Design and evaluation of safety-critical applications based on inter-vehicle communication

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

von der Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)

genehmigte

Dissertation

von

Natalya An

aus Almaty, Kasachstan

Tag der mündlichen Prüfung:	11. Dezember 2015
Erster Gutachter:	Prof. Dr. rer. nat. Hannes Hartenstein Karlsruhe Institute of Technology (KIT)
Zweiter Gutachter:	Prof. Dr. ir. Geert Heijenk University of Twente (UT)

Zusammenfassung

Die stetige Zunahme des Verkehrsvolumens und der daraus resultierenden Unfälle und Verkehrsstaus motiviert die Forschung auf dem Gebiet der Fahrzeug-Kommunikation nun schon seit mehreren Jahrzehnten. Dabei besteht die zugrundeliegende Idee darin, dass Fahrzeuge miteinander Informationen zu ihrem Status und ihrer Umgebung austauschen. Die ausgetauschten Informationen wiederum sollen durch Fahrer-Assistenzsysteme verwertet und zu einer Steigerung der Verkehrssicherheit und der Verkehrsflusseffizienz umgesetzt werden. In den letzten Jahren wurde ein grundlegendes Verständnis der damit verbundenen technischen Aspekte erreicht, was zu den Standards IEEE 802.11p und IEEE 1609.x geführt hat sowie zu der Zielsetzung, einen Standard zu errichten, der für alle Personenfahrzeuge Anforderungen bezüglich der Kommunikationsfähigkeit definiert. Dennoch ist immer noch nicht geklärt, ob die Inter-Fahrzeug-Kommunikationstechnologie schon "fit" für sicherheitsrelevante Anwendungen ist und mit welchen Auswirkungen auf den Straßenverkehr tatsächlich gerechnet werden kann. Diese Unsicherheit zu beseitigen ist die Motivation der vorliegenden Arbeit. Es wird insbesondere eine Methode ausgearbeitet, um sicherheitsrelevante Inter-Fahrzeug-Kommunikationsanwendungen zu entwickeln und zu bewerten, und zwar am Beispiel der *Auffahr-Kollisions-Vermeidung* (Rear-End Collision Avoidance – RECA) und der *Virtuellen Ampeln* (Virtual Traffic Lights – VTL). Das Ziel der Arbeit ist es zu bestimmen, in wie weit die Anforderungen dieser Anwendungen durch die Inter-Fahrzeug-Kommunikation, speziell im Rahmen von IEEE 802.11p, erfüllt werden können, und in wie weit dies zu Sicherheit und Effizienz des Verkehrs beiträgt.

Der erste Teil der Arbeit ist der Anwendungsentwicklungsmetodik gewidmet. Im Sinne der Sicherheitsrelevanz der ausgewählten Anwendungen wird die strenge Anforderung der *fail-safety* bzw. *Fehlersicherheit* gestellt und es wird dargestellt, wie Anwendungen "fail-safe" konzipiert werden können. Dabei wird eine Anwendung als fail-safe definiert, wenn diese Mechanismen beinhaltet, welche die Folgen möglicher Fehler kompensiert. Sowohl für die RECA- als auch für die VTL-Anwendung wird für die beiden offensichtlichsten und häufigsten Fehlerquellen, nämlich der *Unzuverlässigkeit von Fahrzeug-Fahrzeug-Kommunikation* und der *Unvorhersagbarkeit des Fahrerverhaltens*, aufgezeigt, wie fail-safe-Mechanismen integriert werden können. Zusätzlich wird eine Anforderungsanalyse durchgeführt, durch welche nicht nur die erforderliche Funktionalität einer Anwendung bestimmt wird, sondern auch Indikatoren für die Kommunikation bereitgestellt werden, die anzeigen, ob und wann Information empfangen werden muss. Ein dahingehend erweitertes VTL Protokoll

wird durch *formale Modellprüfung* verifiziert. Die dazu gewählte Verifikationsmethode und die vorgegebenen Anwendungsanforderungen ermöglichen die Verifikation nicht nur bezüglich der Anwendungs- und Kommunikationsaspekte durchzuführen, sondern auch bezüglich der Fahrzeugbewegung.

Im zweiten Teil der Arbeit werden die Schwierigkeiten bei der Evaluierung einer Kommunikations-Anwendung diskutiert. Insbesondere wird dabei mit Hilfe eines Kriteriums für den Informationsbedarf, dem sogenannten *Awareness Principle*, eine Verbindung zwischen der Netzwerk- und der Anwendungsebene errichtet. Dieses Kriterium ermöglicht die Anforderungen der RECA-Anwendung in erforderliche Übertragungsparameter der Kommunikation zu überführen. In weiterer Konsequenz wird die Skalierbarkeit der IEEE 802.11p Kommunikation für *zuverlässige* Unterstützung der RECA- und VTL-Anwendung ermittelt und zugehörige Verkehrsflusseffizienzen unter Verkehrsbedingungen mit und ohne kommunikationsgestützte Anwendungen verglichen. Obwohl ein Zielkonflikt zwischen Zuverlässigkeit und Verkehrsflusseffizienz besteht, kann gezeigt werden, dass sicherheitsrelevante Anwendungen durch IEEE 802.11p zuverlässig und bei angemessener Verkehrsflusseffizienz unterstützt werden können.

Darüber hinaus wird eine *Sensitivitätsanalyse* durchgeführt, um den Einfluss von Informationsunschärfen bezüglich Netzwerk- und Funk-Kanal-Eigenschaften, die normalerweise als genau bekannt angenommen werden, auf die Leistungsfähigkeit des Kommunikationsnetzwerkes zu bestimmen. Die Bedingungen, unter welchen die größten Fehler bezüglich der geschätzten Netzwerkleistung auftreten, und die Fehler selbst werden identifiziert und quantifiziert und stehen somit zukünftig der Entwicklung von kommunikationsbasierten Anwendungen, für welche die Eingrenzung von Fehlern durch unbekannte Netzwerkeigenschaften gewünscht ist, zur Verfügung.

Obwohl noch sehr viel Entwicklungsarbeit notwendig ist, bis die ersten kommunikationsbasierten Fahr-Assistenzanwendungen kommerziell erhältlich sein werden, zeigt sich anhand der Ergebnisse dieser Arbeit, dass streng sicherheitsrelevante Anwendungen wie die Auffahr-Kollisions-Vermeidung und die virtuellen Ampeln durch die Kommunikation via IEEE 802.11p betrieben werden können und zu einem sichereren und dennoch effizienten Verkehr führen. Darüber hinaus stehen die in dieser Arbeit entwickelten Methoden zur Verfügung, um Fahrer-Assistenzanwendungen hinsichtlich unsicherer Kommunikationsrandbedingungen und unvorhersehbarem Verhalten von Verkehrsteilnehmern weiter zu entwickeln und zu evaluieren.

Abstract

The persistent increase of road traffic volume, resulting vehicle collisions and traffic congestion, have fueled the research on inter-vehicle communication networks for several decades. Vehicles equipped with communication devices are envisioned to exchange information on their own status as well as on their surroundings. The exchanged information is thought to serve as an input for various driver assistance applications that aim to improve traffic safety and traffic efficiency. After many years of research, a solid understanding of technical aspects has been achieved. This is reflected in the approval of IEEE 802.11p and IEEE 1609.x standards as well as in the decision to create a standard requiring communication capability in all passenger vehicles. However, it is still not clear whether inter-vehicle communication is "fit" for safety-critical applications and what its impact on the road traffic will be. This uncertainty motivated the work behind this thesis. In particular, we contribute by elaborating methodologies on how to design and evaluate safety-critical applications, on the example of Rear-End Collision Avoidance (RECA) and Virtual Traffic Lights (VTL). Our goal is to determine whether these applications can be supported by inter-vehicle communication, in particular IEEE 802.11p, and result in safe and efficient traffic.

The first part of this thesis addresses the challenge of application design. Due to the safety-critical nature of the chosen applications, we embrace the strict requirement of *fail-safety* and present how applications can be designed in a fail-safe manner. We define a fail-safe application as an application that integrates mechanisms to counteract the effect of possible failures. Fail-safe features to counteract the effect of the two most obvious and frequent failure sources are integrated in the design of RECA and VTL applications, namely, *unreliability of vehicular communication* and *unpredictability of driver behavior*. In addition, we perform a requirement analysis which not only determines the necessary functionality of applications but also provides indications for communication when information needs to be received. The resulting VTL protocol has been *formally verified with the model checking method*. The chosen verification approach and the defined application requirements allow us to consider not only application and communication aspects but also movement of vehicles in verification.

