Ding
School of Computing and Communications, Lancaster University, Lancaster, UK
e-mail: Z.ding@lancaster.ac.uk

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_10

10.1 Impact of Vehicle Mobility Pattern on VANET

In the area of vehicular research, the real life macroscopic and microscopic characteristics of vehicles are key issues to focus and analyse. The dynamic behaviour of vehicles on the road due to the road conditions, individual behaviour and surrounding environment affect its mobility pattern. In the vehicular ad-hoc network the timely distribution of information between the vehicles is the key issue. The moving patterns of vehicles impact the network communication as these structures define different road scenarios due to the implementation of traffic flow theory laws.The researchers have modelled the vehicle mobility in VANET by using the concept of different fields of science, engineering and mathematics. These mobility models use important road constraints and incorporate vehicular characteristics and traffic laws to define more realistic road conditions for the VANET. In [6] the influence of mobility patterns of vehicles on the performance of routing protocol in VANET is focused. They implemented the vehicle movement simulation tool SUMO [5] under NS2 simulator to capture more realistic traffic movements. They have presented the impact of traffic constraints such as density, speed and road structure on the performance of VANET. Considering the mobility of vehicles as a key factor to study the connectivity and evaluation of VANET performance [8] presented a concept of equivalent speed between the vehicles moving on the road. The study derived an analytical expression to relate the vehicular speed with connectivity under different mobility patterns and road scenarios.

The connectivity in vehicular ad-hoc network between different nodes has considerable effect under different road infrastructures and vehicle movement plans. The changing network topology of these kinds of networks is another factor which gives different values of connectivity. In real life vehicles are moving on different kinds of roads. For the vehicles moving on a highway, the researchers consider one way and two way moving patterns of vehicles. The vehicles changing speed, interaction with other arriving and departing vehicles are considered for connectivity dynamics.

The vehicles arriving rates are considered Poisson and Exponential in different research studies. The effect of neighbouring vehicles on the moving vehicle pattern is also considered for connectivity dynamics. In [7] the connectivity dynamics under different mobility models such as random way point and Manhattan mobility model are presented. The comparisons of important connectivity related constrains, under these models are presented by using simulated results for VANETs in an urban environment. When vehicles are moving in an urban road environment the microscopic constraints of traffic flow effect the vehicles mobility pattern.

10.2 Considering Vehicular Density for Analysing Connectivity in VANET

As the vehicles movement is effected by the road incidents and infrastructure, this dynamic change in vehicles conditions create a considerable impact on the vehicular density. The distribution of vehicles on the road needs to be considered as a random variable [19]. The vehicular density can be analytically observed for its effect on the connectivity. By utilizing the density dynamics of vehicles in [17] a clustering algorithm is implemented to improve the VANET performance. It provides a stable connectivity on the basis of existing larger groups of vehicles moving together known as a density based clustering algorithm.

The vehicular density depends on the changing road conditions. It goes up and down under the effect of traffic flow conditions. These characteristics of vehicular density provide an area of interest. To improve transmission capabilities of nodes in VANET for enhanced connectivity conditions the study [2] suggested the assignment of a dynamic transmission range for the nodes, on the basis of local density estimation defined by a technique called Dynamic Transmission Range Assignment (DTRA) algorithm. This algorithm considers the local traffic conditions as well as considering the density dynamics of the surrounding environment of a node, for the assignment of optimal transmission power to the nodes. The analysis of message propagation on the basis of asymmetric densities in VANET is presented by [1]. The study focuses the short range transmission technologies and local density on the opposite directions for improving connectivity.

The research considered the different road scenarios for finding density under dynamic road conditions. When the vehicles reached their max value of density they have the effect of jamming density on their speed. In the case of vexhicles moving with free flow the density value is significantly less which influences the connectivity. In real life road environments, there are different types of vehicles moving on the road. The length and speed variation of these vehicles create dynamic values of density. The platoon formation of vehicles on the road signals, turning points, incident points and junctions gives different dynamics of connectivity for analysis.

10.3 Implementation of Microscopic Parameter for Density Estimation of Heterogeneous Traffic Flow for VANET

When thinking about the safety of the passengers and safe journeys, the traffic flow needs to be governed by traffic codes on the roads. These traffic flow rules and regulations are defined by the traffic control authorities [24]. Researchers have introduced traffic flow models for analyzing the road conditions and vehicular behavior. The main focus was to achieve a maximum realistic traffic environment to carry out analytical and simulated data analysis for VANET. The microscopic and macroscopic parameters have been considered in many research studies to get realistic road conditions [14]. The traffic engineers define different relationships between these important parameters of traffic flow to present a new traffic flow model. These models consider linear relationship between speed and density, shockwaves effect due to different densities on streams of same flow, relating flow of traffic with fluid characteristics and the use of kinetic theory concept to define and relate two different flow regimes [9].

It has been observed that due to smaller safety distances between vehicles, accidents happen during the sudden stoppage of the front vehicle as a result of some road constraint. To avoid such situations, transport authorities have observed the law of safety distance between the vehicles very seriously. Several research studies have thus focused on this issue for obtaining secure traffic conditions. The authors in [10] worked on a project 'SASPENCE—Safe Speed and Safe Distance' to provide environment and, technology for the drivers to make their journeys safer by implementing a safe speed and safe distance concept.

The designing and implementation of vehicular ad-hoc network depends on the prevailing road conditions. The density dynamics are used to define transmission range and conditions for stable network connectivity. The use of density dynamics for the assignment of dynamic transmission range in vehicular ad-hoc network has been performed in these studies [3, 22].

The mobility of vehicles on the road is also considered to enhance the performance of vehicular ad-hoc network. The available models of traffic flow theory such as the fluid dynamic model and the car-following model have been used for representing traffic flow in many studies for vehicular ad-hoc network [4, 13]. Analytical modeling of vehicular ad-hoc network based on traffic flow models and density dynamics have been also reported [16, 26].

In this chapter the density of moving vehicles is achieved for different road scenarios and structures such as highway and the signalized urban road. The vehicular density is calculated with the use of fluid dynamic model having the effect of road conditions. A safety distance characteristic of the traffic flow has been implemented in a heterogeneous traffic environment using the Car-following model. Highway Code for safety distance between vehicles is introduced in the jamming density so that impact of jamming density on the velocity and density of vehicles can be analyzed. The implementation of these parameters provides more realistic traffic flow and road conditions.

10.3.1 Calculating Jamming Density for Heterogeneous Traffic Flow

It is evident from the studies of traffic flow theory that the microscopic parameters such as headway, gap and occupancy have an impact on the flow of traffic stream [11, 20]. In a real life scenario, traffic stream consists of different types of vehicles. The vehicles can be categorised as light traffic vehicles (LTV) such as cars and heavy traffic vehicle (HTV) such as buses and trucks. These vehicles have different structural and operating characteristics and exhibit different behaviour under the dynamic traffic flow conditions. Due to this heterogeneous nature of the leading vehicles, the following vehicle has to maintain specific safety distances from the front vehicle in accordance with the traffic safety laws. In a heterogeneous traffic flow, the constraints of headway and gap for the following vehicle become dependable on the characteristics of the leading vehicle. To observe the effect of heterogeneousness of vehicles structure on the dynamics of vehicular density, the traffic stream of two major types of vehicles i.e. cars and buses are considered. The vehicle motion is modelled with velocity profiles, having effects of leading vehicle conditions in term of density, length, and space headway. Vehicles of any type on the road at location 'x' and time 't' move forward with velocity $u_i(x,t)$', where $i = type$ of vehicle available on the road. This velocity can be deterministic or dependent on the front density and jam density. According to the Greenshield [18] in a car following environment, the speed, density and flow relation is given as in Eq. 10.1.

$$U\,(x,t) = u_f\left(1 - \frac{n\,(x,t)}{k_{jam}}\right) \tag{10.1}$$

where u_f is the free speed of vehicle, k_{jam} is the jam density, $n\,(x,t)$ is the vehicular density.

Introducing the safety distance of vehicle from the leading vehicle in heterogeneous traffic flow environment the jam density is given as in Eq. 10.2

$$K_{jam} = \frac{1}{L+h} \tag{10.2}$$

where L defines the length of a vehicle and h defines the safety distance of the specific vehicle from the leading vehicle. In a heterogeneous traffic environment due to the variable length of vehicles the space between the vehicles defined as headway/safety distance for following vehicle from the leading vehicle becomes a random variable. Thus $a - type$ of vehicles safety distance depends on the probability of other type of front vehicle and its characteristics. Let the h_{ab} is the safety distance for a-type of vehicle when the front vehicle is $b - type$. This h_{ab} satisfy that $\left(h_{ax(i)} > h_{bx(i+1)}u_{x(i)f} > u_{x(i+1)f}\right)$. The parameter P_b is the probability that the vehicle exists in front of a-type vehicle is of b-type. This state that the a-type safety distance is characterized as random variable and the safety distance h_{ab} will be given mean of the random variable h_{ab} i.e.

$$h_a = \sum_{b-1}^{N} p_b h_{ab} \tag{10.3}$$

For the jam density k_{jam} under random characteristic of length and safety distance the variables for $i - type$ vehicles, will be given as mean variables, so the variable L and h is given as

$$L = \sum_{i=1}^{N} p_i L_i \tag{10.4}$$

$$h = \sum_{i=1}^{N} p_i h_i \tag{10.5}$$

So the ja density for heterogeneous traffic flow can be define as

$$K_{jam} = \frac{1}{\sum_{i=1}^{N} p_i (L_i + h_i)} \tag{10.6}$$

By substituting the value of Jam density in Eq. (3.1)

$$U\,(x, t) = u_f \left(1 - \frac{n\,(x, t)}{\sum_{i=1}^{N} p_i (L_i + h_i)} \right) \tag{10.7}$$

This expression gives the speed of a vehicle by using the car following model which has the influence of length and safety distance of leading vehicles in heterogeneous traffic environment [23].

10.3.2 Using Deterministic Fluid Dynamic Model for Density Estimation of Heterogeneous Traffic Flow

Road traffic is characterized as fluid flow by the traffic engineers. This fluid like behaviour of moving vehicles on the road leads to represent different traffic models called continuum traffic flow models having characteristics of fluid dynamics. The fluid dynamic model represents traffic flow in the form of conservation law. It provides the traffic flow and density as a function of time and space. It relates the behaviour of traffic in the form of partial differential to represent parameters like flow, speed and density. The model presented in [25], we have considered heterogeneous traffic flow in a single-lane, one way, and semi-infinite highway environment. The location space is characterized with the interval $[0, \infty]$ the starting of the road is marked by boundary point 0, which is considered as spatial origin. The road is divided into a number of road segments represented by $r = 1, 2, 3, 4$ due to road intersections. The vehicles can join or leave the moving stream at these intersections called junctions. For the first segment of the road, the number of arrivals of all types of vehicles up to time t, is counted by an arrival process $(G(t)| - \infty < t < \infty)$, which is assumed to be finite with probability 1. This arrival process is characterized by external arrival rate function for all types of vehicles $\lambda(t)'$ which is non-negative and can be integrated. I have considered two types of vehicles that are car and bus. So the arrival rate for cars and buses is given by an external arrival rate function $\lambda_c(t)$ and $\lambda_b(t)$. The conservation equation relating important parameters of traffic flow is given as:

$$E^{+}(x, t) = N(x, t) + F(x, t) + E^{-}(x, t) \tag{10.8}$$

$N(x,t)$ as Total number of vehicles in location $(0,x)$, $F(x,t)$ Number of vehicles passing past position x, Whereas $E^{+}(x,t)$, $E^{-}(x,t)$ are vehicles arriving and departing rate. Implementing this conservation equation for cars and buses

$$E_c^{+}(x,t) = N_c(x,t) + F_c(x,t) + E_c^{-}(x,t) \tag{10.9a}$$

$$E_b^{+}(x,t) = N_b(x,t) + F_b(x,t) + E_b^{-}(x,t) \tag{10.9b}$$

For having partial differential equation form of conservation equation relating density, flow, arrival and leaving rate. We are differentiating Eqs. 10.9a and 10.9b by time and space [15]. Using operator on the Eqs. 10.9a and 10.9b.

$$\frac{\partial n_c(x,t)}{\partial t} + \frac{\partial f_c(x,t)}{\partial x} = e_c^{+}(x,t) - e_c^{-}(x,t) \tag{10.10a}$$

$$\frac{\partial n_b(x,t)}{\partial t} + \frac{\partial f_b(x,t)}{\partial x} = e_b^{+}(x,t) - e_b^{-}(x,t) \tag{10.10b}$$

According to the fundamental relation of traffic flow theory

$$f(x,t) = n(x,t) \times u(x,t) \tag{10.11}$$

Using Eq. 10.10 we have

$$\frac{\partial n_c(x,t)}{\partial t} + \frac{\partial [n_c(x,t) \times u_c(x,t)]}{\partial x} = e_c^{+}(x,t) - e_c^{-}(x,t) \tag{10.12a}$$

$$\frac{\partial n_b(x,t)}{\partial t} + \frac{\partial [n_b(x,t) \times u_b(x,t)]}{\partial x} = e_b^{+}(x,t) - e_b^{-}(x,t) \tag{10.12b}$$

These relations formed the one dimensional version of generalized conservation law for fluid motion in partial differential form representing cars and buses in heterogeneous traffic flow. By applying a chain rule and defining velocity as

$$u(x(t),t) = \frac{dx(t)}{dt} \tag{10.13}$$

After substituting the values from Eq. 3.8 we get the equation for finding density of two different type of vehicle.

$$\frac{dn_c(x(t),t)}{dt} = e_c^{+}(x,t) - e_c^{-}(x,t) - \frac{\partial u_c(x,t)}{\partial x} n_c(x(t),t) \tag{10.14a}$$

$$\frac{dn_b(x(t),t)}{dt} = e_b^{+}(x,t) - e_b^{-}(x,t) - \frac{\partial u_b(x,t)}{\partial x} n_b(x(t),t) \tag{10.14b}$$

$$N(x,t) = n_c(x(t),t) - n_b(x(t),t) \tag{10.15}$$

The Eqs. 10.14a and 10.14b have the effect of microscopic variables of headway and safety distance for different types of vehicles in the traffic stream and can be applied for finding total vehicular density with the use of Eq. 10.15 having effect of dynamic road conditions [21].

We have introduced a velocity profile from a car following the model defined in Eq. 10.1 in Eqs. 10.10a and 10.10b for density estimation using fluid dynamic model. The algorithm for simulation is run iteratively for getting speed on current time for all locations. The differential equation from fluid dynamic model is then solved for density by using the velocity data at time 't' for all locations 'x'. The effect of heterogeneousness in fluid dynamic model is introduced by using the new value of jamming density in Eq. 10.7 and giving speed of vehicle (i) in heterogeneous environment.

10.4 Numerical Analysis for Highway Traffic Flow

We assume that no vehicle is joining or leaving the highway at junctions so that the continuous flow of vehicles can be achieved on the road. The vehicles arrive only at location 0 at a constant arrival rate $\lambda(t) = 50$ vehicles/min. The constant arrival of 50 vehicles/min creates enough traffic streams to gain vehicular density. This arrival rate is further divided in arrival ratio of two types of vehicles, cars and buses. The arrival rate for cars and buses is defined as $\lambda_c(t)$ and $\lambda_b(t)$ having different arriving ratios for heterogeneous arrival. The arrival rate for a car and a bus is changed for a different simulation run to create scenarios to implement the different traffic flow conditions. We consider that initially there is no vehicle on the road when traffic starts. So $n_c(x, 0) = 0$ and $n_b(x, 0) = 0$ for all 'x' belongs to 'X' where 'X' is location space in km. The initial velocity for all vehicles as calculated from Eq. 10.1 at $(t = 0)$ will be the mean free speed $u_f = 1$ km/min.

- Finding Jamming density for car only case.