In the second part of this thesis we address the challenge of application evaluation. In particular, we establish a connection between network and application layers with the help of *the awareness principle*. The awareness principle allows to translate the RECA application requirements into the required transmission parameters. Consequently, we determine the scalability of IEEE 802.11p communication to *reliably* support RECA and VTL applications and compare resulting traffic efficiency to conditions

with no communication-based applications in place. Although a tradeoff between degree of reliability and traffic efficiency is quantified, it is shown that safety-critical applications can be reliably supported by IEEE 802.11p communication and result in a reasonable traffic efficiency level.

In addition, we perform *a sensitivity analysis* to determine the impact of inaccurate network and radio channel condition information on the network performance, which is assumed to be known in typical evaluations. We quantify the possible errors in estimating network performance and we identify the conditions that lead to the largest errors. This information can be used by future application designers when limiting or avoidance of errors caused by inaccurate information is particularly important.

Although a lot of work still needs to be done before first driver assistance applications will be commercially available, based on the results presented in this thesis, IEEE 802.11p communication is shown to be capable of supporting strict safety-critical applications, such as Rear-End Collision Avoidance and Virtual Traffic Lights, resulting in a safe and efficient traffic. The utilized methods can further be elaborated and used for the design and evaluation of other driver assistance applications.

Contents

List of Figures

List of Tables

1 Introduction

Throughout the whole history of transportation, the quest for safe and efficient traffic has been an inherent part of the design of roads as well as of vehicles. Modern road design incorporates multiple safety measures, e.g., roadside barriers, traffic calming constructions, speed limits, whereas adaptive traffic lights and additional lanes accommodate traffic flows for better traffic efficiency [RPM10], [UBoDotHSM+10]. Vehicle design has also progressed: features such as padded dashboards, seat belts, airbags, energy-absorbing crumple zones, as well as electric motors have made vehicles safer and more efficient. Proliferation of sensors installed in vehicles and along roads has enabled the development of various driver assistance systems. In particular, RADARs, lasers and cameras sense the environment in the direct vicinity of a vehicle or on a road segment, and various driver assistance systems use the acquired information, e.g., to notify drivers of dangerous situations, to take over vehicle control, or to dynamically adjust traffic light signaling in order to improve traffic safety and efficiency. In addition, different standards, like the Federal Motor Vehicle Safety Standards (FMSS) [Theb] and ISO 26262 [ISO11], provide minimum performance requirements for motor vehicles and their (electric/electronic) components. Automobile manufactures must adhere to these requirements to avoid unreasonable risk of crashes occurring as a result of design, construction or performance errors of vehicles.

Despite these efforts, over one million people are killed and over 50 million people are injured in road crashes each year worldwide. This is published in the World Health Organization report on road traffic injury prevention [PSS+04]. The report has estimated the global cost of road crashes to be more than US $500 billion per year. If appropriate actions are not taken, it is predicted that road traffic injuries would be the third leading contributor to the global burden of disease and injury by 2020. Furthermore, road congestion contributes not only to millions of lost travel hours, but also to high fuel consumption and increased CO_2 emissions [Into7]. The International

Transport Forum has predicted that global passenger transport volumes in 2050 could be up to 2.5 times as large as in 2010 and emissions of CO_2 could double [Int12].

The United Nation has announced 2011-2020 a decade of action for road safety [The11]. Many countries across the globe joined the initiative and are committing to take actions to ensure safer roads, safer vehicles, and safer road users. The Global Plan for the decade is based on the "safe system" approach. The approach aims to develop a road transport system that "accommodates human error and takes into consideration the vulnerability of the human body" [The11]. The approach also assumes the shift of the responsibility from road users to those who design the road transport system, including the automotive industry and legislative bodies. Similarly, many countries have made it their goal to reach zero fatalities on the roadways [WH06], [Cen08], [Com09a], [DDC14].

The recent development of vehicular communication, based on amendment IEEE 802.11p, is a promising addition to the previous efforts [Com99], [IEE12]. Two types of message exchange strategies are envisioned for vehicular communication: periodic broadcast of status messages and event-driven broadcast of important notifications. Vehicles are foreseen to communicate between each other (inter-vehicle communication) or with infrastructure (vehicle-to-infrastructure communication). Vehicular communication allows not only a larger coverage but also introduces the capability of cooperation. However, when compared to existing sensor technologies, the unreliable nature of wireless communication requires extra care when designing driver assistance systems. Extensive research over the last decades contributed to a solid understanding of the technical aspects of IEEE 802.11p, e.g., the dependencies between such factors as transmission parameters and network performance have been well understood. Yet, communication-based driver assistance *applications*, i.e., a software component of driver assistance systems, are still in the initial state of their development. Hence, the impact of vehicular communication on traffic safety and efficiency is still to be researched.

This thesis investigates whether inter-vehicle communication, based on the periodic broadcast messages via IEEE 802.11p, can support driver assistance systems that aim to improve traffic safety and efficiency. Particularly this thesis focuses on *two applications*: a traffic safety application that aims to avoid rear-end collisions and a traffic efficiency application which manages intersection crossing without the help of additional infrastructure. Although the two selected applications have different goals, both applications are safety-critical, as their failure or malfunctioning can lead to vehicle collisions and even human fatalities. Taking the safety-critical nature of these applications and the "safe system" approach into account, a strong safety requirement is imposed, in particular, a requirement for fail-safety. In addition, when developing a safe, especially a fail-safe system, care should be taken that a developed system delivers a reasonable level of traffic efficiency. As a consequence, the following general research question is formulated:

Can rear-end collision avoidance and intersection management applications be supported by inter-vehicle communication and result in safe and efficient traffic?

The answer to this question is not straightforward due to two main challenges: namely, the challenge on how to design a fail-safe communication-based application and the challenge on how to evaluate the impact such an application has on traffic. As a consequence, the two following objectives are addressed within this thesis.

Application design

The driver assistance applications that are envisioned to make use of communication are still in the initial phase of their development. The direct adaptation of the design from sensor-based systems is not always possible—the analogy applications might not exist or existing application designs are unsuitable for communication-based applications. In addition, the unreliability of vehicular communication and unpredictability of driver behavior pose major risks for the safe operation of a system. The reception of sent packets cannot be guaranteed, which depending on the traffic situation could be of crucial importance. Driver errors, especially when the driver is expected to take appropriate actions in response to system's notifications, also contribute to ineffectiveness and unsafety of the system. Due to legal restrictions, the first generation of sensor-based driver assistance systems rely on a driver for safe vehicle operation. For this reason efforts have been aimed for estimating and predicting of driver behavior, e.g., his reaction time and braking capability. Although driver behavior can be estimated it cannot be accurately predicted, e.g., a driver's reaction time can vary for the same driver depending on his emotional and physical state as well as on the traffic situation. At the very least unreliability of vehicular communication and unpredictable driver behavior should be considered in the design of a fail-safe safety-critical application. Consequently, the following detailed research question is stated:

How to design a safety-critical application that is fail-safe against unreliability of vehicular communication and unpredictable driver behavior?

The answer to this question requires reasonable design decision-making as well as elaboration of the appropriate fail-safe measures that can be integrated into the application design.

Application evaluation

The goal of application evaluation is two-fold: to determine the scalability limit of IEEE 802.11p communication to reliably support applications, and at the same time to evaluate the impact a communication-based application might have on road traffic. Traditionally, communication networks are evaluated on the network layer, which in case of driver assistance applications evaluation is not sufficient. The impact on road traffic can affect traffic efficiency as well as traffic safety and should be compared to situations when no communication-based applications are utilized. The detailed research question is formulated as follows:

How to determine the scalability of IEEE 802.11p communication to reliably support applications and to evaluate the impact of a fail-safe application on road traffic?

The answer to this question is also two-fold: first of all, the scalability limit of vehicular communications to reliably support the addressed applications has to be determined; secondly, traffic level measures should be used to determine the impact that safety-critical applications might have on road traffic.

Main contributions of this thesis

Main challenges in answering the general question whether the two applications, related to rear-end collision avoidance and intersection management, can be supported by inter-vehicle communication and result in safe and efficient traffic, lie in the area of application design and evaluation. Hence, main contributions of this thesis focus on application design and evaluation and are summarized below:

Identification of important design decisions and their integration into the design of communication-based safety-critical applications: in order to design a safety-critical application it is necessary to understand what the application should do, i.e., to understand its functional requirements. An extensive related work study has been performed in order to identify the key functions that are essential to a rear-end collision avoidance application and to an application that manages intersection crossing without the help of intersection infrastructure. As driver assistance applications face very strict safety requirements, especially those that are safety-critical, we embrace a fail-safety requirement for the applications in focus. A fail-safety application, as assumed in this thesis, is an application that integrates countermeasures in its design to neutralize the effect of possible failure sources. The fail-safety mechanisms that counteract the effect of the two most common failure sources—unreliability of wireless communication and unpredictable driver behavior—have been integrated into the design of the addressed applications. Wireless communication is known to be unreliable, especially in a highly dynamic vehicular environment. This is why applications that are based on inter-vehicle communication should be resistant against this unreliability. In addition, accounting for the "safe approach" and general proneness to errors of human drivers, the designed safety-critical applications can no longer rely purely on the driver to take appropriate actions and should consider unpredictable behavior.