The jamming density is calculated for car only case by using the two different formulas. (i) $(kjam = 1/lc)$ which only considered the length of vehicle so for car only case $lc = 4$ m. (ii) $(kjam = 1/lc + hc)$ having safety distance between the vehicles and length of considered vehicle. For car only case, lc = 4 m and hc = 4 m and 12 m for two different road scenario. I introduced a traffic constraint on the highway between location 3–7 km to capture the effect of jam density and velocity change on vehicular density. The impact on vehicular density between these two locations is later presented at different time intervals. The velocity field $U_i(x, t)$ is calculated from Eq. 10.1 for cars only case and from Eq. 10.7 for heterogeneous case for all $x \geq 0$ when $t \leq 30$ or $t > 45$ min. For $30 < t \leq 45$ min, the velocity field is calculated as:

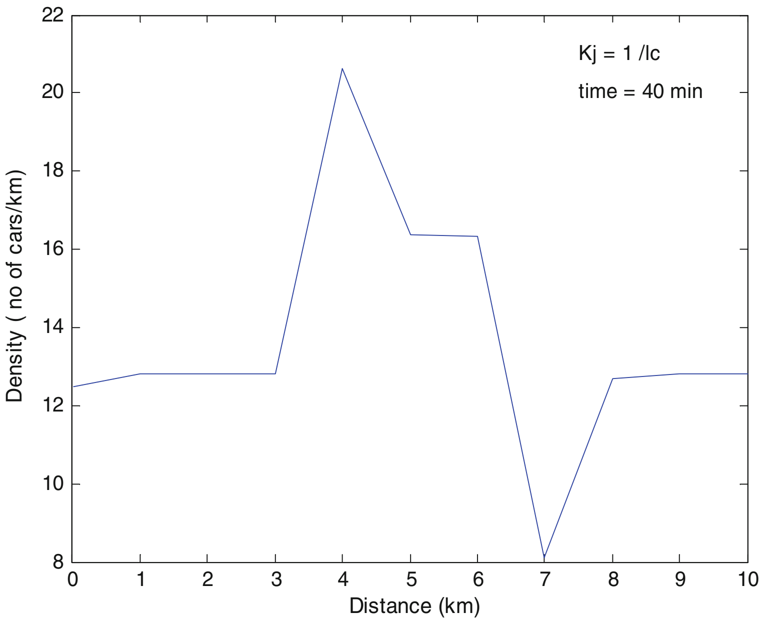


Fig. 10.1 The Vehicular Density for car only case with $K_j = 1/lc$

$$U_i(x,t) = \begin{cases} U_i(x,t) & \text{if } x \leq 3 \\ U_i(x,t) - (\frac{U_i(x,t)}{2}) & \text{if } 3 < x \leq 4 \\ \frac{U_i(x,t)}{2} & \text{if } 4 < x \leq 6 \\ U_i(x,t) + (\frac{U_i(x,t)}{2})(x-6) & \text{if } 6 < x \leq 7 \\ U_i(x,t) & \text{if } x > 7 \end{cases} \tag{10.16}$$

The Fig. 10.1 shows the density for cars on the highway at time 40 min. The density dynamics of vehicles show dynamic behaviour between locations 3–7 km due to a sudden constraint at this location. The change in velocity during this interval affects the vehicular density. As the vehicles move on, the density starts building up.

The vehicular density falls down due to the sudden stop of traffic flow and goes back to constant rate after certain distance. In Fig. 10.2 with the introduction of safety headway between the cars, the jam density is calculated under car length and safety distance constraints. The increase in safety distance affects the velocity as indicated in Fig. 10.2. The velocity graph is achieved for two different safety distances between the cars i.e. 4, 12 m. The variable safety distances between the cars define two different traffic flow conditions. The impact of road constraints on the velocity is expressed with the decrease of velocity profile. As a result of constraint on the road, the velocity decreases and density changes sharply at location 4 m.

The increase in safety distance provides smooth flow of traffic and shows increased density dynamics less than 12 m safety distance case as compared to 4 m case, reflected in Fig. 10.3. The vehicular density shows dynamic behaviour between the distances 3–7 km under the effect of road constraint.

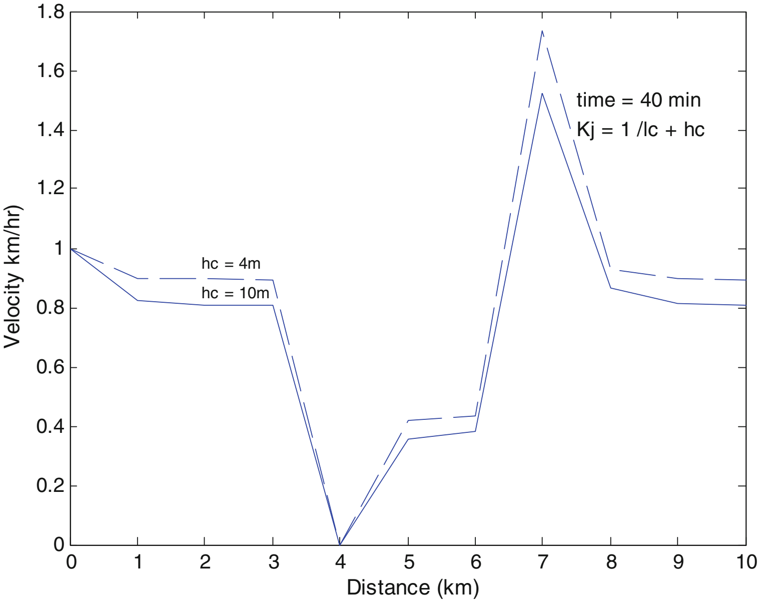


Fig. 10.2 The Vehicle Velocity for car only case with jamming density defines as $K_j = 1/(lc + hc)$

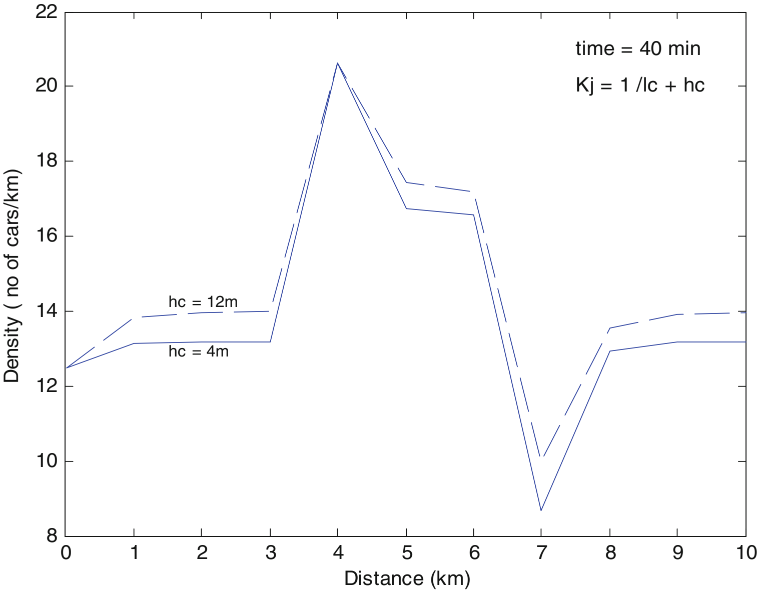


Fig. 10.3 The Vehicle Density for car only case with jamming density defines as $K_j = 1/(lc + hc)$

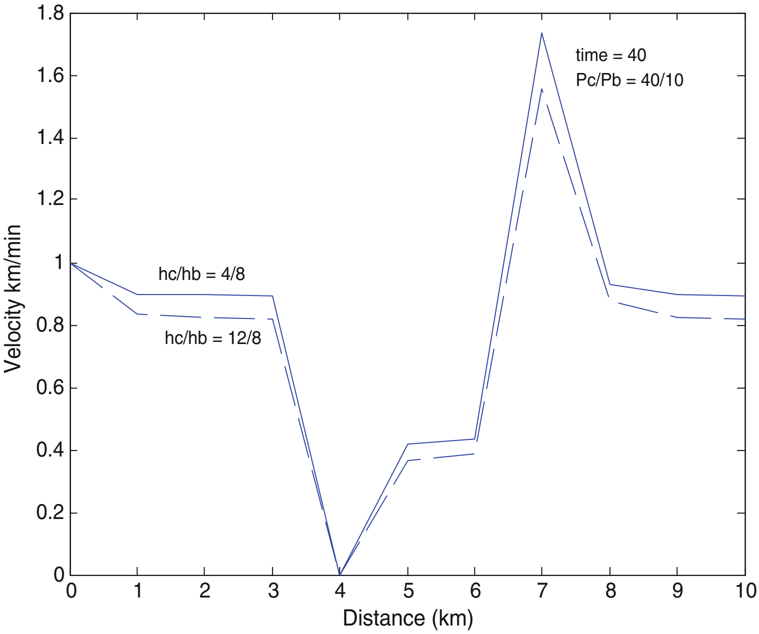


Fig. 10.4 The Vehicle Velocity for cars and buses case and buses case having effect of safety distance in K_j with cars in excess ratio

- Finding Jamming density for heterogeneous traffic flow case

The jam density for heterogeneous traffic flow considering cars and buses is calculated by the Eq. 10.7. This equation has influence of length, safety distance and varying arrival ratio for cars and buses. The external arrival rate for car $\lambda_c(t)$ and bus $\lambda_b(t)$ is considered as $a = \lambda_c(t)/\lambda_b(t) = 40/10$ for Figs. 10.4 and 10.5 and $b = \lambda_c(t)/\lambda_b(t) = 10/40$ for Figs. 10.6 and 10.7. The two different arriving ratios of cars and buses and the safety distance ratio between the car and bus 'hc/hb' are considered to analyse vehicular density dynamics under my mobility model for different road and traffic flow situations. In Figs. 10.4 and 10.5 the $h_c/h_b = 04/08$ and $h_c/h_b = 12/08$ defines the changing safety distance between cars and buses, having 04 and 12 m safety distance between cars whereas keeping the bus safety distance at 08m constant as buses are following the same moving pattern on the road. For the Figs. 10.6 and 10.7 to observe the effect of constant safety distance between the cars on the traffic flow the safety distance between buses is changed. The ratio for car and bus safety distance is defined as $h_c/hb = 08/16$ and $h_c/h_b = 08/04$ having fixed car safety distance and changing safety distance for buses such as 16 and 4 m. In both cases the car length 'lc' is fixed as 4 m and bus length 'lb' is kept at 10 m. The effect of these changes in arriving and safety distance ratios on vehicular velocity and density is shown in Figs. 10.4, 10.5, 10.6 and 10.7. The safety distance varies for different types of vehicles on the road, therefore the ratio 'hc/hb' are varied to

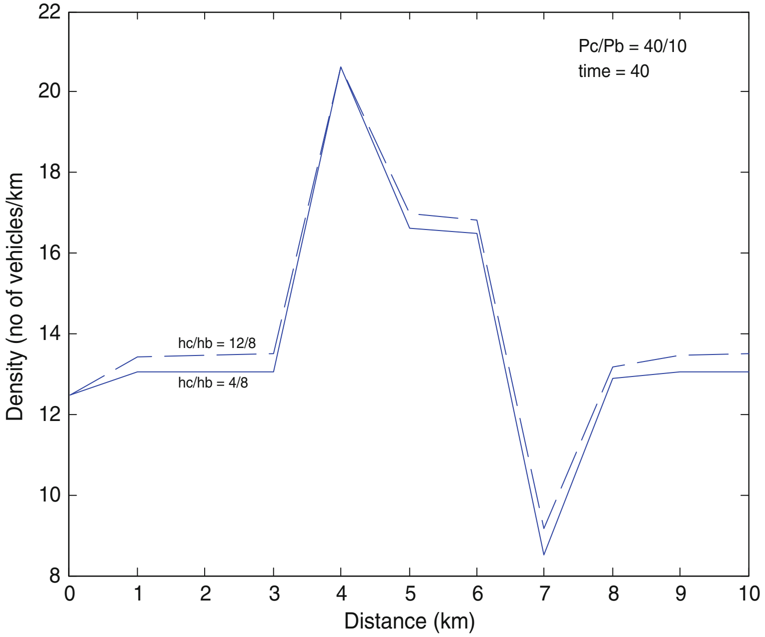


Fig. 10.5 The Vehicle Density for cars and buses case and buses case having effect of safety distance in K_j with cars in excess ratio

get the effect of changing road conditions on vehicular density. These results shows that at time 40 min velocity starts decreasing between locations 3–7 km due to a constraint on the highway in all the cases. The flow conditions under excess number of small vehicles with optimal safety distance provide better results as compared to long vehicles. The arriving ratio and safety distance between vehicles affects the vehicular density. The impact of a microscopic parameter in both the car only, and the car and bus case is captured. The traffic conditions can be manipulated with the proper use of microscopic parameter of safety distance.

10.4.1 Introducing Impact of Leading Traffic Flow in Velocity

In a heterogeneous traffic environment due to the different types of vehicles and their characteristics, the traffic flow exhibits dynamic behaviour. The leading vehicles impact the traffic pattern, which influence the vehicular density on the road. To observe the effect of leading traffic conditions the concept of front density profile is introduced in the previous study [12]. To achieve a realistic traffic condition and density profile, in our work I have introduced the effect of heterogeneous traffic

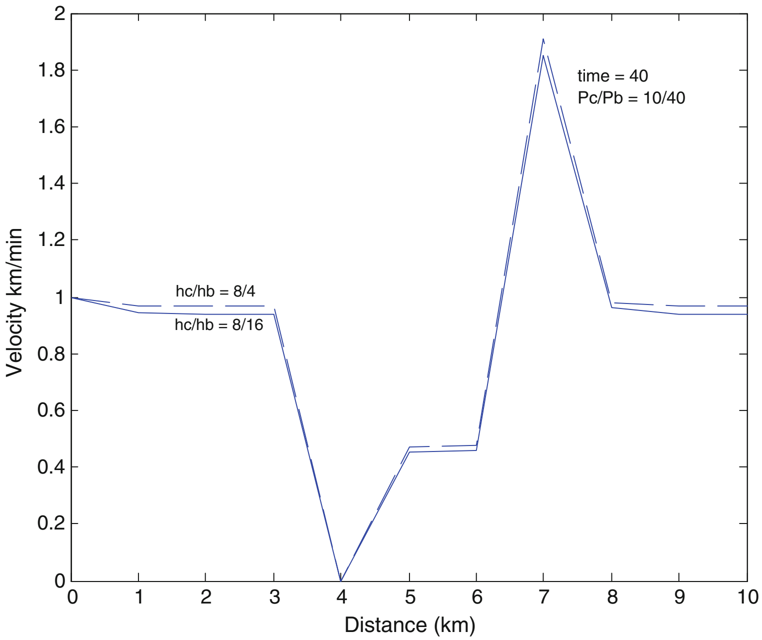


Fig. 10.6 The Vehicle Velocity for cars and buses case and buses case having effect of safety distance in K_j with cars in excess ratio

environment in the front density. As the front density is increased due to traffic signal implementation, the density dynamics of the road for that region is also affected. By using Eq. 10.7 and introducing front density as Δx the velocity equation under front density will be given as

$$U_i(x,t) = u_f[\frac{(1-N_T)(x+\Delta x,t)}{\sum_{i=1}^{N} p(L_i+h_i)}] \tag{10.17}$$

10.4.2 Density Estimation for Heterogeneous Traffic under different Safety Conditions in Signalized Road Structure

In this model we considered heterogeneous traffic flow in a single-lane, one way, semi-infinite signalized road in an inner-city environment. The road is divided into the number of road segments represented by $r = (1, 2, 3, 4. \ldots)$ controlled by traffic lights installed at the point of intersection. We assume that no vehicle is joining or leaving the road at intersections. The vehicles arrive only at location 0 at a constant rate $\lambda(t) = 20$ vehicles/min this includes all type of vehicles. For the heterogeneous

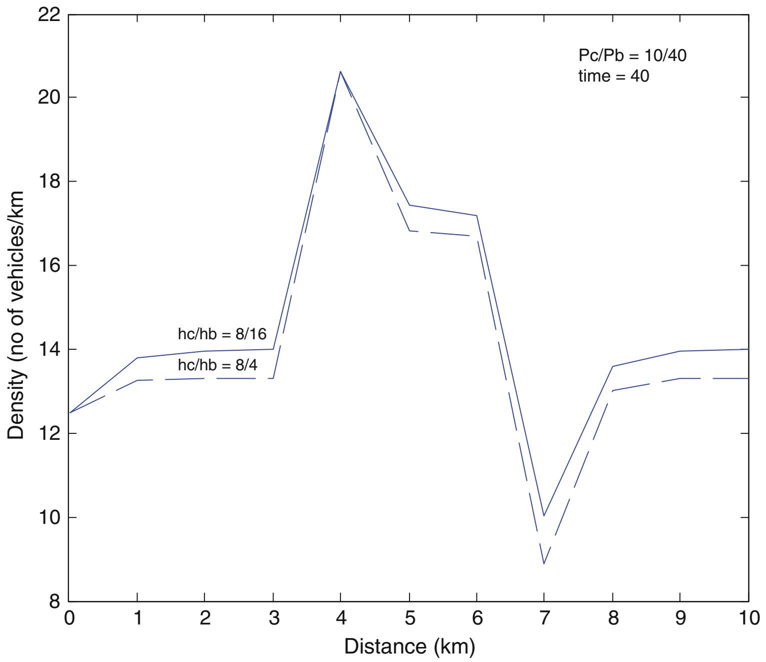


Fig. 10.7 The Vehicle Density for cars and buses case and buses case having effect of safety distance in K_j with cars in excess ratio

traffic flow the arrival rate for cars and buses is defined as $\lambda_c(t)$' and $\lambda_b(t)$ having different arriving ratios for different road scenarios. We consider that initially there are no vehicles on the road when traffic starts. So $n_c(x, 0) = 0$ and $n_b(x, 0) = 0$ for all x belongs to X where X is location space in km. The initial velocity for all vehicles as calculated from Eq. 10.1 at $(t = 0)$ will be the mean free speed $V_f = 1$ km/min. On the road at the distance of 4 km, we have introduced a traffic light to capture the effect of vehicles interaction due to the safety distance and front density of the road traffic on vehicular density. The velocity field $U_i(x, t)$ is calculated from Eq. 10.1 for cars only case and from Eq. 10.17 for heterogeneous case under front density profile. During the red traffic light period, traffic is stopped for 4–4.5 min for 30 s. For the implementation of a road junction, an extra 0.012 Km distance is considered before the traffic light. During the stopping period, the velocity profile for road traffic is calculated as:

$$U_i(x,t) = \begin{cases} U_i(x,t) & \text{if} \quad x \le 3.98 \\ (\frac{U_i(x,t)}{0.02})(4-x) & \text{if} \quad 3.98 < x \le 4 \\ 0 & \text{if} \quad 4 < x \le 4.012 \\ (\frac{U_i(x,t)}{0.02})(x-4.032) & \text{if} \quad 4.012 < x \le 4.032 \\ U_i(x,t) & \text{if} \quad x > 4.032 \end{cases} \tag{10.18}$$

The traffic condition on the road is dynamic due to the changing arriving rates for different types of vehicles. Due to these arriving patterns the safety distance between the vehicles is also affected. In Figs. 10.8, 10.9 and 10.10 to capture the effect of heterogeneousness traffic flow on density for cars and buses case I have assumed 20 vehicles such as the external arrival rate for two types of vehicle car and bus is given as: Car $(\lambda_c(t)) = 12/20$ and bus $(\lambda_b(t)) = 08/20$ The different arriving ratio for car and bus creates an impact of different vehicular characteristics due to their types on the vehicular density. For the three different road scenarios under the influence of increasing safety distance between the cars, we have considered three different safety distances between the cars (sfdc) such as

- 2 × car-length for case (a) in Fig. 10.8
- 3 × car-length for case (b) in Fig. 10.9
- 4 × car-length for case (c) in Fig. 10.10
- The safety distance for buses is fixed as 0.012 Km as they are following the same pattern.

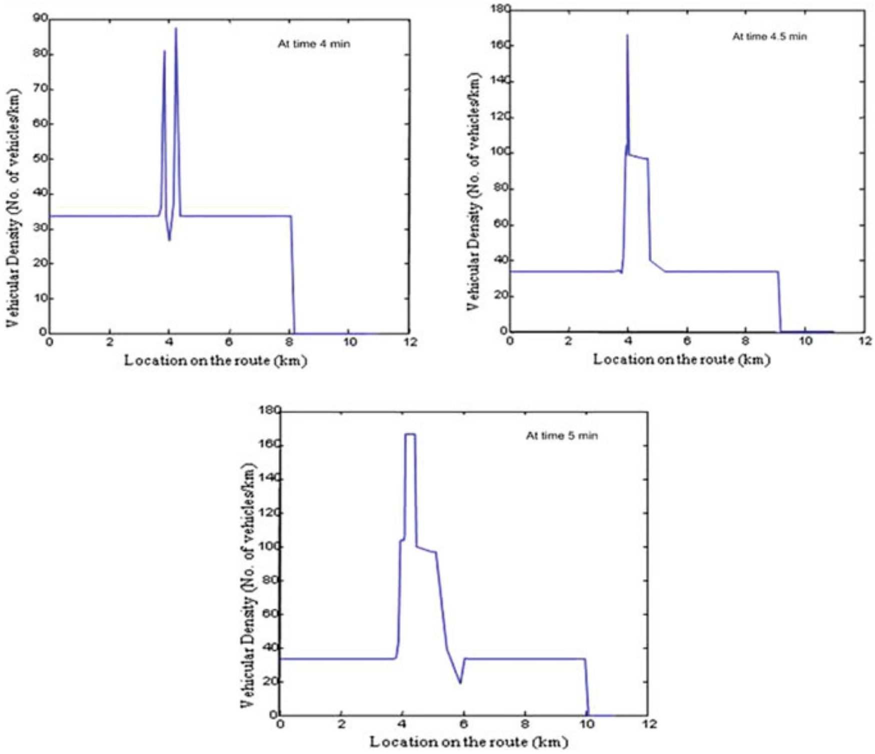


Fig. 10.8 Vehicle Density on the signalized road at time 4 min, 4.5 m and 5 min under safety distance $= 2 \times car - length$

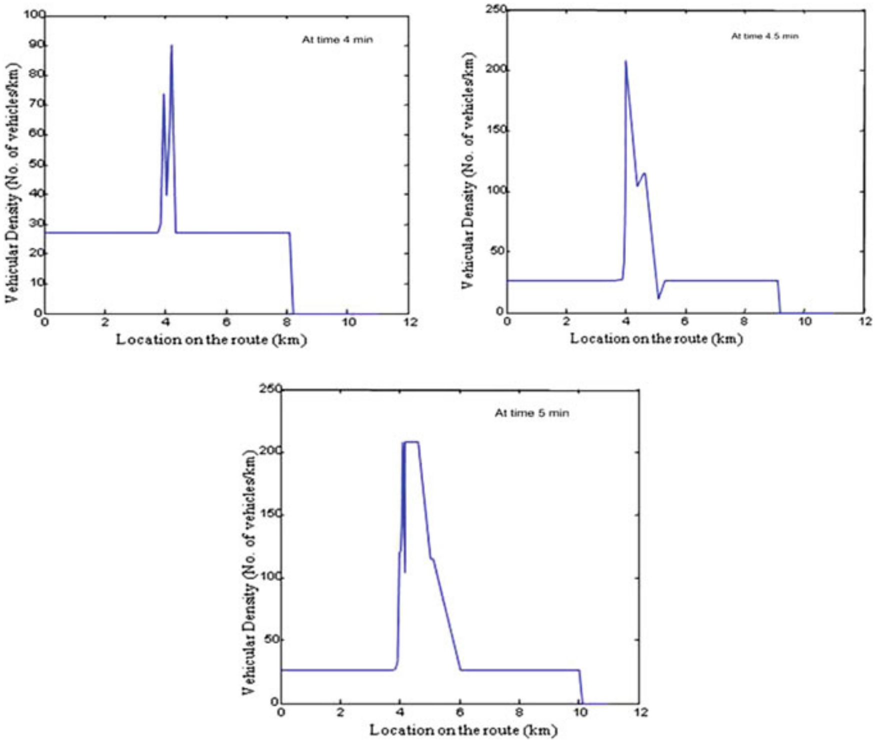


Fig. 10.9 Vehicle Density on the signalized road at time 4 min, 4.5 m and 5 min under safety distance $= 3 \times car - length$

As the outcome of the fluid dynamic model through the solution of differential equations provides vehicular density, whereas for the analysis of connectivity dynamics we need a number of vehicles in the covered area. This required data is achieved by the integration of vehicular density. The expression is given as: The mean number of vehicles within the covered distance (x1, x2) is

$$E[N_T(x_1, x_2, t)] = \int_{x_1}^{x_2} n(x, t)dx \tag{10.19}$$

In Fig. 10.8 the effect on vehicular density under the road condition such as: Safety distance between the $cars = Sdfc = 2 \times car - length = 8\,\text{m}$ where car length $=$ 4 m is captured at time 4, 4.5 and 5 min. As the traffic flows on the road the velocity profile is built under the influence of safety distance and front density of road traffic. The calculated velocity is affecting the road density due to the iterative process as it is used by the differential equation for density estimation in Eqs. 10.14a and 10.14b.

At the time 4 min due to the traffic signal the vehicle platoon formation is created which lasts for a stopping period shown as time 4.5 min in the graph. As the light

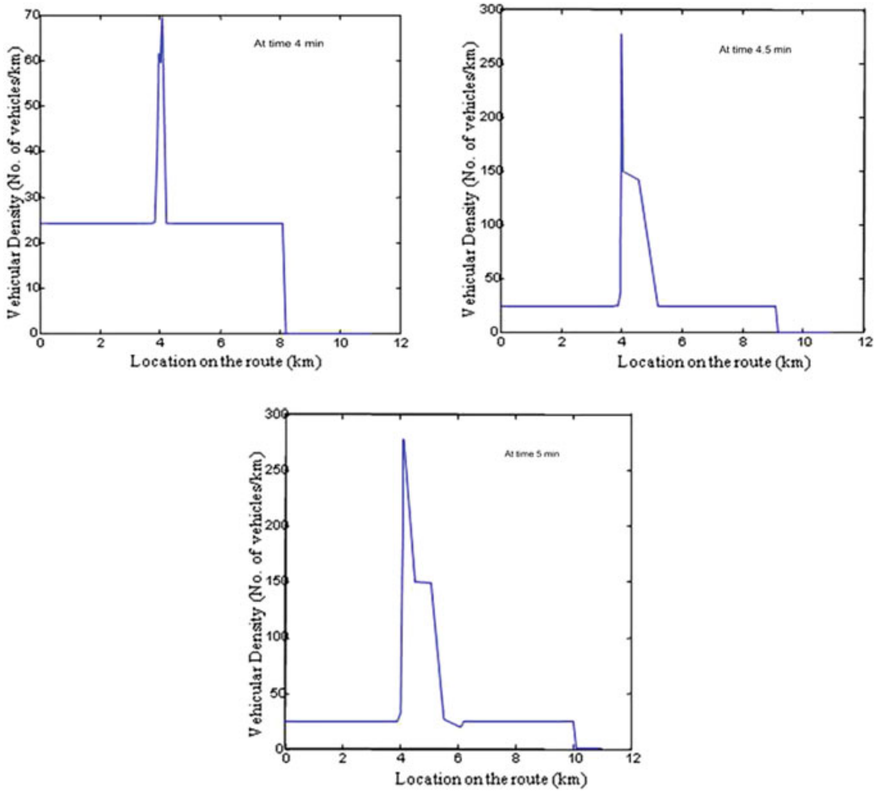


Fig. 10.10 Vehicle Density on the signalized road at time 4 min, 4.5 m and 5 min under safety distance $= 4 \times car - length$

goes green the vehicles disperse and the density curve shows normal behaviour after the time of 5 min.

In Fig. 10.9 the density dynamics are captured at different times during the traffic flow under assumed safety distance between cars defined as: Safety distance between the $cars = Sdfc = 3 * car - length = 8$ m where car length $= 4$ m The density graph starts building at 4 min at location 4 Km along the road due to the traffic signal implementation as the traffic light turns green the vehicles fast dispersion make an impact on vehicular density which is shown at time 4.5 and 5 min.

For the Fig. 10.10 the safety distance between the cars is assumed as: Safety distance between the $cars = Sdfc = 4 \times car - length = 16$ m where car length $= 4$ m.

The vehicular densities under these assumptions are achieved at time 4, 4.5 and 5 min. The variation in safety distances impact the traffic flow on the road. A smooth vehicular density peak is built at 4 min when red light is in operation. The vehicles dispersed with the green light at 4.5 and at 5 min traffic flow makes the density graph at a constant level.

The vehicular density under $2 \times car - length$ safety distance shows high formation of vehicle platoon. The red light implementation stops vehicles at 4 min and creates vehicles stoppage.

The formation and dispersion of vehicles on the road is well captured at 5 min. The dissipation of vehicles during safety distance $2 \times car - length$ is smooth. Due to an adequate number of vehicles on the road by keeping less safety distance between them, platoon formation is captured more clearly as compared to other cases when safety distance between the cars is 3*car-length and 4*car-length.

10.5 Summary

The traffic mobility model presented in this chapter is able to capture more realistic traffic flow conditions for the different road scenarios. With the use of the fluid dynamic model and the implementation of key microscopic parameters of safety distance and vehicular length under heterogeneous traffic flow environments the achieved vehicular densities provide more realistic data as compared to previous studies. The availability of different types of vehicles is considered in a mathematical model by using partial differential equations to find the total vehicular density. The influence of important constraints of vehicular structural characteristics and moving patterns on the density is focused in further study. The impact of microscopic parameters of safety distance and leading vehicles on vehicular density and velocity is captured. The mobility of vehicles can be manipulated with the optimal use of safety distance between the vehicles. The mobility model also provides useful data for further different VANET analysis.

References

1. A. Agarwal, T.D.C. Little, Impact of asymmetric traffic densities on delay tolerant vehicular ad hoc network, in *Vehicular Networking Conference (VNC), 2009 IEEE*, (IEEE, 2009), pp. 1–8
2. M. Artimy, Local density estimation and dynamic transmission-range assignment in vehicular ad hoc network, in *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 3 (2007), pp. 400–412
3. M.M. Artimy, W. Robertson, W.J. Phillips, Assignment of dynamic transmission range based on estimation of vehicle density, in *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, (ACM, 2005), pp. 40–48
4. M.M. Artimy, W. Robertson, W.J. Phillips, Minimum transmission range in vehicular ad hoc networks over uninterrupted highways, in *Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE*, (IEEE, 2006), pp. 1400–1405
5. M. Behrisch et al., SUMO-Simulation of Urban MObility, in *The Third International Conference on Advances in System Simulation (SIMUL 2011)*, (Barcelona, Spain, 2011)
6. W.F. Chan, M.L. Sim, S.W. Lee, Performance analysis of vehicular ad hoc networks with realistic mobility pattern, in *IEEE International Conference on Telecommunications and Malaysia International Conference on Communications, 2007. ICT-MICC 2007*, (IEEE, 2007), pp. 318–323

7. H. Conceicao, M. Ferreira, J. Barros, A cautionary view of mobility and connectivity modeling in vehicular ad-hoc networks, in *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, (IEEE, 2009), pp. 1–5
8. S. Durrani, X. Zhou, A. Chandra, Effect of vehicle mobility on connectivity of vehicular ad hoc networks, in *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, (IEEE, 2010), pp. 1–5
9. FHWA, http://www.fhwa.dot.gov/research. Accessed 10 May 2015
10. M. Fiorani et al. SASPENCE-Safe speed and safe distance: project overview and customer benefit analysis of a novel driver's collision avoidance support system, in *Proceedings of the 5th European Congress and Exhibition on Intelligent Transport Systems and Services*, (Hannover, Germany, 2005)
11. F.L. Hall, Traffic stream characteristics, in *Traffic Flow Theory. US Federal Highway Administration* (1996)
12. I.W.-H. Ho, K.K. Leung, J.W. Polak, Stochastic model and connectivity dynamics for VANETs in signalized road systems. In: *IEEE/ACM Transactions on Networking (TON)*, vol. 19, no. 1 (2011), pp. 195–208
13. I.W.H. Ho, K.K, Leung, Node connectivity in vehicular ad hoc networks with structured mobility, in *32nd IEEE Conference on Local Computer Networks, 2007. LCN 2007*, (IEEE, 2007), pp. 635–642
14. S.P. Hoogendoorn, P.H.L. Bovy, State-of-the-art of vehicular trafficflow modelling, in *Proceedings of the Institution of Mechanical Engineers*. Part I: J. Syst. Control Eng. **215**(4), 283–303 (2001)
15. B.S. Kerner, *Introduction to Modern Traffic Flow Theory and Control: The Long Toad to Three-Phase Traffic Theory*, (Springer Science & Business Media, 2009)
16. M. Khabazian, M.K. Mehmet Ali, A performance modeling of connectivity in vehicular ad hoc networks, in *IEEE Transactions on Vehicular Technology*, vol. 57, no. 4 (2008), pp. 2440–2450
17. S. Kukliński, G, Wolny, Density based clustering algorithm for VANETs, in *5th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities and Workshops, 2009. TridentCom 2009*, (IEEE, 2009), pp. 1–6
18. M. Kutz, *Handbook of Transportation Engineering*, vol. 768 (McGraw-Hill, New York, 2004)
19. K.K. Leung, W. Massey, W. Whitt et al., Traffic models for wireless communication networks. IEEE J. Sel. Areas Commun. **12**(8), 1353–1364 (1994)
20. R.T. Luttinen et al., *Statistical Analysis of Vehicle Time Headways*, (Helsinki University of Technology, 1996)
21. W.A. Massey, W. Whitt, A stochastic model to capture space and time dynamics in wireless communication systems, in *Probability in the Engineering and Informational Sciences*, vol. 8, no. 04 (1994), pp. 541–569
22. S. Panichpapiboon, W. Pattara-atikom, Evaluation of a neighborbased vehicle density estimation scheme, in: *8th International Conference on ITS Telecommunications, 2008. ITST 2008*, (IEEE, 2008), pp. 294–298
23. T.Q. Tang et al., A new dynamic model for heterogeneous traffic flow. Phys. Lett. A **373**(29), 2461–2466 (2009)
24. The Highway Code, http://www.direct.gov.uk. Accessed 10 May 2015
25. T. Umer et al., Implementation of microscopic parameters for density estimation of heterogeneous traffic flow for VANET, in *2010 7th International Symposium on Communication Systems Networks and Digital Signal Processing (CSNDSP)*, (IEEE, 2010), pp. 66–70
26. S. Yousefi et al., Improving connectivity in vehicular ad hoc networks: an analytical study. Comput. Commun. **31**(9), 1653–1659 (2008)