Verification of application design: the important aspect for the application design is a proper design verification. Driver assistance systems act on three different levels: communication, application and movement. Application's decisions can only be discrete and are impacted by reception of communication packets which also happen in a discrete domain, but applications influence continuous movement of vehicles. In contrast to the related work when formal verification has been performed only on some parts of driver assistance systems, we address formal verification considering all three levels relevant for driver assistance systems.

Quantification of IEEE 802.11p communication scalability to reliably support the addressed applications: in order to evaluate how inter-vehicle communication can support the designed applications, the appropriate connection between network layer and application layer has to be established. A connection between network and application layers allows adjustment of transmission parameters to satisfy application

requirements rather than simply following the default settings. We establish such a connection with the help of the awareness principle. In addition, we determine the scalability of IEEE 802.11p communication to reliably support the two selected applications.

Sensitivity analysis that determines the importance of accurate network and radio channel information: we performed a sensitivity analysis to analyze which of the networking and radio channel parameters impact the network performance the most. Inaccurate information on networking parameters can lead to errors and application designers need to be aware of possible negative effects. This information can be used in future work to adjust application designs to successfully perform in the presence of inaccurate network and radio knowledge.

Quantification of the impact on the traffic: a potential drawback of applications that support traffic safety is the deterioration of traffic efficiency. The related work does not usually consider both traffic efficiency and traffic safety when certain applications are being developed. We provide a quantification of resulting traffic efficiency for the designed fail-safe driver assistance applications. The resulting traffic efficiency is compared to what can be achieved with non-communication based applications. The reliability of IEEE 802.11p communication to support applications can be seen as an indirect measure describing traffic safety.

Design and evaluation of two safety-critical applications: two safety-critical applications have been designed and evaluated based on the methods devised in this thesis: the first application is referred to as Rear-End Collision Avoidance (RECA) and the second application is referred to as Virtual Traffic Lights (VTL).

Main contributions of this thesis have been previously published in the following papers:

- Accurate knowledge of radio channel and network conditions – When does it matter? Natalya An, Jens Mittag, Felix Schmidt-Eisenlohr and Marc Torrent-Moreno. In: Proceedings of the 8th IEEE/IFIP Conference on Wireless On-demand Network Systems and Services, Bardonecchia, Italy, January 2011

- VANET: Is 95 % probability of packet reception safe? Natalya An, Tristan Gaugel and Hannes Hartenstein. In: Proceedings of the 11th International Conference on Telecommunications for Intelligent Transport Systems, Saint-Petersburg, Russia, August 2011

- Feasibility of Virtual Traffic Lights in non-line-of-sight environments. Till Neudecker, Natalya An, Ozan Tonguz, Tristan Gaugel and Jens Mittag. In: Proceedings of the 9th ACM International Workshop on VehiculAr Inter-NETworking, Systems, and Applications, Ambleside, England, June 2012

- Balancing the requirements for a zero false positive/negative Forward Collision Warning. Natalya An, Michael Maile, Daniel Jiang, Jens Mittag and Hannes Hartenstein. In: Proceedings of the 10th IEEE/IFIP Conference on Wireless On-Demand Network Systems and Services, Banff, Canada, March 2013

- Verification and evaluation of fail-safe Virtual Traffic Light applications. Till Neudecker, Natalya An and Hannes Hartenstein. In: Proceedings of the IEEE Vehicular Networking Conference, Boston, USA, December 2013
- Designing fail-safe and traffic efficient 802.11p-based Rear-End Collision Avoidance. Natalya An, Jens Mittag and Hannes Hartenstein. In: Proceedings of the IEEE Vehicular Networking Conference, Paderborn, Germany, December 2014
- Designing fail-safe and traffic efficient 802.11p-based Rear-End Collision Avoidance. Natalya An, Jens Mittag and Hannes Hartenstein. In: Elsevier Journal on Ad Hoc Networks, 2015

Overview of this thesis

This thesis is structured as follows: The topic of application design is covered in Chapter 2 which consists of two parts. The first part outlines the fundamental aspects and the related work on vehicular communication application design. The second part outlines application design decisions made in this thesis. The related work starts with an overview of different supporting technologies that are used to acquire information on the surroundings for driver assistance applications. One type of these technologies is based on in-vehicle sensors, like RADAR, camera, LiDAR, and the other is based on communication, in particular, IEEE 802.11p. The drawbacks and benefits of these two types of technologies are also described and compared with each other. Furthermore, the main design principles of safety-critical applications are discussed in the first part of the chapter. In particular, we outline the requirements to incorporate fail-safety features into the design along with the requirements posed by standard ISO 26262 [ISO11]. The aspects of the tradeoff between traffic safety and efficiency, together with the human factor are also covered. The main part of related work in Chapter 2 is dedicated to the existing application designs related to applications chosen for this thesis. We analyze the functional requirements of existing applications and how these applications address safety requirements. The generic classification for each application case is also provided. The second part of Chapter 2 explains the design decisions that are chosen in this thesis to design a communication-based safety-critical application. The chapter is concluded with an outline of assumptions made in this thesis and aspects that are left out of scope.

The application evaluation methodology is described in Chapter 3. In particular, the necessary steps are described in order to evaluate the performance of a fail-safe communication-based application. Chapter 3 starts with an overview of a typical performance evaluation process together with outlining of the open issues. In addition, Chapter 3 summarizes existing methods to evaluate sensor-based and communication-based systems. The second part of the chapter describes solutions proposed in this thesis to the previously stated open issues. Firstly, the sensitivity analysis of the most influencing factors on the networking level performance is described. This knowledge can be used by application designers in order to counteract possible operation failures due to information inaccuracy. Secondly, the awareness principle is utilized

to link network and application layers which is practical for application-aware adjusting of communication parameters. The general method to evaluate an application performance "up" to the impact on the traffic is presented at the end of the chapter.

Chapters 4 and 5 describe the complete process of design and evaluation of Rear-End Collision Avoidance and Virtual Traffic Lights applications, respectively. Finally, the last Chapter 6 provides conclusion for this thesis and indicates possible research directions for future work.

2
Basic principles of vehicular application design

The following chapter introduces essential knowledge necessary for the design of safety-critical driver assistance applications. Section 2.1 provides fundamental background, such as description of technologies that are able to support driver assistance applications, information on driver reaction behavior, the concept of road safety, and the tight interrelation of traffic safety and traffic efficiency. The corresponding challenges are also outlined. The next section, Section 2.2, provides an overview of existing application designs that address rear-end collision avoidance and intersection management. Finally, the last section evaluates existing application designs and summarizes design decisions made for this thesis. The chapter is concluded by outlining the assumptions made in this thesis and aspects left out of scope.

2.1 Fundamentals and challenges

The terminology used throughout this thesis is explained in the following: the term *system* (as in driver assistance system) refers to a general implementation of an application together with all the software and hardware components. The term *application* refers to a computer program, or a software component, that describes intended logic and behavior. Similarly, the terms *algorithm* and *protocol* refer to a stepwise procedure that is followed by an application.

The current section outlines fundamental aspects together with their challenges that are relevant for the design of driver assistance applications. Technologies such as RADAR, LiDAR, cameras, as well as vehicular communication, are capable of supporting driver assistance applications by gathering information about the en-

vironment. Their description, benefits, and drawbacks are briefly summarized in Section 2.1.1. Section 2.1.2 explores existing studies on driver behavior as a reaction to road or vehicle stimuli. In Section 2.1.3 the concept of safety applicable to driver assistance applications is elaborated. Finally, in Section 2.1.4 the tradeoff that arises when dealing with traffic safety and efficiency is described.

2.1.1 Supporting technologies

Various technologies are capable to support driver assistance applications by collecting information about the surrounding of the *host vehicle* (also called *own vehicle* or *ego vehicle*). Sensor-based technologies, such as RADAR, LiDAR, and cameras, as well as communication-based technologies, such as based on IEEE 802.11p, are considered for supporting the applications. The main advantage of sensor-based technologies is that supported applications do not require other vehicles to be equipped with similar technology. Hence, most of the driver assistance applications that exist on the market today are based on sensor-based technologies. Applications that rely on communication require other vehicles to possess communicating capability in order to operate. Communication allows not only two-way information exchange but also enables delivery of information that cannot be measured by sensors. Moreover, communication is capable of functioning in the non-line-of-sight environments and has larger coverage. Although vehicular communication brings advantages to driver assistance applications, its nature is unreliable, and compared to sensor-based technologies, vehicular communication is less researched.