Chapter 11
HDy Copilot: A Mobile Application for Automatic Accident Detection and Multimodal Alert Dissemination

Bruno Fernandes, Muhammad Alam, Vitor Gomes, Joaquim Ferreira and Arnaldo Oliveira

Abstract The rapid technological growth is now providing global opportunities to enable intelligent transportation system (ITS) to tackle road traffic accidents which is considered one of the world's largest public injury prevention problem. For this purpose, eCall is an initiative by EU with the purpose to bring rapid assistance to an accident location presents HDy Copilot, an application for accident detection integrated with multimodal alert dissemination, both via eCall and IEEE 802.11p (ITS-G5). The proposed accident detection algorithm receives inputs from the vehicle, via ODB-II, and from the smartphone sensors, namely the accelerometer, the magnetometer and the gyroscope. Android smartphone is used as human machine interface, so that the driver can configure the application, receive road hazard warnings issued by other vehicles in the vicinity and cancel countdown procedures upon false accident detection. The HDy Copilot is developed for Android OS as it provides open source APIs that allow access to its hardware resources. The application is implemented and tested on IEEE 802.11p based prototype and the generated results show that it successfully detects collisions, rollovers and performs the eCall along with sending Minimum Set of Data (MSD).

This chapter is the extended version of the paper published in the IEEE VTC2015-Spring Conference.

B. Fernandes (✉) · M. Alam
Instituto de Telecomunicações, Campus Universitário de Santiago,
Aveiro, Portugal
e-mail: brunofernandes@ua.pt

M. Alam
e-mail: alam@av.it.pt

J. Ferreira
Instituto de Telecomunicações, ESTGA—Universidade de Aveiro, Aveiro, Portugal
e-mail: jjcf@ua.pt

V. Gomes (✉) · A. Oliveira
Instituto de Telecomunicações, DETI - Universidade de Aveiro, Aveiro, Portugal
e-mail: vitor.g@ua.pt

A. Oliveira
e-mail: arnaldo.oliveira@ua.pt

M. Alam et al. (eds.), *Intelligent Transportation Systems*,
Studies in Systems, Decision and Control 52,
DOI 10.1007/978-3-319-28183-4_11

11.1 Introduction

Despite the progresses that automotive industry achieved in producing safer and more efficient vehicles over the last years, road accidents are still high and more than 26,000 people died on the roads of the European Union (EU) in 2013 [11]. According to EU statistics 1,054,745 accidents were recorded in 2013 causing 1,387,957 injuries in 2013 and this is the lowest number since the last 10 years [11]. To address these concerns, new technological capabilities are being introduced in road vehicles. With the vehicular communication systems development, the Intelligent Transportation Systems (ITS) concept is emerging, targeting innovative services for traffic management to ensure safer roads and, well connected and coordinated transport networks. Currently, ITS is a strong research topic in the vehicular communications scientific community. A great number of universities, institutes, vehicle manufacturers and telecommunication companies are researching and developing solutions to be deployed at a large scale. The need for standardization is a concern, in order to unite and direct the research efforts. Therefore, the European Telecommunications Standards Institute (ETSI) and the Institute of Electrical and Electronics Engineers (IEEE) have already published standards to be followed in this research field.

With the advancement in telecommunications, it is expected that the communication between vehicles will be able to provide drivers more information about their surroundings, thus allowing them to make better decisions, resulting in the increase of their safety and efficiency. With more information, drivers can decide the best route to take, or even carefully approach a certain location within their route, knowing that the location is marked as unsafe.

High-end vehicles, today, offer some ITS services, such as turn-by-turn GPS navigation systems, accident detection system, as well as traffic, weather and entertainment applications built-in on vehicle's on-board computers. On older and lower end vehicles, smartphones are already being used, to bring those same features and services. Smartphones nowadays are a valuable solution to push ITS since they are powerful devices (in terms of performance and sensor capabilities) that can be integrated with vehicles.

Transportation safety, pollution reduction and time/costs efficiency are some of the most important goals in ITS. This concept aims to bring benefits to people's lives. According to [16], the benefits can be grouped into the following three categories: transport efficiency, environment preservation and safety increase. In this chapter we focus mainly on safety increase applications, namely: automatic accident detection, emergency assistance and road hazard warning dissemination.

11.1.1 Vehicular Accident Detection

Vehicle accidents that often cause more damage and human injuries happen due to, collisions and rollovers. When involved in a accident, a car can have a frontal, lateral,

Table 11.1 ASI and Theoretical Head Impact Velocity (THIV) scale values [14]

Impact severity level	Index values
A	ASI $\leq$ 1.0 $\wedge$ THIV $\leq$ 33 km/h
B	ASI $\leq$ 1.4 $\wedge$ THIV $\leq$ 33 km/h
C	ASI $\leq$ 1.9 $\wedge$ THIV $\leq$ 33 km/h

rear or even diagonal collision. Any of these directions are possible so, in order to develop an effective Autonomous Accident Detection (AAD) mechanism, all those types of accidents should be considered and detected.

11.1.1.1 Collision Detection

A collision generates a sudden change of speed over usually a short period of time and happens when an object slam into another object. The severity of the collision depends on the direction, orientation, duration of the variation of speed and velocity of both the colliding objects. If the objects are moving in the same direction but with different orientation, the collision will be more violent than in the case when they move with same direction and orientation. This means that, when the relative speed among objects increases, the collision will be more severe. This variation of speed over time $\left(\frac{\partial v}{\partial t}\right)$ is called acceleration. The acceleration generated during an accident is an important parameter to consider in collision/accident detection systems. Authors such as Weiner in [28], Thompson et al. in [26] and Kumar et al. in [23], describe in their publications accident detection systems, that use the 4g $\left(\mathrm{g} = 9.8\ \mathrm{m/s}^2\right)$ threshold, above which, an accident takes place. Thompson et al. also show that smartphone falls and light car breaks are unlikely to surpass the 4g threshold, which proves that this threshold acts as a correct filter for false detections. European road restraint systems are used to reduce the severity of accidents of vehicles leaving the road. To achieve this, these systems are evaluated based on the European standard EN1317 [13, 14]. This standard is based on the Acceleration Severity Index (ASI) and Theoretical Head Impact Velocity (THIV). Table 11.1 presents the ASI scale values.

This scale measures a collision impact severity and is divided in three levels. Impact severity level A is the less severe and C is the most severe. Level A designates light injury if any. This shows that on level B and above there is the risk of serious injury. Studies performed by Gabauer et al. in [17] and Shojaati in [24] demonstrate the relation between Acceleration Severity Index (ASI) and both Head Injury Criteria (HIC) and Abbreviated Injury Scale (ASI). Both HIC and AIS are metrics used to describe the injury severity of a vehicle occupant. To determine ASI it is required a tri-axial accelerometer to measure longitudinal (A_x), lateral $\left(\mathrm{A}_y\right)$ and vertical (A_z) acceleration components.

The procedure to compute ASI is the following:

1. Record the acceleration components (A_x, A_y and A_z) values.
2. Filter data with a four-pole phaseless Butterworth digital filter

   a. Evaluation components

$$T = \frac{1}{S} = \text{sampling time in seconds},\ \text{S} = \text{sample frequency}.$$

$$CFR = 13\,\text{Hz} = \text{filter cut-off frequency}$$

$$W_d = 2\pi CFR$$

$$W_a = \tan\left(W_d \frac{T}{2}\right)$$

$$a_0 = \frac{{W_a}^2}{1 + \sqrt{2}W_a + {W_a}^2}$$

$$a_1 = 2a_0$$

$$a_2 = a_0$$

$$b_1 = \frac{-2({W_a}^2 - 1)}{1 + \sqrt{2}W_a + {W_a}^2}$$

$$b_2 = \frac{-1 + \sqrt{2}W_a - {W_a}^2}{1 + \sqrt{2}W_a + {W_a}^2}$$

   b. For each of the three acceleration components, if $X(k)$ is the kth element of any series of measurements and $Y(k)$ is the kth element of the filtered series, then,

$$Y(k) = a_0X(k) + a_1X(k-1) + a_2X(k-2) + b_1Y(k-1) + b_2Y(k-2) \tag{11.1}$$

Equation 11.1 is a two pole filter. To perform the required four-pole filter data should pass to the filter twice (Eq. 11.2). Figure 11.1 depicts how the data is filtered.

$$Z(k) = a_0Y(k) + a_1Y(k-1) + a_2Y(k-2) + b_1Z(k-1) + b_2Z(k-2) \tag{11.2}$$

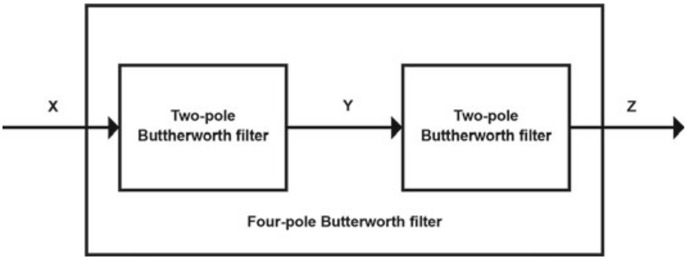


Fig. 11.1 Butterworth four-pole filter

3. Compute ASI as a function of time:

$$ASI = \sqrt{\left(\frac{\bar{A}_x}{\hat{A}_x}\right)^2 + \left(\frac{\bar{A}_y}{\hat{A}_y}\right)^2 + \left(\frac{\bar{A}_z}{\hat{A}_z}\right)^2} \tag{11.3}$$

   where $\bar{A}_x$, $\bar{A}_y$ and $\bar{A}_z$ are the filtered components of vehicle acceleration and $\hat{A}_x$, $\hat{A}_y$ and $\hat{A}_z$ are threshold values defined in EN 1317 [13]. For vehicle occupants with the seatbelt fasten, $\hat{A}_x = 12$, $\hat{A}_y = 9$ and $\hat{A}_z = 10$.
4. ASI should be calculated to at least two decimal places and report to one decimal place by mathematical rounding, i.e., 1,44 = 1,4 and 1.45 = 1,5.

11.1.1.2 Rollover Detection

A rollover happens when a vehicle rolls over it's main axis. To detect such rotations, it is necessary to analyse the rotation of the car's three main axis over time. Rollover accidents often are more damaging causing severe injuries and more damage to the vehicles. Therefore, ITS research is focusing more and more in rollover detection and proposing rollover systems for real time detection. As shown in the Fig. 11.2 the axis rotate with the car, i.e., they are fixed to the car, a rollover occurs when the vehicle rotates at least 45° over the X axis.

11.1.2 eCall

In 2013 there were 26,000 fatalities in European Union (EU) roads and this was the lowest number since 2001 [10]. The time to an injured person receive proper care from the Emergency Medical System (EMS) is related to the probability of death and trauma. According to Henriksson et al. [18] death and trauma rates can be reduced if there is a quicker reaction from the EMS. A quicker reaction can be obtained if help

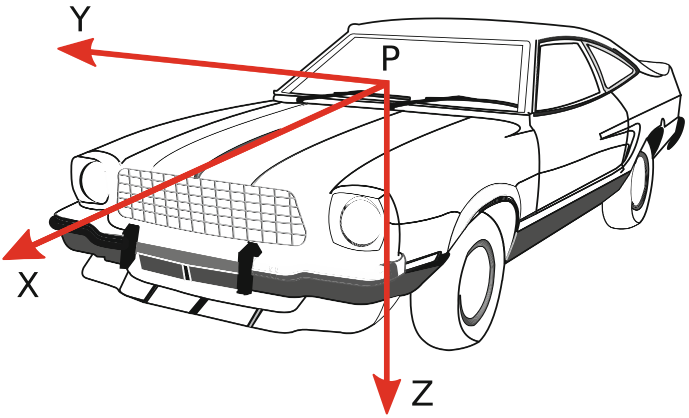


Fig. 11.2 Accelerometer axis direction and orientation

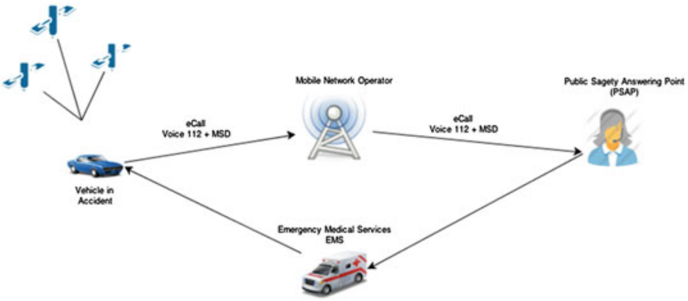
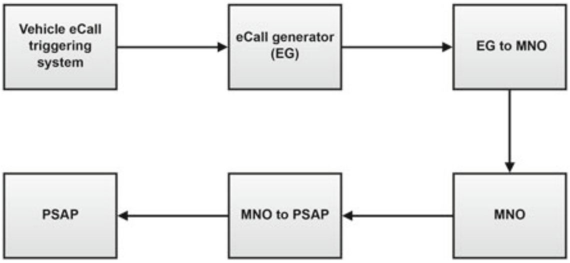


Fig. 11.3 eCall chain

is requested immediately after the accident event occur. Also if the exact location of the accident along with other extra information is provided to the EMS, a quicker and better response is possible.

The European Commission, in an attempt to provide a faster response from European EMS, declared the mandatory deployment of eCall in new cars by the end of 2017 or early 2018 [8]. eCall is a automatic accident detector that in the presence of an accident automatically requests help to the EMS through the European 112 emergency number. When an accident occurs, the car system performs an eCall that is composed by the voice call and a Minimum Set of Data (MSD) that is also transmitted, through the Mobile Network Operator (MNO), to the most appropriate Public Safety Answering Point (PSAP). The solution adopted for the MSD transmission, is an in-band modem that transmits data in the voice channel. The MSD should contain information to help speed up the EMS arrival to the accident scene. According with the eCall Driving Group recommendations [6] the MSD [12] should be sent in a 140 bytes packet.

Fig. 11.4 eCall service chain domains



11.1.2.1 eCall Chain

To perform an eCall, several technological aspects must be implemented in all the intervening parts of eCall chain. This chain is depicted in Fig. 11.3.

There are three parts involved in the eCall chain. These are the car manufacturers, the MNO and the participant countries. The European Commission adopted regulatory measures to mandate the technology deployment and upgrade on the three parts. Each part is responsible for upgrading their involved technology according to the eCall specifications.

As Fig. 11.3 depicts when a accident occurs, the car system performs an eCall that is composed by the voice call and a MSD that is also transmitted, through the MNO, to the most appropriate PSAP.

The European Commission, in an attempt to provide a faster response from European EMS, declared the mandatory deployment of eCall in new cars from 2017 or early 2018. The solution adopted for the MSD transmition, is an in-band modem that transmits data in the voice channel [1]. The MNO, in order to support this type of connection, need to upgrade their networks before the specified dates. European Commission also directs the PSAPs upgrading so that the information received, as a MSD, can be properly analysed. According to the eCall Driving Group recommendations [7], the MSD should be sent in a 140 Bytes containing the following information:

- **Control**—One Byte to specify if the eCall was automatically or manually triggered, if its a test call and if there is confidence in the location provided.
- **VIN**—Twenty Bytes for sending VIN according to ISO 3779 norm. Because VIN database for national and foreign registered vehicles might not be available in all Member States, the advantages of using this information on the PSAP should be further assessed.
- **Timestamp**—Four Bytes. The time of the accident should be provided in the UTC metric.
- **Location**—Latitude (four Bytes), Longitude (four Bytes) and direction of travel (one Byte) based on the last three positions.
- **Service Provider**—Four Bytes for service provider IP address in IPv4. Optional field.
- **Optional data**—Up to 106 bytes for other informations. Optional field.

The eCall system is supposed to work seamlessly in all participant countries, i.e., if a driver as an accident with his vehicle, in a foreign country, the eCall is performed to one of the PSAPs of that foreign country.

In the eCall Driver Group recommendations [7], the eCall service chain can be found. There are six main domains of this chain as it is depicted in Fig. 11.4. Each domain description is presented following:

- **Vehicle eCall Triggering System**—Compose by sensors that should detect front, side, rear and roll crashes. The trigger should be generate by the airbag module and/or a combination of other sensor data (e.g. gyro, radar, speed. Trigger thresholds based on speed variations could also be sent as optional data to help PSAPs predict the likeliness of serious injuries. The eCall can also be triggered manually.
- **EG**—In vehicle software triggers the eCall, provided the necessary info from the triggering system, and initiates the 112 call and MSD transmission through the in-band module.
- **EG to MNO**—The network receives the 112 call and the MSD.
- **MNO**—The mobile network operator (MNO) enriches the 112 call with CLI, MSD and cellular location.
- **MNO to PSAP**—Forwards the enriched 112 call to the appropriate PSAP.
- **PSAP**—Answers 112 voice call, decodes and visualises cell location and PSAP.

11.2 Related Work

Most of the (if not all) vehicle productions companies have their own built in systems to monitor and detect the accident systems such as Porsche Car Connect (PCC) system [21]. The PCC has auxiliary sensors in the bumper help to detect frontal collisions and also detects a threat of rollover and triggers the curtain airbags and seat belt pretensioners. The Porsche app includes features such as an emergency call is automatically made in the event of an accident, and the vehicle can also be located in most of Europe if it is stolen. Again, this and most related systems are limited to particular manufactures available on optional extra and bear high costs thus ignores availability of low cost smartphones.

An automatic traffic accident detection and notification mechanism called WreckWatch with smartphones is presented in [29]. The paper presents how smartphones can automatically detect traffic accidents using accelerometers and acoustic data and to notify a central emergency server after an accident. WreckWatch has a number of limitations such as acoustic data is not sufficient for detecting traffic accidents, ignores on-board sensors for detecting etc.

Accident detection and reporting system using GPS and GSM has been present in [2]. The GPS has been mainly used for monitoring the location and GSM to inform via call the monitoring authorities. The proposed algorithm has a number shortcomings as it does not monitor the actual accident and only considers speed as a parameter for accidents which is far away from reality. Going beyond the sole use of GPS, in [5] the authors have proposed an integrated system for emergency rescue services in the event of a road accident. The system (Black Box) focuses on building towards a commercial infrastructure which vehicle safety authorities can implement to enhance the reporting of vehicle crashes, provide post-crash analysis using motion sensors, record of the event in images and reduce the time it takes for emergency rescue to arrive at the crash location. The proposed system is good for commercial use only as it lacks the functionalities to consider the available sensors and devices on board and in the cell phones. Beside this, the system is very costly and not all cars can retrofit it.

A survey of both Android and iOS application repositories for applications promoting road safety, showed that only few applications have high ratings and positive user feedback. Examples of such applications are summarized next:

- **SaveDrives**—is a safety increase application but mainly used by the users for recording video of the journey and GPS tracking. SaveDrives also claims to provide a SaveDrives emergency service which with the help of built in accelerometer will detect serious accident and will inform the user's family and friends. This claim is not supported by technical details of the accident detection mechanism and also do not considers the specific eCall implementation.
- **WreckCheck**—is an auto accident checklist and mobile app that guides the users through essential steps to consider following an accident. The app uses mobile device's location service, audio recorder and camera to document the accident. Although the app is presented by National Association of Insurance Commissioners [20] to help claims of the insurance, it ignores a number of properties such as considering the on-board sensor's data etc.
- **Avertino**—is a safety increase application. Generates permanent, regular or even temporary road danger location based warnings. These events are reported by application users and are subject to confirmation by other users. When approaching a marked location, the application warns the user through a visual and audible alert. It also provides the possibility to visualize the reported event on a map. Available for iOS and Android.
- **iOnRoad Augmented Driving**—is another safety increase application. Uses augmented reality to analyze in real time, using the device camera, all the objects that are in front of the vehicle while the user drives. It generates alerts when the user is not respecting the minimum safety distance between cars or when an exit/entrance is approaching. Available for iOS and Android.
- **CarSafe**—is a driver safety increase application that, which combines information from vehicle's front and back cameras and others embedded sensors on the phone to detect unsafe driving conditions. The application is mainly used to monitor the drivers and road conditions and produce alert messages, it does not support accident detections.
- **Sprint Drive First**—is an application mainly focusing on driver's assistance. It automatically detects when users are driving, silences phone, and auto-replies to texts and calls. It does not detect any accident or initiate calls in case of accidents.

To authors' best knowledge, no other application implements an Automatic Accident Detection (AAD) and an eCall help request with the EU ITS-G5 standard. Most of the available applications facilitates drivers by providing either the information about the accident locations or about short and safe routes to the destination. Although, some R&D projects have also explored the smartphone capabilities in ITS but research findings are limited and most of them are simulations based [25]. The work in [19] describes the use of smartphones to assess road surface quality. The application takes advantage of the hardware features of modern smartphones like accelerometers and GPS. The sudden movements that occur when a car is upon a pothole produces acceleration and are interpreted in different ways. These accelerations along with the locations are reported through internet to a web service that can be accessed by the application users. The prior knowledge of the potholes are displayed on a map in the application to facilitate drivers decision making. In [4] Cano et al. present an Android application that connects to the car OBD-II system through Bluetooth. The aim is to detect accidents using smartphone's accelerometer and OBD-II airbag signal. When a 5g or higher acceleration is detected or when the airbag signal is triggered, the application starts a 1 min countdown timer. If the user does not cancel the countdown, the application validate the event as an accident and start the automatic help request procedure. This procedure consists in sending e-mail and an Short Message Service (SMS), with information regarding the accident such as location, to a specific service and to predefined user contacts. After the SMS and email are sent, a call is performed to the emergency services.

Some other similar research projects were found presenting basic work on accident detections, however no project was found integrating smartphones and IEEE 802.11p based vehicular communications and lacks implementation of MSD for eCall.

11.3 System Implementation

In order to provide an AAD mechanism and eCall implementation, a native Android application has been developed and connected to the IT2S platform through a Single Board Computer (SBC). Figure 11.5 depicts the hardware architecture while Fig. 11.6 shows the whole prototype implementation for the proposed mechanisms. The smartphone has been connected to the single board computer via USB connection, instead of using Bluetooth. The reasons for this design decision are twofold. Firstly, smartphones consumes more power when connected via Bluetooth. Secondly, it is difficult to guarantee real-time communications on top of a Bluetooth link compared to a USB connection.

The IT2S platform [15], is an ITS-G5 platform developed from scratch at Telecommunications Institute, Aveiro Portugal, funded by two research projects: Highway Environment ADvanced WArning sYstem (HEADWAY) [3] and Intelligent Cooperative Sensing for Improved traffic efficiency (ICSI) [9]. The main features of the platform useful for this project are the availability of a Global Positioning System (GPS) receiver, two Radio Frequency (RF) modules, a Field Programmable Gate

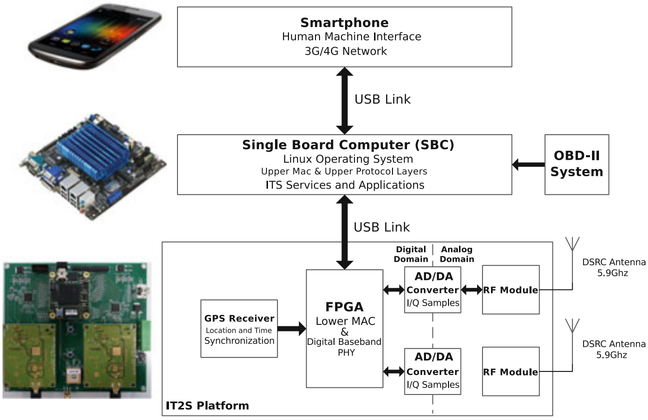


Fig. 11.5 Hardware architecture

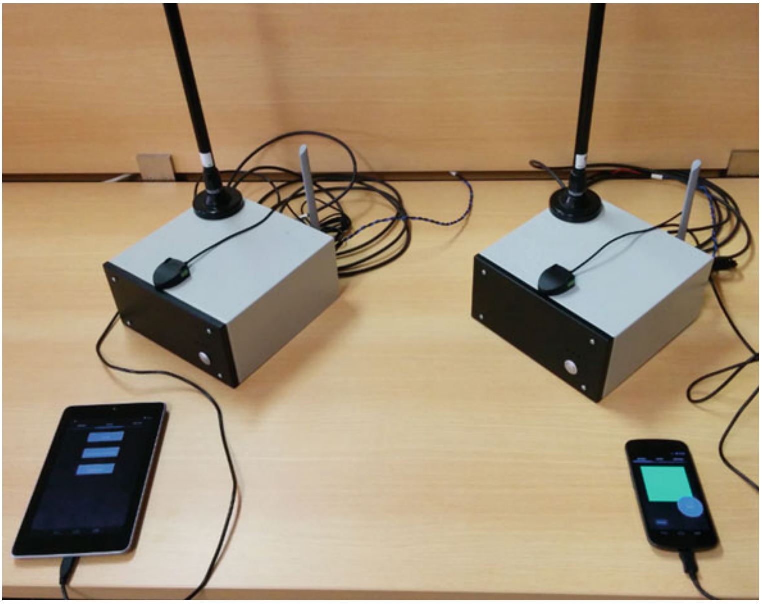


Fig. 11.6 System prototype

Array (FPGA) for lower MAC and baseband PHY layer implementation and Universal Serial Bus (USB) connections as shown in Fig. 11.5.

The IT2S platform provides external communication by USB. The Android device is integrated with the IT2S platform as an Application Unit (AU) to constantly exchange data. Two types of data are exchanged: the vehicle sensor data and the Road Hazard Warnings (RHWs) data. The vehicle sensor data, gathered from the vehicle

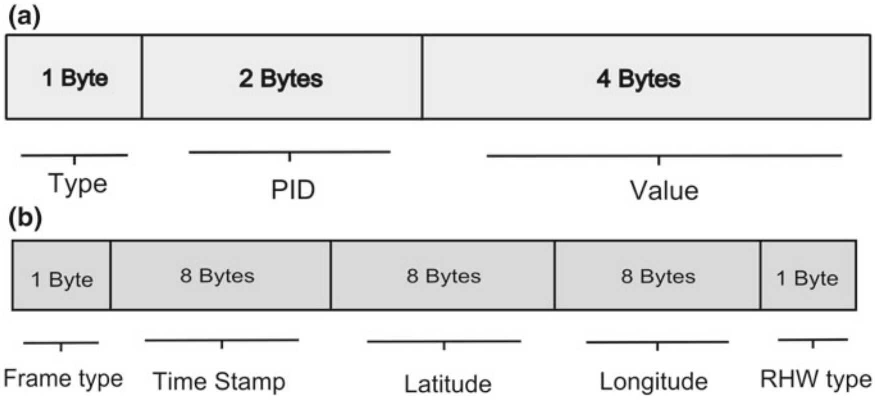


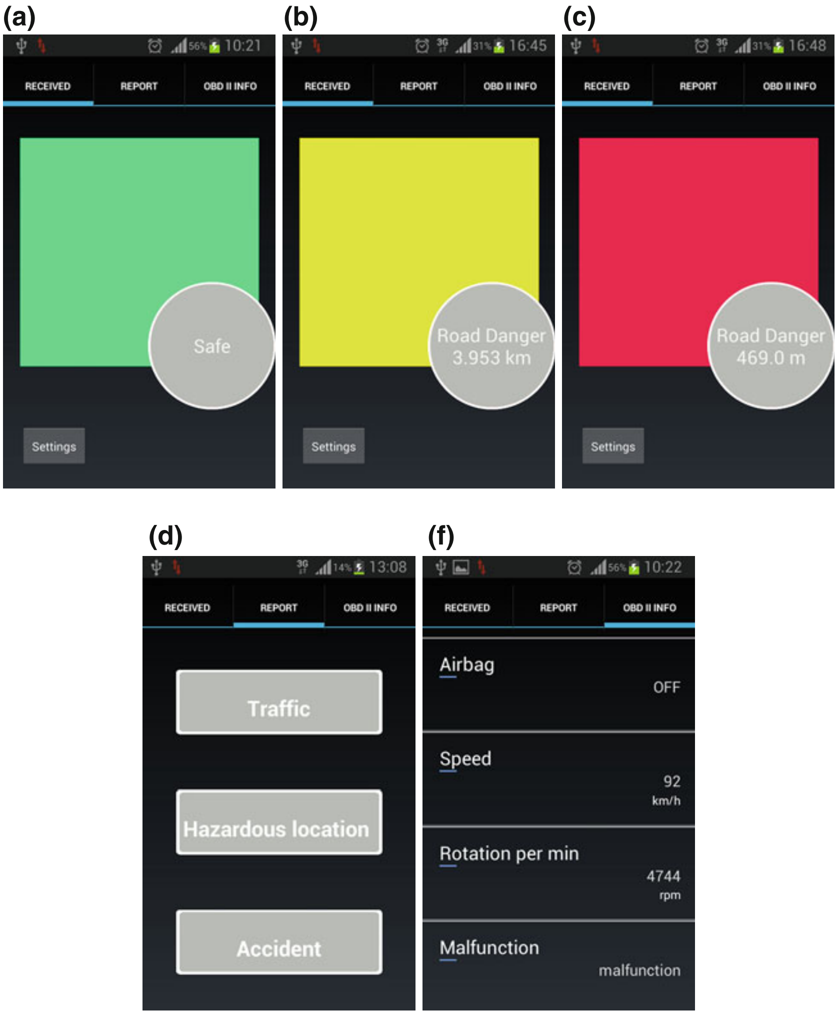
Fig. 11.7 Agreed data frames: **a** OBD-II data frame and **b** RHW data frame

using OBD-II protocol, is exchanged from the vehicle to the device. The RHW can be transmitted by both the device (RHW manual report, or AAD) and the vehicle (incoming RHW). To manage these data, the data frames agreed by the both communication entities are depicted in Fig. 11.7. As presented in Fig. 11.7a, the OBD-II data frame is composed by a frame type followed by the Parameter Identificator (PID) and the value. The frame type indicator is encoded with one byte and represents the type of data carried by the frame. The OBD-II signal PID field is encoded by two bytes and indicates the OBD-II signal (speed, airbag, etc.). The value field contains the value/quantity of the signal. For the majority of the signals two or four bytes are needed to encode this data.

Figure 11.7b presents the RHW data frame. Like the OBD-II data frame, this frame contains a frame type indicator encoded with one byte, followed by the RHW time stamp, latitude, longitude and type. The time stamp, latitude and longitude are encoded with eight bytes each. The RHW type is encoded with one byte. The USB connection can also charge the android device since it provides 500 mA current. This current allows the device to maintain its charge even when running the ADA.

A smartphone has been used as an AU due to its hardware resources and software (programmable) capabilities. In addition, it provides the three axis linear accelerometers and computational power required to estimate ASI and thus acts as an AAD mechanism. Furthermore, the Global System for Mobile Communications (GSM)/General Packet Radio Service (GPRS) capabilities are useful for eCall implementation. The developed application can be sub-divided into two modules:

- **Design**: is related to the Graphical User Interface (GUI) elements, such as icons, colours, layouts, images and visual effects.
- **Core**: is related to the application's functionality. It allows the GUI elements to perform tasks when demanded as well as AAD and eCall processing.