Other communication technologies that can support driver assistance applications, e.g., 4G communication, LTE, WiMAX, and visible light communication [DBG+10], [VAG+11], [Vin12], [JTWL14], are left out of scope of this thesis. In the following, the basic sensor-based technologies and the IEEE 802.11p standard that defines wireless access in vehicular environment are described in detail.

Sensor-based technologies

The common classification of vehicle sensors assumes division into active and passive sensors [BBF00], [Jan05]. Active sensors, such as RADAR or LiDAR, detect objects by emitting a signal and measuring the reflection. Passive sensors, include vision-based sensors, detect objects in a non-invasive way and do not alter the environment.

RADAR, coined as an acronym for **RA**dio **D**etection **A**nd **R**anging, is a technology that can be used to detect objects, as well as to determine such parameters as objects' range and speed [Mei98], [Jan05]. Originally used for military applications, is now widely used in terrestrial, marine and aircraft transportation, as well as surveillance and astronomy. The RADAR sensor emits electro-magnetic radiation that is reflected from objects. Typically, a single antenna is used for emitting and receiving RADAR waves which requires frequent mode switching. Most automotive RADARs use frequencies in the region of 76–77 GHz, although RADARs operating at 5 GHz and 24 GHz also exist.

A laser RADAR or a LiDAR, coined from an acronym **Li**ght **D**etection **A**nd **R**anging, emits a modulated, intense source of light, and uses an optical receiver to detect

reflected signal [OMFM99], [Jan05], [RSS11]. LiDARs are capable of measuring range, range rate, and elevation of an object.

Vision sensors include one or several cameras together with a microprocessor that performs image processing [BBF00], [DMSS04], [Jan05], [NDM+09]. Typically, cameras operate in the visible light region and their capabilities are similar to that of human eyes. Infrared cameras are sensitive to heat radiation and are typically used for night vision systems.

RADARs, LiDARs and cameras have advantages and disadvantages, e.g., RADARs and LiDARs, in contrast to vision-based sensors, are insensitive to lighting [Jan05], [Wen05], [RSS11], [GS15]. Bad weather conditions may degrade performance of LiDAR and vision systems, but RADAR performance, especially if range is less than 200 m is not degraded by weather. In addition, dirt can hinder LiDARs and cameras but not RADARs. Although accuracy of a RADAR is better than 1 m, RADARs may falsely detect other reflecting objects as vehicles, e.g., traffic signs. Moreover, the problem of mutual interference for RADARs is becoming more prominent as traffic density and the number of RADAR-equipped vehicles increase [His95], [Bro07], [GBM10]. Finally, the vision systems require computationally intensive image processing algorithms. In general, detection of slow moving or stopped vehicles at long range is a limitation for all type of sensors [BC98].

Irrespective of several drawbacks, sensor-based systems are considered highly reliable and safe [GS15]. Notwithstanding, data fusion from several different sensors is often suggested in order to increase reliability and detection robustness [DMSS04], [Jan05], [NDM+09], [GS15]. More detailed information on sensor-based technologies and their characteristics can be found in [GD97], [Mei98], [BBF00], [Spe01], [Jan05], and [Wen05].

The IEEE 802.11p standard

In 1999 the Federal Communication Commission of the United States has allocated 75 MHz of spectrum for use by Dedicated Short Range Communications (DSRC) of Intelligent Transportation Systems (ITS). DSRC systems are characterized by a short range wireless link to transfer information between vehicles and roadside systems. The ITS services comprise of initiatives improving driving safety, decreasing traffic congestion, and reducing fuel consumption and air pollution [Com99]. Up until now, active research is being conducted in order to bring DSRC-based ITS services on the roads.

Cooperative systems of ITS foresee vehicle-to-vehicle (V2V), infrastructure-to-vehicle (I2V), vehicle-to-infrastructure (V2I), and infrastructure-to-infrastructure (I2I) communications for the exchange of information. Two types of broadcasting by vehicles are envisioned: periodic exchange of status information, also called Cooperative Awareness Messages (CAMs) or beacons, and event-driven broadcast of messages, also called Decentralized Environmental Notification Messages (DENMs). Various driver assistance applications are thought to profit from such information exchange to assist the driver for safer and more efficient driving.

The technical aspects behind vehicular communications are well studied and even standardized. The lower layers specifications of the DSRC are standardized by the

American Society for Testing and Materials (ASTM) in 2010 [AST10] and are based on Wireless LAN Medium Access Control and Physical Layer Specifications of IEEE 802.11a standard. In Europe, DSRC is known as IEEE 802.11p Wireless Access in Vehicular Environments (WAVE). The Wireless LAN Medium Access Control and Physical Layer Specifications for WAVE have been standardized in 2012 [IEE12]. In addition, family of IEEE 1609 standards, meant to be used in conjunction with WAVE, defines security and networking services, multi-channel operation, and more [IEE14]. In 2009, the European Commission issued an ITS harmonization mandate to achieve interoperability between various standards [Com09b].

The medium access of IEEE 802.11p deploys random access control protocols, such as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), which utilizes a "listen-before-talk" principle. Each vehicle has to detect a free channel in order to transmit its packet; if the channel is busy, the transmission has to be delayed. Just as in standard wireless networks, hidden node terminal problem can lead to packet collisions, and exposed node terminal problem to delayed transmissions.

The standard by SAE International J2735, called Dedicated Short Range Communications (DSRC) Message Set Dictionary [SAE15], and Specification of Cooperative Awareness Basic Service [AAE+14] by European Telecommunications Standards Institute (ETSI), define a message format with its data frames and elements to be used by applications utilizing vehicular communication.

The work on defining the driver assistance applications, although already in progress, is still ongoing and sometimes, left for individual manufactures [Theo5], [Inso9], [F. 11a], [HPY+14]. The SAE standard J2945 is a work-in-progress standard that defines DSRC minimum performance requirements. In particular, it defines requirements on safety awareness applications.

More and more small- and even large-scale Field Operational Tests (FOTs) are being conducted worldwide that support feasibility and validation of vehicle communication [Geno5], [Bato7], [HTBo8], [NLS+11], [DRI].

In 2014, the National Highway Traffic Safety Administration (NHTSA) of U.S. of America initiated a rule-making "that would propose to create a new Federal Motor Vehicle Safety Standard (FMVSS), FMVSS No. 150, to require vehicle-to-vehicle (V2V) communication capability for light vehicles (passenger cars and light truck vehicles (LTVs)) and to create minimum performance requirements for V2V devices and messages [...] By mandating V2V technology in all new vehicles, but not requiring specific safety applications, it is NHTSA's belief that such capability will in turn facilitate market-driven development and introduction of a variety of safety applications" [Dep14]. In light of this rule-making, it is of most importance to conduct a fundamental work on understanding the capabilities of IEEE 802.11p to support specific safety-critical applications.

The recent document "Vehicle-to-Vehicle Communications: Readiness of V2V Technology for Application" [HPY+14] assesses the readiness of V2V communication for applications. In particular, it is reported that "the safety applications enabled by V2V have proven effective in mitigating or preventing potential crashes", but it is recognized that further refinements to the prototype of safety applications are needed.

In addition, due to legal restrictions, the safety applications that are considered by NHTSA are driver warning applications which do not take vehicle control away from the driver. But as traffic safety responsibility is now being shared among drivers and road and vehicle designers, the automation becomes a promising solution to achieve traffic safety irrespectively of human fallibility.

For further information on IEEE 802.11p communication please refer to [IEE12], [HL10], [DZ12], [TSW13], and [HPY+14].

2.1.2 Driver reaction behavior

The knowledge of driver reaction behavior plays an important role in the design of driver assistance systems. Successful interaction of the system with human drivers assumes correct prediction of driver reaction to various stimuli from the application. The incorrect prediction can render the application useless and even unsafe. The knowledge of driver reaction behavior is especially relevant for applications that provide warnings and notifications, and thus rely on the driver's response to achieve their traffic safety or efficiency goals. For example, a driver assistance system that warns the driver that there is a probability of collision with the vehicle in front needs to consider the driver's reaction time and how strong he can brake when deciding on the timing to present the warning. If the warning is given too early, the driver might perceive such warning as a nuisance alert or *a false positive* (FP), if the warning is given too late, so that the driver is not able to decelerate in time and avoid the collision, the warning is called *a false negative* (FN) [AR04], [J.L07], [XQX14]. In general terms of a driver assistance system that addresses traffic safety, a false positive is an incorrect classification of a safe situation as unsafe and a false negative is an incorrect classification of a dangerous situation as safe. The driver acceptance of a driver assistance system and as a consequence the system's effectiveness, depend on the correct balance between FP and FN [KOUK97], [KLP+99], [AR04], [LP05], [JLC08], [BR11]. On one hand, if the warning is given too early, the driver can easily get annoyed and can even switch the system off. On the other hand, the frequent FN renders the system useless, as collisions would not be avoided. For this reason, estimation of driver behavior characteristics, such as *the driver's reaction time* and *the braking intensity*, is an important part of traffic safety research.