Airbag OFF

Speed 92 km/h

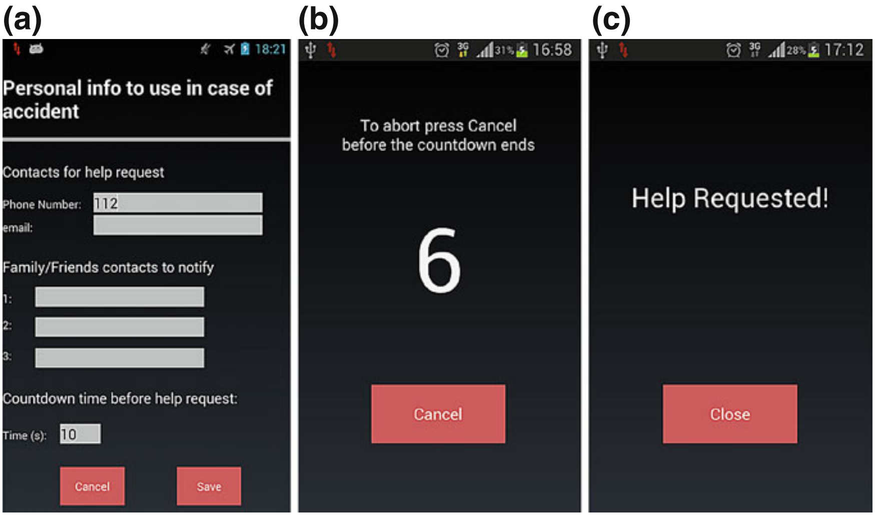
Rotation per min 4744 rpm

Malfunction malfunction

Fig. 11.8 Notification Activity GUI: **a** Safe situation. **b** Danger ahead. **c** Danger situation eminent. **d** Report Activity. **e** OBD-II Activity

11.3.1 Design

The Notification Activity GUI layout of the application is depicted in Fig. 11.8. It is composed by a large square panel that changes colours to: green, yellow and red, depending on the distance to a specific event. The main square panel has a grey circle that displays textual information about the type of event. There is also a "Settings" button to configure the program's settings on the lower left side of main screen. The Notification Activity has been developed to warn the driver of road's hazards by showing Road Hazard Warnings (RHWs) and the exact location where they occurred ahead on his route. The layout design is simple and enables users to retrieve information in a visual glance. At the Report Activity Fig. 11.8d, the user can, by clicking on the button's events, report about "Traffic", "Hazardous location" and "Accident" at current car's location. This event is also broadcast through IT2S platform to the nearby IT2S or other ITS-G5 platforms. Therefore, each connected device to the platform running this application will show the event on it's *Notification Activity*. The OBD-II info Activity Fig. 11.8e shows the information from OBD-II reader which is connected to the SBC as shown in Fig. 11.6. Figure 11.9a displays the eCall configuration screen. Here, the user can configure the national emergency contact number (eCall center's contact) as well as other contact numbers to be informed in case of an emergency. When the application detects an accident or vehicle rollover, a countdown event is initialized and displayed to user as shown in Fig. 11.9b. The countdown provides enough time to users to cancel the initiation of eCall or other relevant activity in case of false detection; for example, in the case of a phone drop. Once the countdown period expires an eCall (phone call + send MSD) is performed and a broadcast event is sent to IT2S platform as shown in Fig. 11.9c.



Personal info to use in case of accident

Contacts for help request

Phone Number: 112

email:

Family/Friends contacts to notify

1:

2:

3:

Countdown time before help request:

Time (s): 10

To abort press Cancel before the countdown ends

6

Help Requested!

Fig. 11.9 **a** eCall configuration settings Activity. **b** Accident detected (accident or rollover). **c** Help requested information

11.3.2 Core

11.3.2.1 Activity Diagram

To elaborate it further that how HDy Copilot behaves in a deeper perspective, an Activity diagram was elaborated. This diagram represents graphically HDy Copilot's work flow, i.e., the procedures resulting from the system functionalities and its interaction with the user and other actors.

In the Activity diagram the rounded objects are called actions. These actions define how the application deals with certain situations and together they describe the work flow of the application. The square object is a note. In this type of diagram parallel actions can be presented through the fork/join elements. HDy Copilot's Activity diagram is presented in Fig. 11.10. The application launches as soon as the Android device is connected through USB to IT2S platform. In Android there is always an activity responsible for launching the application. In this case its the MainActivity. Once this activity is created (Activities are created through the onCreate() callback method), it starts a service, named USBService, to establish a USB session. If the session initiation succeeds, the application starts exchanging data with IT2S platform. When the application launches, the MainActivity creates the Received, Report and OBDII activities and instantiates some of the required data, such as sensors, location systems, etc. It also proceeds with the Preferences action. This action's sub Activity diagram is depicted in Fig. 11.11. As Fig. 11.11 depicts, if the database is empty (happens when the application is launched for the first time) the UserSettings activity is launched. If the data fields are correctly filed, the data is saved to the database, the UserSettings activity is closed through the callback method onStop(). Since most devices possess a GPS location systems, that provides a fairly accurate location data, HDy Copilot uses the GPS for its ADA, for the RHW manual report and for calculating the distance between the user and an incoming RHW.

If the GPS is not activated, the user is prompt with a dialogue box the user has two choices, to press the Enable button, which leads to the Location Settings activity (activity not belonging to the application), or the Quit button, that leads to the presentation of a goodbye screen. If the user is at the Location Settings activity, he can activate the GPS. When he taps the device back button to navigate back into HDy Copilot, a verification is again made, and if the GPS is not activated the application will present the goodbye screen and terminate. If the GPS is enabled the Location Settings activity closes, the MainActivity is resumed and the user is presented with the application GUI.

At this moment the application launch state is over and its not on the execution state. The ADA is initiated and starts executing. The Run ADA action also has a rake symbol on the right hand side. For simplicity reasons, this action sub Activity is described later and depicted in Fig. 11.15.

Data is being exchanged with IT2S platform constantly. OBD-II messages and RHWs are the two types of data exchanged. The incoming data frames are constantly being analysed and in the case a RHW data frame is received, its information is dis-

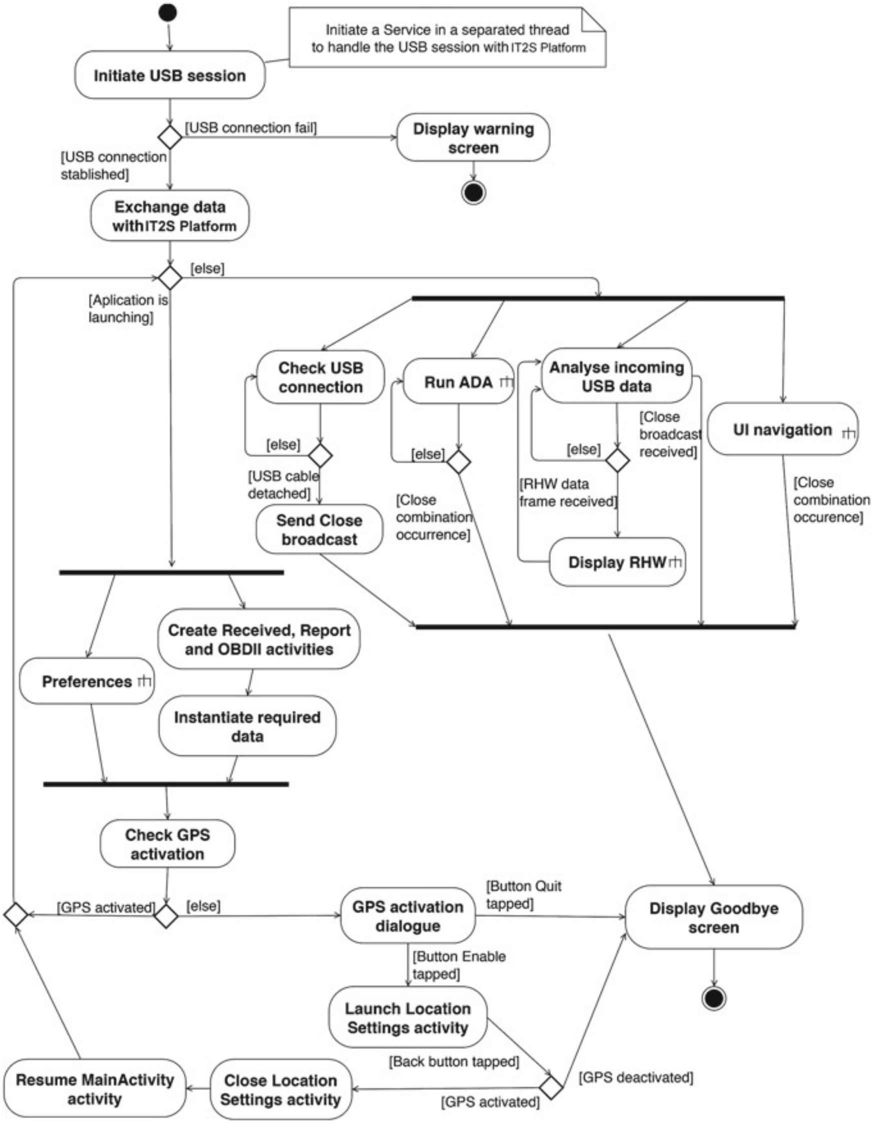

Initiate a Service in a separated thread to handle the USB session with IT2S Platform
Initiate USB session
[USB connection fail]
Display warning screen
[USB connection stablished]
Exchange data with IT2S Platform
[else]
[Aplication is launching]
Check USB connection
[else]
[USB cable detached]
Send Close broadcast
Run ADA
[else]
[Close combination occurrence]
Analyse incoming USB data
[Close broadcast received]
[else]
[RHW data frame received]
Display RHW
UI navigation
[Close combination occurence]
Create Received, Report and OBDII activities
Preferences
Instantiate required data
Check GPS activation
[GPS activated]
[else]
GPS activation dialogue
[Button Quit tapped]
Display Goodbye screen
[Button Enable tapped]
Launch Location Settings activity
[Back button tapped]
[GPS deactivated]
[GPS activated]
Close Location Settings activity
Resume MainActivity activity

Fig. 11.10 HDy Copilot's Activity diagram

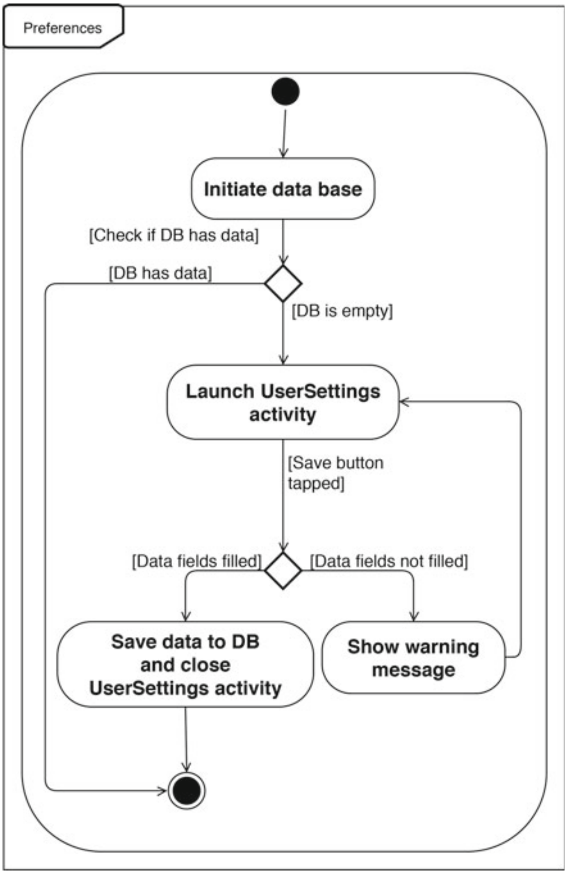


Fig. 11.11 Preferences sub Activity diagram

played to the user. This sub Activity diagram is explained and depicted in Fig. 11.12. Another important aspect of the application work flow, is the user interaction with it. This interaction is bundled in the User Interface (UI) Navigation action. This action sub Activity is depicted in Fig. 11.13.

11.3.2.2 Accident Detection Algorithm

The Accident Detection Algorithm (ADA) is at the core of the eCall system and aims to provide the AU the means to automatically detect vehicle accidents. To correctly identify vehicle accidents, the system should detect both collisions and rollovers. vehicles collisions produce certain acceleration values that can be used to predict the severity of injuries inflicted on passengers. As mentioned earlier, the Acceleration Severity Index (ASI) is used in Europe to evaluate the potential for occupant risk in full-scale crash tests involving roadside safety hardware. The acceleration gen-

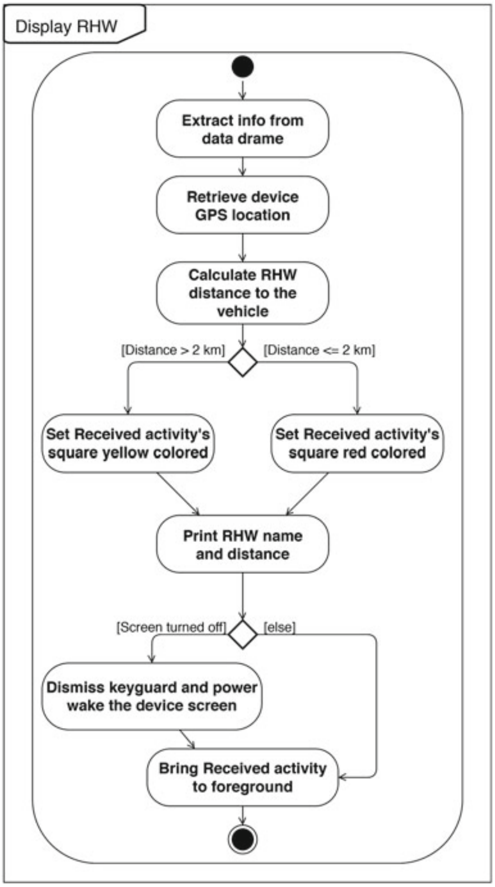

Display RHW
Extract info from data drame
Retrieve device GPS location
Calculate RHW distance to the vehicle
[Distance > 2 km]
[Distance <= 2 km]
Set Received activity's square yellow colored
Set Received activity's square red colored
Print RHW name and distance
[Screen turned off]
[else]
Dismiss keyguard and power wake the device screen
Bring Received activity to foreground

Fig. 11.12 Display RHW action sub Activity diagram

erated during an accident is studied by several authors [22, 27, 28], and the agreed threshold value is set to 4g($g = 9.8\ \text{m}/\text{s}^2$) and above this value an accident takes place. Therefore, in our system we use the threshold value 4g along with the ASI metric, i.e., the 4g threshold assesses if a car collision happens and the ASI metric provides an estimate of its severity. The 4g threshold is achieved during laboratory tests, proving that it is possible to detect such accelerations using smartphones.

To properly detect a collision, the vehicle and the smartphone must experience the same inelastic bound, so that the forces felt by the smartphone are equal as the vehicle. For this purpose, the smartphone should be firmly placed in the vehicle with the help of a smartphone holder or a similar solution. Another method used for detecting collisions is the airbag deployment. This information is available through the OBD-II messages transmitted by the vehicles' On-Board Diagnostic (OBD) system.