A reaction time, also called "a perception-reaction time" is the time that elapses from the occurrence of a stimulus to the action of the response to it [T.J82], [BKPP02], [BHL+13]. It has been studied using driving simulators and on test tracks where a wide distribution of driver population is faced with the same experiment—reacting to a road situation change. Typically, a normal reaction time towards unexpected natural and emergency driving scenarios are measured, e.g., the braking lights of the vehicle ahead or the traffic light switch, or the emergency braking of the vehicle ahead [JR71], [Tao89], [KMH+93], [Gre00], [Sum00], [ZAG06], [BMCR07]. Average reported reaction time varies in most studies between one and two seconds. The distribution of reaction times can be modeled with log-normal distribution with median 1.1 and dispersion parameter 0.53 [BKPP02], Gamma, or Weibull distributions [Tao89],

[Hug02]. In reality, the upper limit for reaction time is infinite, as a driver might not react at all, e.g., in case of a sickness.

A braking intensity is the deceleration capability with which a driver reacts to a deceleration stimulus [BKPP02], [KCF+03]. It is typically measured together with the reaction time and is reported to be on average equal to −0.6 g, where *g* is the g-force. The report of [BKPP02] depicts distribution of braking intensity of a drivers' population as a truncated Gaussian distribution with mean −0.6 g, standard deviation of 0.1 g, and maximum of −0.8 g and minimum of −0.3 g. Naturally, the braking intensity of a driver can be equal to zero in case a driver does not respond to the warning at all.

Although the average values represent the average driver's population well, it is not reliable for reducing or eliminating the FN, which in case of safety-related driver assistance systems is essential. An early warning for one driver, might be a late warning for the other. Moreover, the same driver might require a different reaction time depending on his physical and emotional state as well as on the current traffic situation.

The report of Knipling et al. [KMH+93] suggests using reaction time and deceleration values that are higher than the population average in order for driver assistance systems to be conservative enough to allow slow-reacting drivers to react to the system in time. Moreover, several stages of warnings, that accommodate various reaction times and braking intensities, can be used to describe different levels of driver alertness or different types of drivers [Gre00], [BMCR07]. Some approaches observe driver attention [KLTY08] or tend to recognize driver tiredness [vJKS+05]. Others consider driving styles, gender, age, and experience [JLC08] in order to better estimate driver reaction to the assistance system. These efforts allow to tailor warnings for individual driver classes. But the exact reaction behavior, for each individual driver, in each particular traffic situation, cannot be reliably estimated.

The design of driver assistance system's human-machine interface (HMI) has a direct influence on the driver's reaction to a system. The HMI should be designed in a way that the message to the driver would be quickly and unambiguously understood, and that the driver's distraction and cognitive load is kept at minimum. The interaction between the system and the driver might follow audio, visual or haptic means, which has different levels of effectiveness [COM96], [J.L07].

The "best-available human factors information" for crash warning systems as well as guidelines for interface design and driver performance are reported in [J.L07] and in the work-in-progress standard SAE J2400 called "Human Factors in Forward Collision Warning Systems: Operating Characteristics and User Interface Requirements".

2.1.3 The concept of road safety

Depending on the country, between 20 % and 30 % of all vehicle collisions account for rear-end collisions. Another 40 % are related to intersection crossing, see [KMH+93], [KOUK97], [WBMD97], [E.H10], [T. 11], [tra12], [H. 13]. Achieving road safety, by reducing collision probability, is a complex task which is not limited by integration of appropriate driver assistance systems. Road and vehicle design, driver training, as well as information security, also contribute to road safety, although left out of

scope of this thesis. Key aspects applicable to driver assistance systems that contribute to road safety are explained below in detail.

The standard ISO 26262

The standard ISO 26262 [ISO11] is the main safety standard for automotive electrical and electronic systems. The standard is adapted from a generic functional safety standard for safety-related electrical and electronic systems issued by the International Electrotechnical Commission (IEC) called IEC 61508. The ISO 26262 indicates necessary safety processes and provides guidance on how to avoid systematic and random failure risks. Provided guidance can be applied during the whole automotive life cycle, i.e., for management, development, production, operation, service and decommissioning. The major part of the standard is an automotive-specific risk-based approach to determine integrity levels of a safety-related electrical or electronic system called Automotive Safety Integrity Levels (ASIL). Based on the determined ASIL, the ISO 26262 specifies applicable measures, including requirements for validation and verification, to ensure a sufficient level of safety and avoidance of unreasonable residual risk.

The determination of the ASIL is based on a combination of three measures: *severity*, *probability of exposure* and *controllability*. Severity is the estimate of the physical injury to the health of one or more individuals that can occur in a potentially hazardous situation and can potentially cause harm due to malfunctioning behavior during the operation of a system. Severity is divided into four classes from "No injuries" to "Life-threatening injuries or fatal injuries" which are marked from S0 to S3. Probability of exposure is a probability of being in a hazardous situation coincident with a system's failure and can have five classes from "Incredible" to "High probability" which are marked from E0 to E4. Controllability is the ability to avoid a specified injury or damage through the timely reactions of the persons involved, and has four classes from "Controllable in general" to "Difficult to control or uncontrollable" and marked from C0 to C3.

When the three measures, severity, probability of exposure and controllability, are determined, the ASIL (A, B, C, or D) can be established based on Table 4 provided in Part 3 of ISO 26262 [ISO11]. The rest of the standard indicates methods to ensure functional safety, with the appropriate recommendation degree, based on the specific ASIL. The methods cover the system, hardware, software and process levels, and range from design analysis and verification to testing. In addition, Table 7 of Part 8 [ISO11] provides limits for observable failure occurrence rate for each ASIL. Thus, the ASIL A should have less than 10^{-7} errors per hour of operation, the ASIL B less than $< 10^{-8}$ errors per hour of operation, the ASIL C less than 10^{-8} errors per hour of operation, and the ASIL D less than 10^{-9} error per operation hour.

The systems with ASIL D, to which safety-critical driver assistance applications can be assigned, have a strict requirement for validation and verification. Through validation and verification the majority of potential failure causes can be eliminated, especially on the software level. Validation and verification are procedures that involve checking that the system (or its components) meets the specifications and requirements as well as fulfills the intended purpose [oEE11], [BF14]. Validation deter-

mines whether the right system is being built and is often referred to as a high-level checking. Verification determines whether the system is being built correctly and can utilize mathematical methods.

The fail-safety property

The system can be verified to be correct and its components highly reliable, but the traffic can still be unsafe if the system's initial *functional requirements* are incorrect or incomplete to begin with. Moreover, new technologies introduced in the automotive industry induce new failure modes, especially those that involve human interaction. Rather than only learning from occurred failures or making the system's components more reliable, it should be aimed to predict failures before they occur and build safety into the system design [Lev00], [GPSV06]. In other words, safe-critical driver assistance systems should be designed in a *fail-safe* manner.

Definition 2.1. ***A fail-safe*** *system incorporates features for automatically counteracting the effect of an anticipated possible source of failure* [1].

In the context of this thesis, a fail-safe property is central for the design of safety-critical driver assistance applications.

Verification

Verification methods include review, walk-through, inspection, formal verification, simulation, engineering analysis, demonstration, and testing [ISO11], [BK08], [oEE11], [BF14]. The authors of [Per99] underline the importance of ensuring safety not only against failures of individual components but also against failures that arise because of interactions among different components, e.g., electromechanical, digital, and human. Communication-based driver assistance applications represent interaction in the following three domains:

- **Application:** represents the algorithm running on each vehicle, typically modeled as a state machine, hence operating on discrete values, states, and events;
- **Communication:** represents the (imperfect) information exchange among vehicles that is used by application for its decisions, modeled by discrete packet reception, reception probabilities, etc.;
- **Movement:** represents continuous movement of the vehicle influenced by application, dealing with time, speed, acceleration, etc.

For such hybrid systems, formal verification without appropriate abstraction might be infeasible, and verification with simulation or demonstration might be a better solution.

[1] according to Merriam-Webster dictionary.

Dependability

Dependability of a computing system, i.e., the ability to deliver service that can be justifiably trusted, comprises safety as an "absence of catastrophic consequences on the user(s) and the environment" along with availability, reliability, confidentiality, integrity and maintainability [ALR00]. The means to attain dependability also include fault forecasting together with fault prevention, fault tolerance, and fault removal, which includes mentioned earlier validation and verification techniques. Avižienis et al. in [ALR00] describe the *fault-error-failure dependency* that causes threats to dependability of a system: "A system may fail either because it does not comply with the specification, or because the specification did not adequately describe its function. An error is that part of the system state that may cause a subsequent failure: a failure occurs when an error reaches the service interface and alters the service. A fault is the adjudged or hypothesized cause of an error. A fault is active when it produces an error; otherwise it is dormant." In such a way, a lost broadcast packet can be seen as a fault, but it does not necessarily cause an error in the system or a general failure of it; in other words, it does not cause traffic to become unsafe. For a system to fail, other faults and errors should happen at the same time, e.g., a driver gets distracted, or a hardware or software component fails. A dependable system does not fail, even in the face of faults and errors. It is a good practice for safety-critical systems to be designed accounting for dependability.

More details on system safety and dependability can be found in [Lev95], [Lev00], [ALR00], and [Vin14].

2.1.4 Tradeoff between traffic safety and efficiency

Maximizing traffic safety and traffic efficiency at the same time is a non-trivial task—to achieve safe traffic the vehicles should drive slower and with large inter-vehicle distances which automatically reduces traffic efficiency. To achieve efficient traffic, the vehicles should be moving faster and closer to each other, which potentially undermines traffic safety. To achieve perfect traffic safety, vehicles must stand still, which makes the whole concept of transportation useless [Ker04]. The Swedish project Vision Zero [TH99] describes an ethical approach to traffic safety and efficiency, where the vision is that "eventually no one will be killed or seriously injured within the road transport system". In other words, the aim for the traffic safety is not a zero vehicle collision rate but rather no serious human injuries. In addition, the traffic efficiency cannot be obtained at the expense of traffic safety but follows out of traffic safety, e.g., the safer road infrastructure can afford higher speed limits. With responsibility sharing between road users and with designers of different transportation system components, like roads, vehicles, and driver assistance systems, the challenge is to design not only a system that is safe but that delivers reasonable level of efficiency.

2.2 Related work

In the current section related work on various designs of the applications related to rear-end collisions and intersection management is analyzed. The goal is not to present an exhaustive list of all existing applications but rather to systematize the most relevant works according to their prominent functional requirements as well as safety-related features. In the end of the section a classification is given in a tabular form with major functional and safety features.

2.2.1 Applications related to rear-end collision avoidance

Multiple applications deal with rear-end collisions in one way or another. In this section we identify the relationship between various applications depending on their objectives, which is followed by a separate description of each of the application. These applications are either based on sensors and might already exist as a ready-to-buy product, or communication is envisioned to support them, and such applications are still under active development and research. The special focus is then laid on applications that actively mitigate or avoid rear-end collisions. The applications' functional requirements, i.e., what applications should do, along with how safety aspects are approached in their design are highlighted.

Overview

Several driver assistance applications address or are related to rear-end collisions in one degree or another. Those are: Adaptive Cruise Control (ACC), Platooning, Forward Collision Warning (FCW), Rear-End Collision Avoidance (RECA), and Emergency Electronic Brake Lights (EEBL). ACC is an application that automatically adjusts vehicle's speed to maintain a safe distance to a vehicle ahead, thus relieving a driver from frequent alternating between braking and acceleration. Platooning is an approach to group moving vehicles in order to decrease the distance between them, thus increasing the road capacity. As opposite to optimizing driving comfort or traffic efficiency, FCW and RECA applications aim to improve traffic safety by mitigating or avoiding rear-end collisions. FCW provides a warning to a driver in case of a potential rear-end collision, and RECA additionally includes an automatic braking in case a driver did not respond sufficiently. The automatic braking of RECA is typically preceded by a warning, as in FCW. Both ACC and Platooning are susceptible to rear-end collisions and, as a result, are often combined with FCW or RECA applications. All these applications share similar hardware equipment, e.g., a sensor to measure the distance to the vehicle ahead, and a Global Navigation Satellite system (GNSS) receiver to obtain own location data. The first generation RADAR-based ACC, FCW, and RECA applications are already sold under various trade names and variations by different automotive manufactures. An EEBL application is envisioned to wirelessly broadcast information in case a host vehicle performs a sudden braking. The reception of such information, if found appropriate, triggers actions from FCW or RECA applications at the receiving vehicles.

The relationship between these applications are described in several works, e.g.,

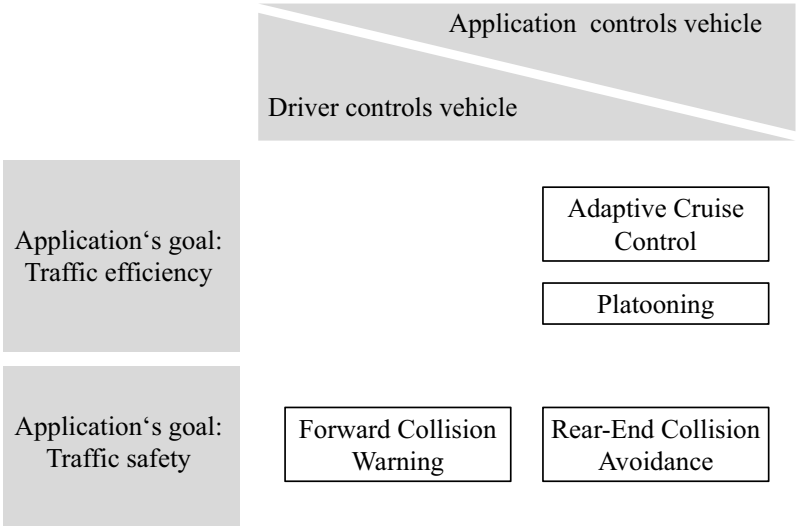


Figure 2.1: Overview of applications related to rear-end collisions

[WBMD97], [WBA98], [ISO13b]. The work of [WBMD97] categorizes ACC, FCW, and RECA as variations of a forward-looking collision warning system. In 1998, the authors of [WBA98] presented a "Collision Avoidance System Evolution", which starts with Cruise Control (a forerunner of ACC), proceeded to FCW and towards Collision Avoidance Systems (same as RECA). The further into the future the more complexity was envisioned with more vehicle control taken away from the driver and given to the system.

In Figure 2.1 we graphically arrange ACC, Platooning, FCW, and RECA applications based on *the degree of the vehicle control* which is either given to the driver or to the application versus *the application's goal*, which is either to improve traffic efficiency (or driver's comfort) or to improve traffic safety. Here, vehicle control implies throttle and/or brakes control. The more control is given to the application, the less vehicle control is left to the driver. As seen in the Figure 2.1, the FCW application expects the most involvement from a driver as it relies on the driver's response to a warning. Whereas ACC, Platooning, and RECA, although pursuing different objectives, provide the most control over the vehicle to the application. In the following these applications are discussed in greater detail.

Adaptive Cruise Control – The predecessor of an ACC is a simple Cruise Control system[2] which was first fitted to vehicles as early as in 1900. The Cruise Control system is intended for the use during long-time driving on roads with sparse traffic density and is essentially a speed control. The Cruise Control provides to the driver additional comfort of not needing to press the acceleration pedal all the time while maintaining a constant speed [CFKV15]. The Adaptive Cruise Control relies on in-vehicle sensors to recognize an upstream moving vehicle (also called leading vehicle) and to derive the distance to that vehicle. The ACC adjusts the speed of the own vehicle (also called following vehicle) automatically in order to maintain a safe time

[2] http://www.carhistory4u.com/the-last-100-years/parts-of-the-car/cruise-control

headway and/or a small relative speed, thus making human intervention unnecessary [Raj12]. Therefore, characteristics of the human driver are relatively unimportant for the ACC. Other names of the ACC include an autonomous cruise control system or a RADAR cruise control.

The standard ISO 15622 [ISO10] describes basic functionality and performance requirements of Adaptive Cruise Control. According to ISO 15622 "the goal of ACC is a partial automation of the longitudinal vehicle control and the reduction of the workload of the driver with the aim of supporting and relieving the driver in a convenient manner". The ISO standard describes two types of ACC—ACC type 1 and ACC type 2. The ACC type 2 performs "active brake intervention with a clutch pedal". Although ACC possesses capabilities to automatically decelerate if the distance to the leading vehicle is decreasing, the deceleration is limited (shall not exceed 3.5 m/s^2 averaged over 2 s) and might not be enough to avoid a rear-end collision. In addition, the ISO 15622 describes ACC system reactions to failures depending on which subsystem fails. All failures shall result in immediate notification of the driver, the engine control shall be relinquished, and the driver shall always be able to override ACC control signals. The limitations of ACC, e.g., that ACC is not required to respond to stationary objects, should be clearly communicated to the driver via the manual or caution labels.