To detect rollovers, the algorithm constantly monitors the smartphone's orientation using sensor fusion technique. Using Android SensorManager API, the device orientation is determined through the `SensorManager.getOrientation()`

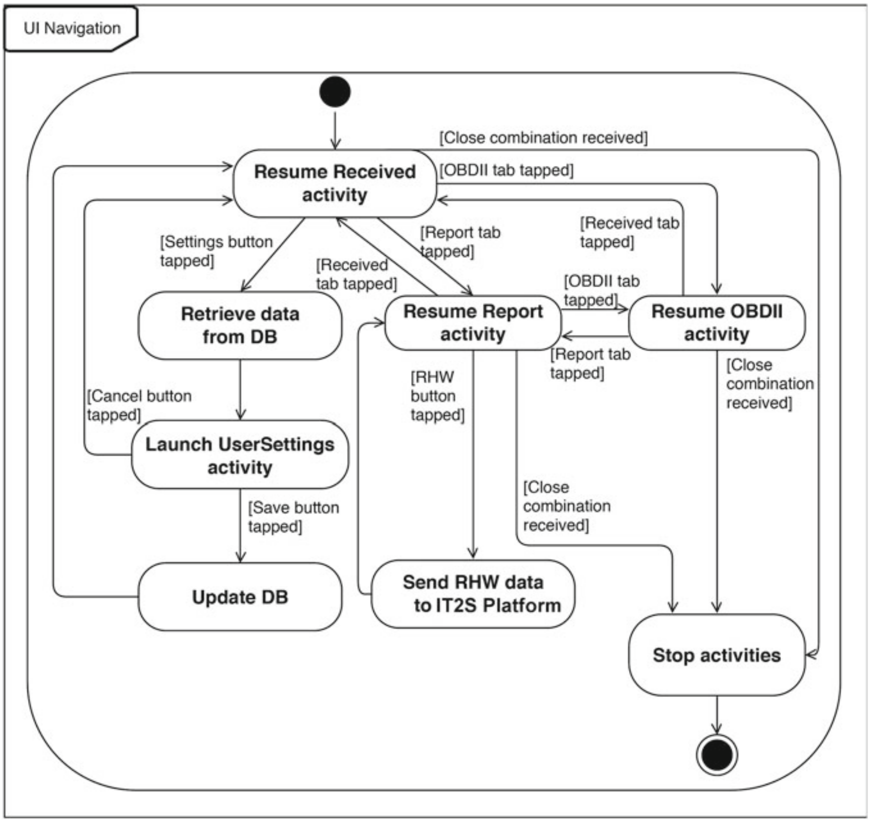


Fig. 11.13 UI navigation action sub Activity diagram

method using accelerometer and magnetometer data. The output of this method provides data with high frequency noise, because of the magnetometer. To remove the noise, a low-pass filter is applied. On the other hand, the sensor fusion algorithm is used to retrieve data from the gyroscope. This data is then multiplied by the sampling interval to determine rotation increment. The device orientation is the sum of all the rotations. To eliminate the resulting drift, a high-pass filter is applied. The resulting orientation is the sum of the low frequency components from the accelerometer/magnetometer orientation and the high frequency component of the gyroscope orientation. The sensor fusion flowchart is presented in Fig. 11.14.

This technique outputs the X, Y and Z orientation changes, in degrees, relative to the initial position. The decision of rollover occurrence is made when the smartphone rotates at least 45° over the device’s Z axis from its initial position.

The *Run ADA Activity* diagram is depicted in Fig. 11.15. After the algorithm initialization, it is continuously executed until the application is terminated. The algorithm uses the Linear Accelerometer, the Magnetometer and Gyroscope sensors

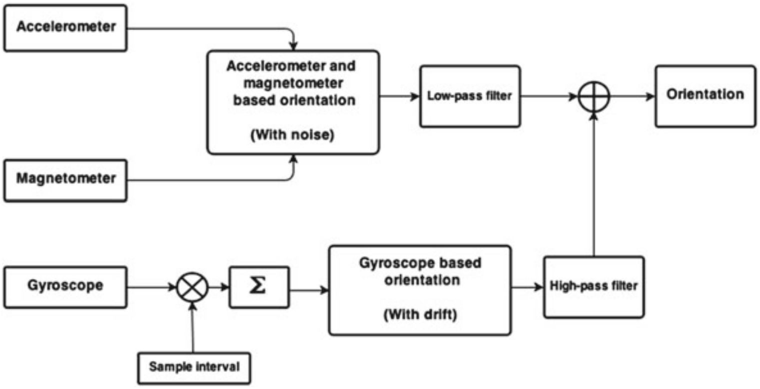


Fig. 11.14 Sensor fusion for the accident detection algorithm

from the Android smartphone. In addition, the algorithm also uses the vehicle sensor data, particularly, the airbag deployment signal. The sensors data, along with the user decision by not aborting the accident validation, is used to assess whether an accident has occurred or not.

Once the algorithm is initialized, it constantly monitors three data sources: the incoming USB data frames, the required device sensors and the device GPS location updates. They are represented by the Analyse Incoming USB Data, Read Device Sensors and Request Location Updates actions. The sensor fusion technique is applied at each new sensor sample. The algorithm detects accidents when one of three situations occur: the airbag deployment, rollovers or collisions. The algorithm is constantly analysing the incoming USB data and particularly, the OBD-II data frames. Once these frames are received, the data is extracted, and if it carries the airbag deployed signal, accident occurrence is validated and an accident detected broadcast is emitted.

The device's sensors are constantly being analysed all at the same sampling frequency and are calibrated at startup time. Then at each new sample, the sensors are read and the fusion sensor technique is applied (except for the linear accelerometer). Whenever the acceleration is below the 4g threshold, the process is repeated for the next sample. If the 4g threshold is surpassed, collision occurrence is validated, and the algorithm proceeds with the calculation of ASI and the transmission of the accident alert. If the device' position varies more than 45° from the initial position, the device mean speed is checked. This mean speed is calculated using the GPS API, and is constantly being updated. If the mean speed is greater than 20 km/h, the process continues, otherwise it ignores the change in position and calculates it again. Speed verification has been implemented to avoid false positives. The 20 km/h threshold is calculated based on the last 10 s prior to the device position change. This threshold assures that the vehicle was moving before the rollover occurred. After the device position change, the algorithm checks the instantaneous speed for 10 s. Subsequent to a rollover, the vehicle is usually immobilized. If that is the case, the instantaneous

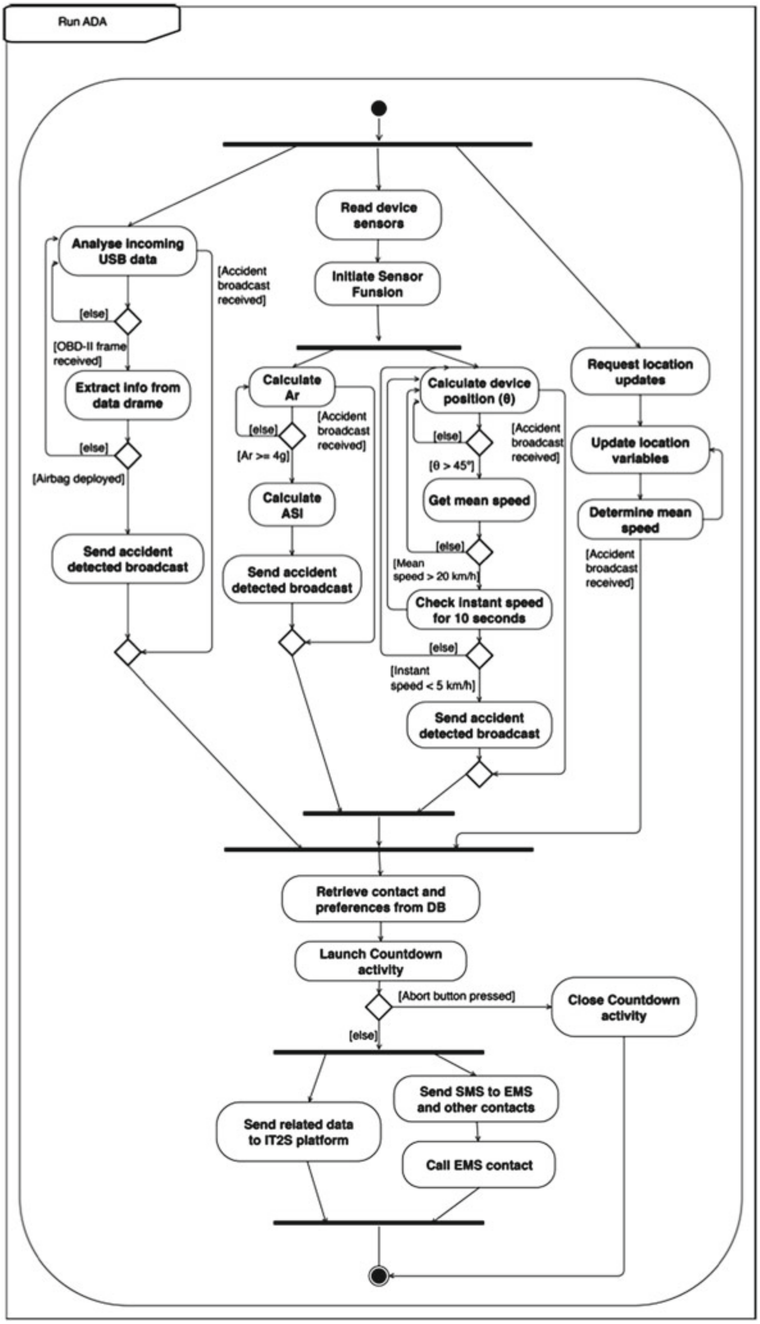

Run ADA
Read device sensors
Initiate Sensor Funsion
Analyse incoming USB data
[Accident broadcast received]
[else]
[OBD-II frame received]
Extract info from data drame
[else]
[Airbag deployed]
Send accident detected broadcast
Calculate Ar
[else]
[Ar >= 4g]
[Accident broadcast received]
Calculate ASI
Send accident detected broadcast
Calculate device position (θ)
[else]
[θ > 45°]
[Accident broadcast received]
Get mean speed
[else]
[Mean speed > 20 km/h]
Check instant speed for 10 seconds
[else]
[Instant speed < 5 km/h]
Send accident detected broadcast
Request location updates
Update location variables
Determine mean speed
[Accident broadcast received]
Retrieve contact and preferences from DB
Launch Countdown activity
[Abort button pressed]
Close Countdown activity
[else]
Send related data to IT2S platform
Send SMS to EMS and other contacts
Call EMS contact

Fig. 11.15 Activity diagram of the accident detection algorithm

speed should reach null. The threshold used is 5 km/h due to a verified GPS speed calculation inconsistencies at low speeds. If the instantaneous speed is lower than 5 km/h, the algorithm validates the rollover occurrence and broadcasts an alert message, otherwise it repeats the process from the beginning. Both the waiting time and the mean speed threshold are configurable values used for demonstration purposes only and can be modified, if they prove not be effective during real case tests.

Once the three accident detectors validate an accident occurrence, the algorithm proceeds to retrieve stored information in database and launches the *Countdown Activity*. The countdown time is configured by the user in the *UserSettings activity*.

The accident alert is broadcast by two sources. The first is the vehicular network, by transmitting a Decentralized Environmental Notification Message (DENM) message containing a Road Hazard Warning via the IT2S platform. The second is the Emergency Medical System (EMS), by performing an eCall. During development it was verified that it was difficult to perform an eCall with the provided APIs. The solution found to surpass this problem was to send an SMS, containing the eCall minimum set of data, followed by a voice call to the EMS. In the SMS's content only the Vehicle Identification Number (VIN) is not included. This is due to the fact that, it is still under assessment if in fact should be used, by the eCall responsible authorities. The SMS sent to the EMS is also sent to the stored friends/family contacts.

To provide the required functionality, the following software services are implemented:

- **Android Manager**: is a service that handles the communication and events between Smartphone and other peripherals
- **OBD-II Manager**: is responsible to read and send OBD-II data.
- **IT2S Manager**: manages the broadcast messages and sends them to the IT2S platform.

11.4 System Validation

After the HDy Copilot has been implemented, laboratory tests are performed to assess its robustness and weaknesses. The most important features of HDy Copilot is AAD mechanism, and eCall. The accident detection algorithm is developed to detect car accidents, in case of collisions or rollovers an eCall is performed. Several tests have been performed to assess if the algorithm and the device itself can detect both types of accidents. Airbag deployment signals are not validated as they require a real car.

11.4.1 Collision Detection Tests

The ADA detects collisions using the airbag OBD-II signal and the linear accelerometer built in smartphone. The accelerometer outputs accelerations sensed by the three

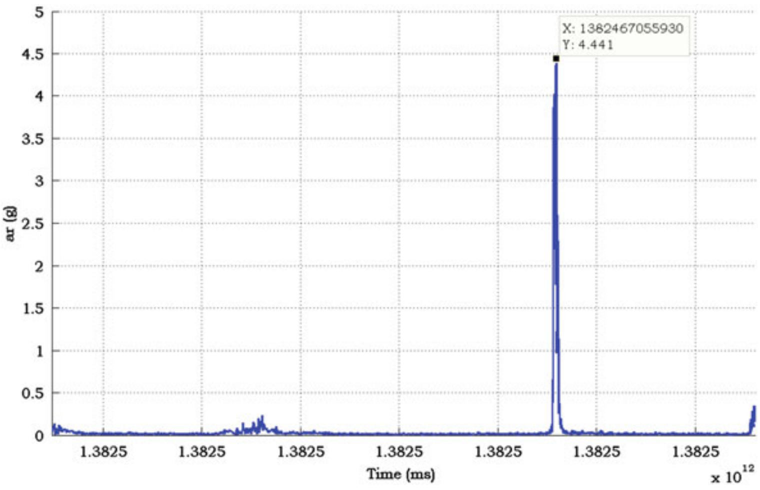


Fig. 11.16 Acceleration in case of slim pulse

axis without the influence of gravity. To validate an accident, the resulting acceleration ($\overrightarrow{ar}$) must be equal or greater than $4g$, as explained in the algorithm section. The first test performed is to assess if the smartphone linear accelerometer could sense accelerations at this magnitude. To achieve elevated accelerations, high variations of speed are required. This sudden change in speed is difficult to achieve without damaging the device. To address this issue, the acceleration values have been generated by shaking the device violently.

Figure 11.16 presents the generated $\overrightarrow{ar}$ value over the sampling time for the first test in which the device has been violently jiggled once (one swing). The sampling time is the time stamp in Coordinated Universal Time (UTC) of each new linear acceleration sensor sample recorded during the test. The pulse maximum value in this test recorded $\overrightarrow{ar} = 4.441$g at the 1382467055930 UTC instant. This proves that it is possible to surpass the 4g threshold and also indicates that the device can detect collisions once the $\overrightarrow{ar}$ value is greater than the threshold. ASI evolution for this test is presented in Fig. 11.17, the maximum ASI value detected is ASI = 0.34, which leads to an ASI level of A, the least severe level.

Figure 11.18 depicted the amplified view of the pulse. The pulse has the duration of approximately 505 ms. The measurement is made using the elapsed time between the closest sample to $\overrightarrow{ar} = 1$g at the rise moment (1382467055629 UTC) and the falling moment (1382467056134 UTC), as depicted in Fig. 11.18. With this test (one swing), it is not possible to achieve significantly higher $\overrightarrow{ar}$ and ASI, as a consequence the level of severity of the accident is not high enough to initiate the AAD.

This means that for larger $\overrightarrow{ar}$ pulse widths, the resulting maximum ASI value obtained should be greater. To prove this statement, another test has been performed which consisted in violently shaking the device several times (several swings) in order to keep the $\overrightarrow{ar}$ values high for an longer period of time. Figure 11.19 presents the

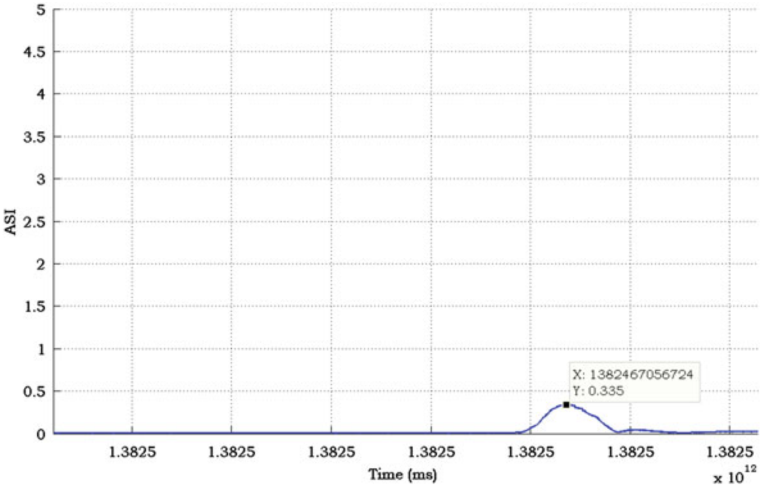


Fig. 11.17 ASI value resulting form ar slim pulse

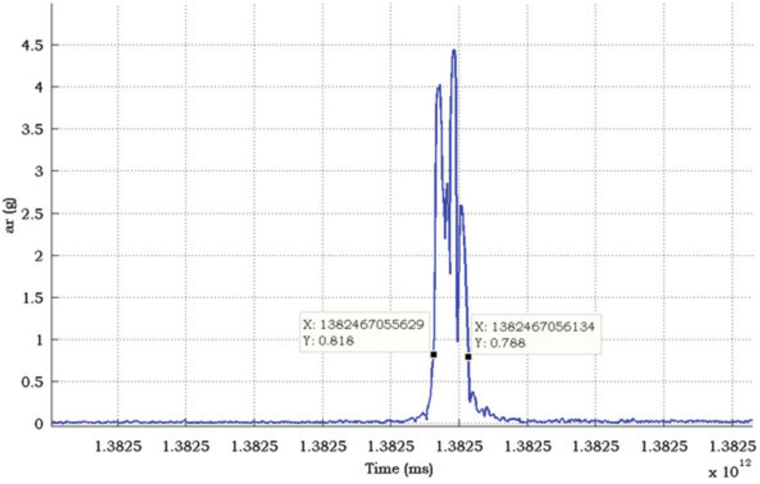


Fig. 11.18 **ar** slim pulse amplified view

results of this test. Now the maximum $\overrightarrow{ar} = 4.461$g and the pulse width is 1918 ms. The pulse amplified view is depicted in Fig. 11.20 and maximum ASI in Fig. 11.21. The maximum ASI value resulting from this pulse is ASI = 1.73, as presented in Fig. 11.21, corresponds to an ASI level of C. This leads to the conclusion that the application will report an ASI level not only depending on the maximum $\overrightarrow{ar}$, but also taking in account the direction of the collision in consideration.

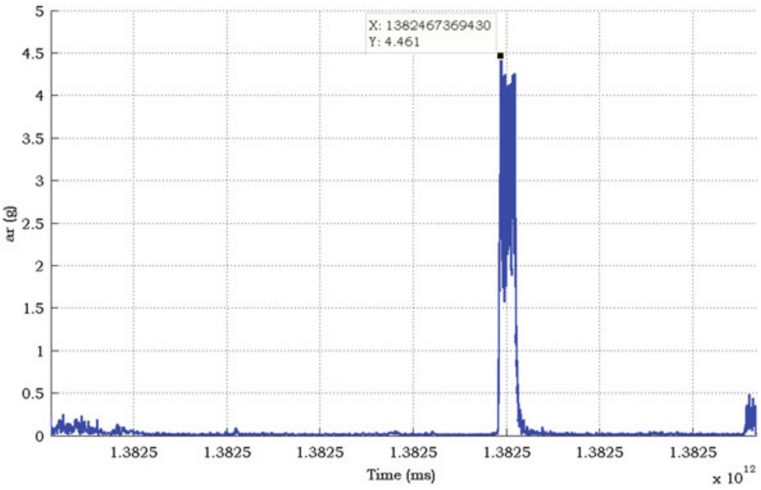


Fig. 11.19 **ar** large pulse evolution

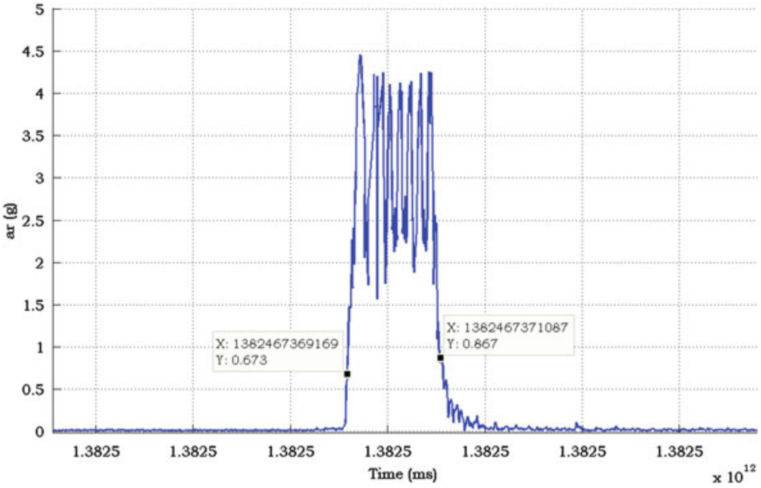


Fig. 11.20 **ar** large pulse amplified view

11.4.2 Rollover Detection Tests

To validate a rollover occurrence, a minimum of 45° change from the device initial position, over its Z axis (azimuth) should occur. The validation of rollover considers the mean speed of the vehicle and its instant speed after the position changes. The test performed aims to verify if the output of the sensor fusion technique is accurate. It is based on changing the device from portrait mode (initial position) to landscape mode over a flat table. This test enables a rotation over the device's Z axis only, leaving the

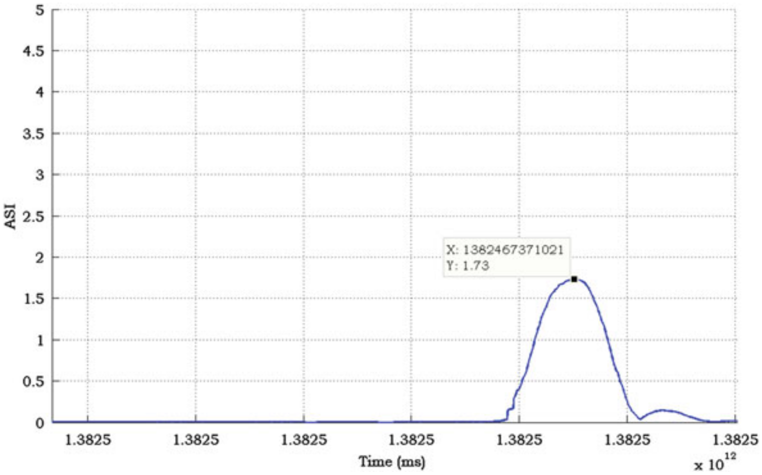


Fig. 11.21 a**r** large pulse amplified view

X and Y axis rotation null. Figure 11.22 presents the result of the test. It is observed that the position change between portrait and landscape is 90°, which the sensor fusion technique captured correctly. With these results, its possible to conclude that the device position is correctly detected, allowing the application to detect rollovers.

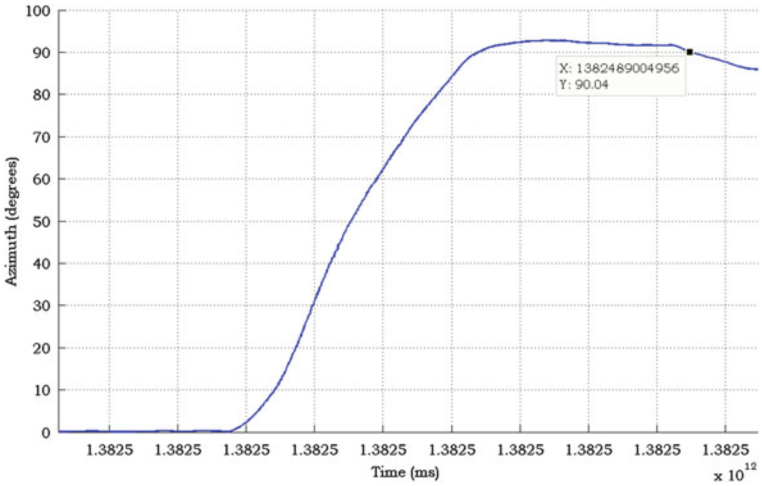


Fig. 11.22 Azimuth (Z axis rotation) evolution

11.4.3 Robustness Tests

After validating the capability to detect accidents based on the device's sensors, the application behaviour is tested in different scenarios to assess its robustness. These tests have been performed to the application at runtime and the various observed behaviour is recorded and presented in Tables 11.2, 11.3 and 11.4.

The first robustness test aims to verify the rollover. It is performed with mean speed ($\bar{v}$) and instant speed ($\mathbf{v}_i$) values. Once the application detects a rollover, it responds by launching the application's *Countdown activity*. As expected, it only validated the rollover occurrence when mean speed $\geq$ 20 m/s and instant speed $\leq$ 5 m/s. The elapsed time between the rollover validation and the launch of the *Countdown Activity* is 690 ms, which proves that the application responds quickly. The different tests performed are represented in Table 11.2.

When the *Countdown Activity* is launched and if the countdown is not interrupted, HDy Copilot proceeds to alarm transmission: data is send to the IT2S platform after 10494 ms, a SMS is sent after 10,094 ms and a voice call is placed after 5093 ms of

Table 11.2 Rollover detection robustness test

Rotation detected ($\theta \geq 45°$)		
Scenario	Response	Elapsed time from detection (ms)
$v = 21 \wedge v_i = 4$	Nothing happens	–
$v = 19 \wedge v_i = 10$	Nothing happens	–
$v = 19 \wedge v_i = 4$	Nothing happens	–
$v = 20 \wedge v_i = 5$	Countdown launch	625

Table 11.3 Help request procedure elapsed time

Start event	End event	Elapsed time (ms)
Countdown activity launch	Send data to OBU	10,494
Countdown activity launch	Send data to OBU	10,094
SMS delivered	Call activity launch	5093

Table 11.4 Background execution robustness tests

Event	Scenario	Response
Collision detected	USB connected	Help request concluded in full
Collision detected	USB disconnected	Help request concluded. No data sent to OBU
Rollover detected	USB connected	Help request concluded in full
Rollover detected	USB disconnected	Help request concluded. No data sent to OBU
RHW received	USB connected	Received activity brought to foreground

delivering the SMSs. These time measurements are the result of one test only and are presented to provide an idea of how the alarm broadcast is handled as represented in Table 11.3. In this particular test, the countdown time has been set for 10 s to verify that both the SMSs and data to the OBU were sent after the countdown timer expires. After the SMSs are sent, the call is performed once the delivery confirmation is received. The process time depends on the mobile network provider and does not depend on the application directly, and for this reason has not been accounted in the tests. Once the SMS delivered confirmation is received, the application waits 5 s to launch the voice call. This waiting time is proven required throughout the application development to guarantee that the GSM connectivity is not obstructed, otherwise the call could fail.

The last test was performed to assess the background execution capabilities of the HDy Copilot to detect accidents and transmit road hazard warnings. The experimental results (depicted in Table 11.4) show that the application does not depend fully on the USB connection to detect accidents. If the USB connection is down, HDy Copilot communication with the IT2S platform is terminated. In addition, it will stop receiving the OBD-II data and therefore, the vehicle airbag deployment signal will not trigger an alarm broadcast. However, this disconnection with the IT2S does not stop the application execution. For instance, in the scenario of a real car accident, the smartphone can be thrown away from its holder, causing the detachment of USB cable. In this situation, the application continues its execution and is able to detect collisions, rollovers and perform the implemented eCall solution and notify the user's previously configured contacts. When disconnected from the IT2S platform, HDy Copilot also ceases to receive and report road hazard warnings, as it is designed to transmit and received information to and from the OBU through USB only.

11.5 Conclusions

The use of smartphones, as an accident detection platform, presents a low-cost and portable solution compared to vendor specific built-in systems. This chapter presented an accident detection mechanism based on an Android smartphone, the ODB-II data, vehicular communications, and integrated eCall. The HDy Copilot has been developed for Android OS as it provides open source APIs that allow the access to almost all its hardware. The application uses Acceleration Severity Index to evaluate the potential for occupant risk in full-scale vehicle crash tests involving roadside safety hardware. Once the accident detectors validate an accident occurrence the algorithm initiates a countdown time. In case of accident (collision and Rollover) detection, a DENM message is broadcast to all the cars in the vicinity, in parallel with SMS and voice call to emergency numbers. To further control false positives, the warning notifications are only transmitted if the driver fails to interrupt a countdown sequence, which is automatically initiated by the accident detection algorithm. The system is thoroughly evaluated and verified fully functional by conducting collision and robustness tests.

References

1. 3GPP, eCall data transfer; In-band modem solution. TR 26.967. 3rd Generation Partnership Project (3GPP), Oct 2012. http://www.etsi.org/deliver/etsi_tr/126900_126999/126967/11.00.00_60/
2. M.S. Amin, J. Jalil, M.B.I. Reaz, Accident detection and reporting system using GPS, GPRS and GSM technology, in *2012 International Conference on Informatics, Electronics & Vision (ICIEV)* (IEEE, 2012), pp. 640–643
3. Brisa Inovação, HEADWAY—Connecting vehicles and highways. http://www.brisainovacao.pt/en/innovation/projects/headway (2015)
4. J. Carlos Cano et al., Providing accident detection in vehicular networks through OBD-II devices and android-based smartphones, in *Proceedings of the 5th IEEE Workshop On User Mobility and Vehicular Networks* (2011)
5. S. Chaklader et al., Black Box: an emergency rescue dispatch system for road vehicles for instant notification of road accidents and post crash analysis, in *2014 International Conference on Informatics, Electronics & Vision (ICIEV)* (IEEE, 2014), pp. 1–6
6. eCall Driving Group, Recommendations of the DG eCall for the introduction of the pan-European eCall. TS, Apr 2006
7. eCall Driving Group, Recommendations of the DG eCall for the introduction of the pan-European eCall. TS. Safety Forum. http://www.ecall.fi/Position_papers_DG_eCall_v2.pdf
8. European Commission, eCall: automated emergency call for road accidents mandatory in cars from 2015. http://europa.eu/rapid/pressrelease_IP-13-534_en.htm (2014)
9. European Commission, Intelligent cooperative sensing for improved traffic efficiency. http://www.ict-icsi.eu/ (2015)
10. European Commission, Statistics—accidents data. http://ec.europa.eu/transport/road_safety/specialist/statistics/index_en.htm (2015)
11. European Commission, Statistics—accidents data. http://ec.europa.eu/transport/road_safety/specialist/statistics/index_en.htm
12. European Committee for Standardization, Intelligent transport systems—eSafety—eCall: HGV/GV additional data concept specification. Technical report, Sept 2011
13. European Committee for Standardization, Road restraint systems—Part 1: Terminology and general criteria for test methods. Technical report, July 2010
14. European Committee for Standardization, Road restraint systems—Part 2: Performance classes, impact test acceptance criteria and test methods for safety barriers including vehicle parapets. Technical report, July 2010
15. J. Ferreira et al., Fail silent road side unit for vehicular communications, in *Proceedings of Workshop ASCoMS (Architecting Safety in Collaborative Mobile Systems) of the 32nd International Conference on Computer Safety, Reliability and Security* (2013)
16. L. Figueiredo, Sistemas Inteligentes de Transporte. Ph.D. thesis. Faculdade de Engenharia da Universidade do Porto, Feb 2005
17. D. Gabauer, H.C. Gabler, Evaluation of threshold values of acceleration severity index by using event data recorder technology. Transp. Res. Rec. J. Transp. Res. Board **1904**(1), 37–45 (2005)
18. E. Henriksson, M. Ostrom, A. Eriksson, Preventability of vehicle-related fatalities. Accid. Anal. Prev. **33**, 467–475 (2001)
19. G. Kanonirs et al., Towards vehicular sensor networks with android smartphones for road surface monitoring, in *Proceedings of the Second International Workshop on Networks of Cooperating Objects (CONET'11)* (2011)
20. National Association of Insurance Commissioners. http://www.naic.org/ (2015)
21. Porsche AG, Porsche Car Connect (PCC). http://www.porsche.com/usa/models/macan/macan-s/comfort/car-connect (2015)
22. D. Punetha, D. Kumar, V. Mehta, Article: design and realization of the Accelerometer based Transportation System (ATS). Int. J. Comput. Appl. **49**(15), 17–20 (2012)
23. D. Punetha, D. Kumar, V. Mehta, Design and realization of the accelerometer based transportation system. Int. J. Comput. Appl. **50** (2012)

24. M. Shojaati, Correlation between injury risk and impact severity index ASI. ETH Zurich (2003)
25. F.A. Teixeira et al., Vehicular networks using the IEEE 802.11p standard: an experimental analysis. Veh. Commun. **1**(2), 91–96 (2014). ISSN: 2214–2096
26. C. Thompson et al., Using smartphones to detect car accidents and provide situational awareness to emergency responders, in *Mobile Wireless Middleware, Operating Systems, and Applications* (Springer, 2010), pp. 29–42
27. C. Thompson et al., Using smartphones to detect car accidents and provide situational awareness to emergency responders, in *MOBILWARE'10* (2010), pp. 29–42
28. S. Weiner, Feasibility of a 802.11 VANET Based Car Accident Alert System. http://origin.www.ieee.org/documents/weiner_feasibility_802.11.pdf (2010)
29. J. White et al., Wreckwatch: automatic traffic accident detection and notification with smartphones. Mobile Netw. Appl. **16**(3), 285–303 (2011)