As conventional ACC is not intended to operate at lower speeds, a Full Speed Range Adaptive Cruise Control (FSRA), standardized as ISO 22179 [ISO09], was developed to function at all vehicle speeds from start to stop. The main functions and reactions to failures are the same as those of ACC. In contrast to ACC, which is intended to function under free-flowing traffic conditions, FSRA is also intended to operate in congested traffic conditions and is required to track a stopping leading vehicle to a full stop.

ACC systems mostly rely on RADAR sensors, with some systems utilizing LiDARs or cameras [CFKV15], [Raj12]. A Cooperative Adaptive Cruise Control (CACC) is an extension of ACC where additional information, such as an acceleration of the leading vehicle, is acquired via wireless communication [GdSMH01], [dBKvKN04], [vAvDV06], [PSvN+11], [LvEW+11], [KSdPM12], and [MSS+14]. The major benefit of CACC over ACC is a string stability introduced by the CACC system. Without string stability the oscillations which are introduced by braking and accelerating vehicles can be amplified and eventually lead to traffic jams or even to rear-end collisions. CACC increases traffic efficiency by allowing smaller gaps between vehicles, as opposed to ACC that requires a minimum time headway of 1 s [ISO10]. CACC application is envisioned not only for automated following of one vehicle by another, as in the case of ACC, but even larger benefit with respect to traffic efficiency is foreseen for larger number of vehicles that follow each other in a group or in a platoon [PSvN+11].

Platooning – A concept similar to CACC called Platooning has been a topic of research for many decades already. The standard SAE J2945/6 (currently under development) is to define the differences between Platooning and CACC, as well as the necessary communication data exchange which is required to coordinate vehicle maneuvers. With Platooning vehicles are thought to be grouped into platoons in order to increase the road's capacity and the driver's comfort through automation.

First longitudinal vehicle control algorithms, that are necessary to control spacing between vehicles and their speed, have been published starting 1970's [CG78], [Shl78], and [HMNS91]. Starting 1990's, first demonstrations took place proving the technical feasibility of automated driving with Platooning [Mil92], [TJP97], and [RCC10]. Platooning typically requires installation of magnetic markers on the road to facilitate vehicle tracking and to guide the steering, whereas RADARs are utilized to track the distance to the following and leading vehicles [TJP97]. Additionally, the usage of wireless communication to control platoons has been increasing with the progressing development of communication itself [TJP97], [RCC10]. Platoons promise not only to increase traffic efficiency and relieve the drivers but also to save fuel, due to reduced aerodynamic drag force for the vehicles in the platoon [BM00], [RCC10].

ACC and Platooning systems, although control the vehicle's braking to reduce the need for the driver intervention, are only designed to increase driver's comfort and convenience but not to prevent collisions, especially those that result due to sudden braking. For an active collision prevention a system that relies on human follow-up actions, like Forward Collision Warning, or a system with automatic collision avoidance features, like Rear-End Collision Avoidance, is required [BC98], [LP05].

Forward Collision Warning – A Forward Collision Warning system is a pre-crash safety system that warns the driver if there is a possibility of a collision with a preceding vehicle due to either too high relative speed or too small inter-vehicle distance [KLP+99]. The main functional requirement of the FCW system is to support the driver with a timely warning so that the driver could react, by decelerating or changing the lane. Thus, the safety benefit of the FCW system depends on the adequate response of the driver, which depends on driver reaction behavior, i.e., his reaction time and braking intensity, see Section 2.1.2. As accommodating a wide range of human characteristics is non-trivial, the mitigation of a rear-end collision severity is also considered to be a benefit of FCW.

The first document containing preliminary guidelines for an FCW system has been published by SAE International in 1997 [WBMD97]. Later, in 2002 and 2013 the International Organization for Standardization published the standard ISO 15623 [ISO13a] describing an FCW system. The SAE paper [WBMD97] presents preliminary guidelines for a forward-looking collision warning system that aims to mitigate or eliminate rear-end collisions by presenting a warning to the driver. The major system performance requirements are addressed in the document, whereas detailed implementation decisions are left to the manufactures. The guidelines include system limitations, operation ranges, electromagnetic safety requirements, driver interface requirements, as well as qualification tests to verify correct operation of the system. The SAE paper defines two types of warnings: the first warning is addressing an inattentive driver situation and the second warning is addressing a following-too-closely situation. *A warning distance* is calculated and compared with the distance to the vehicle in front and if exceeded, the driver is warned. A limited automatic braking is utilized as a warning, rather than as a rear-end collision avoidance feature. Additionally, a "push-back" accelerator pedal warning is foreseen that "resists the force placed by the driver on the accelerator pedal". The "push-back" accelerator pedal

warning is also intended to only inform the driver of following-too-closely to the vehicle in front and does not prevent the driver from overriding the "push-back".

The standard ISO 15623 [ISO13a] describes performance requirements and test procedures for an FCW system, called a Forward Vehicle Collision Warning System (FVCWS), which is based on a RADAR technology. The purpose of the FVCWS "is to inform the driver of the need to take action in order to avoid or reduce the severity of a possible imminent rear-end collision". The warnings should be provided in a timely manner, such that drivers avoid most common rear-end crashes by applying brakes only. Hence, the FVCWS do not overtake vehicle control to mitigate the crash. Although optional *warning braking* is foreseen, it should last less than 1 s and should not result in a speed reduction exceeding 2 m/s, and thus does not necessarily serve as a collision avoidance feature. The standard describes general design guidelines, operational limits, warning types and warning triggers, conditions when a warning is not needed, as well as reactions to failures. Implementation details are left to the manufactures. The FVCWS is required to provide warnings for moving obstacle vehicles, including those that have been detected as moving by the sensor and now stopped, but warning of stationary objects is optional. The system's limitations are required to be clearly stated in the manual or described with the use of caution labels. As the driver always remains responsible for the safe operation of the vehicle, the driver should be informed if a fault has been detected. The warning is issued if the deceleration required by a following vehicle to avoid a rear-end collision with a leading vehicle exceeds a threshold of 0.68 g. The error of the warning distance is required to be at most ±2 m or ±15 %. The required deceleration calculation considers the driver reaction time, which should not be less than 0.8 s.

In the following, all applications that utilize warnings to notify the driver about possibility of a rear-end collision are classified as FCW applications.

Rear-End Collision Avoidance – A Rear-End Collision Avoidance application, also called Forward Collision Avoidance (FCA) and Automatic Emergency Braking System (AEBS) [CFKV15]), is an advanced version of the FCW system with additional automatic braking capability in case the driver fails to respond to the warning. The standard ISO 22839 [ISO13b] describes a Forward Vehicle Collision Mitigation System (FVCMS) which is an "extension" of ISO 15623. The FVCMS automatically brakes to reduce the relative speed if the likelihood of a rear-end collision is judged to be high. The automatic braking is preceded by a warning, which is provided in accordance with ISO 15623 [ISO13a]. The main functionality, general operational limits, performance, and validation requirements are outlined in ISO 22839; although implementation details are left to the manufactures. The allowed average deceleration values are defined in the document: the average deceleration should not exceed 4.0 m/s² when a FVCMS first initiated and can increase up to 6.0 m/s² (averaged over 1 s). In contrast to FCW which is a collision mitigation application, RECA is a collision avoidance application. As in ISO 15623, the driver should be immediately informed in case of a failure. Whereas faulty interventions are tolerated in the case of FCW, the RECA application has a lower fault tolerance as it actively overtakes vehicle control [Jan05].

Similarly, the EU Regulation No. 347/2012 [Com12] specifies the requirements and

testing procedures for advanced emergency braking system (AEBS) that detect a rear-end collision possibility. AEBS is thought to warn the driver and to automatically brake if the driver takes no action.

Authors of [MPGO12] suggest that an automatic braking should be "a last resort" and a rear-end collision avoidance via steering should be preferred. Rear-end collision avoidance through steering is left out of scope of this thesis since additional knowledge about the situation on the adjacent lanes is require.

In the following, all applications that engage automatic braking to avoid rear-end collisions are classified as the RECA applications.

Emergency Electronic Brake Lights – A communication-based application envisioned to address rear-end collisions is called Emergency Electronic Brake Lights (EEBL), or Emergency Warning Message (EWM), where a vehicle is broadcasting a self-generated EEBL event information to the surrounding vehicles [Ins09], [YLVF04], and [SC13]. The EEBL application is envisioned to benefit not only the immediate neighbors of the broadcasting vehicle but also those vehicles to whom the line-of-sight is obstructed due to, e.g., another vehicle or weather conditions. If a vehicle receives an EEBL message from a vehicle in front, the FCW or RECA applications would be activated. As the EEBL application results in the activation of the FCW or RECA applications, it will not be further discussed.

The FCW and the RECA applications are activated automatically, whereas the ACC system should be turned on by a driver [WBMD97]. The FCW or the RECA systems can be combined with the ACC, CACC, or Platooning to mitigate the severity or to avoid possible rear-end collisions [GdSMH01], [Gen05], [AKG06], [MMY09].

Application requirements

In the following, existing applications related to FCW and RECA are discussed in detail, as these applications actively aim to mitigate and avoid rear-end collisions. Although existing application approaches are mainly based on sensor technologies, it is essential to study their design prior to designing an application that is purely based on wireless communication. Subsequently, the focus is put on the functional requirements of FCW and RECA, i.e., what FCW and RECA applications should do.

The main requirement for an FCW and an RECA application is to timely warn the driver of the host vehicle, also called the following vehicle (FV), of a potential rear-end collision with the preceding vehicle, also called the leading vehicle (LV). Additionally, the RECA system is required to automatically brake in case the driver fails to adequately respond to the warning. The decision to warn or to brake is taken based on information about the preceding vehicle – which is received with the help of either on-board sensors or over wireless communication, on the information about host vehicle – which is available via the vehicle's CAN-bus (Controller Area Network), and on the predefined algorithm – which varies from one manufacture or research institution to another. The differences between gaining information via sensors or communication, together with the drawbacks and benefits, are discussed in Section 2.1.1. The information on the host vehicle is assumed to be available and error-free, see Section 2.3.3. Thus the algorithms according to which a decision to

warn or to automatically brake is taken, or more precisely, the algorithm's triggering conditions are of a particular interest.

The algorithms of FCW or RECA applications define triggering conditions that are based primarily on *distance* or *time*. Some authors [Ati10], [BR11], and [BLM01] classify these algorithms as based on *kinematic approach* (same as based on distance) or as based on *perceptual approach* (same as based on time). In the distance-based approach a kinematic distance, that is required by a following vehicle to adjust its speed in order to avoid the rear-end collision, is calculated. In the time-based approach a time metric, typically a time-to-collision (TTC), is calculated. A warning or an automatic braking is triggered when a following vehicle crosses the kinematic distance or a TTC falls below a predefined threshold, thus making the alerting mechanism a discrete decision process.

Distance-based approach – In the distance-based approach a distance D is calculated, which is required by a following vehicle to adjust its speed (to decelerate) and to come to the same speed as a leading vehicle without colliding with it. A distance at which a driver gets a warning is called a warning distance [KMH+93], [ISO13a]. The warning distance considers human factors or driver reaction characteristics like the driver's reaction time t_R and the braking intensity a_F. In such a way the driver has enough time to react and to start decelerating in order to achieve the same speed as the leading vehicle with whom the rear-end collision is predicted. Thus, whether a warning would be successful in alarming a driver to take actions, depends on the estimation of driver's reaction characteristics. The RECA application additionally calculates a distance, called *an automatic braking distance*, where an automatic braking should be triggered. The braking intensity is often set to a constant but physically feasible value. The driver's reaction time does not need to be considered since the system acts automatically.

The calculation of both kinematic distances is dependent on the speed of the leading v_L and the following vehicles v_F (naturally, valid if the speed of the following vehicle is larger than the speed of the leading vehicle $v_F > v_L$), as well as on the acceleration of the leading vehicle a_L. If the acceleration of the leading vehicle is zero, the leading vehicle is either at stop or moving with the constant speed. According to kinematic motion laws, the calculation of distance D is performed as following:

$$D = \frac{(v_F - v_L)^2}{-2a_F} + (v_F - v_L) \cdot t_R \tag{2.1}$$

as also stated in [KMH+93], [BCM+98], and [ISO13a].

A system and brake delay can be added to the reaction time and a safety gap can be added to the overall distance D to account for the system delay and to include a safety buffer between the following and the leading vehicle at the end of the maneuver.

If the acceleration of the leading vehicle is negative ($a_L < 0$), the leading vehicle is decelerating, and the calculation of the distance D is performed as following:

$$D = \left(\frac{v_F^2}{-2a_F} + v_F \cdot t_R\right) - \frac{v_L^2}{-2a_L} \tag{2.2}$$

where $v_F > v_L$, $a_F < 0$, $a_L < 0$, as also reported in [KMH+93], [WBMD97], [BCM+98], and [ISO13a].

Time-based approach – Algorithms that follow perceptual approach typically use a time-to-collision metric as an indicator for collision risk. The TTC is the "time required for two vehicles to collide if they continue at their present speed and on the same path" [Hay71]. The warning is triggered when the TTC falls below a certain threshold. Although the TTC is "frequently used in literature as a descriptor of how urgent a situation has become, as well as potentially how a driver perceives stimuli during an event", choosing the right threshold is a very challenging task [MHDK09], [TY10]. Most commonly, e.g., in the works of [vdH90], [MHDK09], and [ISO13a], the TTC is calculated based on an inter-vehicle distance or a relative distance between two vehicles Δr divided by a relative speed Δv between both involved vehicles:

$$TTC = \frac{\Delta r}{\Delta v} \tag{2.3}$$

Typically, an analysis of accident situations or measuring the test drivers' reaction to a danger allows determining the minimum TTC to use as a threshold for differentiating collision risk and safe situations [vdH90], [BC98], [SM03], [MMY09], [KG11]. The reported values range between 1 s to 4 s. The driver's reaction characteristics, like driver's reaction time and braking intensity, are considered in TTC only indirectly.

The major weakness of using the TTC is the assumption of constant velocities during the pre-collision phase. Some modifications have been proposed that relax the assumption of constant velocity and acceleration, e.g., an enhanced TTC by [ISO13a], an inverse TTC by [KCF+03], [KLF05], and [MMY09], or a Time-to-Last-Second-Braking by [ZAG06].

Both approaches, the distance- or the time-based can be calculated irrespective of each other and from the same set of input data. For some algorithms, e.g., [KLP+99], [LMZW08], [Ati10], and [ISO13a], a warning is triggered when the **deceleration** required to avoid the accident crosses a specified threshold. This approach is similar to the distance-based approach as it utilizes the same kinematic motion laws to calculate the required deceleration. It is also similar to the time-based approach as crossing of some predefined threshold triggers the warning. The approaches that calculate the required deceleration are treated here as distance-based approaches.

Application requirements on receiving information

In addition to functional requirements there are also requirements that applications have on receiving information. Sensor-based applications sample information on the environment with a very high frequency creating almost continuous feeding of information. Requirements on information for communication-based applications is typically expressed as required update frequency of 1–10 Hz, communication range of 100–300 m, and allowable message latency of 100–300 ms for various applications [Theo5], [Ins09].

Considered driver reaction behavior

A warning triggered by an FCW or an RECA application anticipates the driver response. Hence, a prediction or an estimation of the driver's reaction behavior is necessary when deciding on the timing of a warning. The application designs that utilize distance-based approach to define the warning triggering conditions make use of studies that estimate drivers reactions to road stimuli under various kinematic situations. Typically, a reaction time distribution and a braking intensity distribution are utilized, as summarized in Section 2.1.2. The algorithm designers integrate either mean values of distributions to address an average driver or use graded warnings, e.g., a warning at 95th and 75th percentile of the drivers' reaction time or braking intensity distributions, which allows the warning to be beneficial for a wider range of drivers [WBMD97], [SSH98], [BKPP02], [LHH04]. The driver is sometimes allowed to perform limited adjusting of the warning timing. The applications that rely on the time-based approach to determine the warning timing also make use of human studies. Such human driver studies record timestamps, when drivers react to specific driving situations. Thus, driver's reaction time and braking intensity is accounted implicitly. Typically, the mean TTC is used as a threshold when a warning should be triggered.

Integrated fail-safety features

Even if the warning timing has been accurately calculated, the driver can still fail to adequately respond to the warning due to inattention or even a health condition. Many applications integrate an automatic braking as a fail-safety feature against inadequate reaction from a driver. Thus, the RECA application is a fail-safe version of the FCW application, at least, with respect to failures in estimating drivers' reactions. In addition, typical fail-safe features of most of sensor-based FCW and RECA applications include notifying the driver about a failure.

Verification methods

The importance of the design verification, especially for safety-critical applications like an FCW or an RECA, has been discussed in Section 2.1.3. Many project reports that document development of an application addressing rear-end collisions as well as standardization documentations include detailed *verification or qualification tests*. These tests are used to test the designed application for correct operation behavior, see e.g., [WBMD97] and [ISO13a]. The verification tests include description of the most typical rear-end collision scenarios which are based on accident statistics gathered around the world. Verification can then be performed either via a computer simulation, a test in a driving simulator, or on the test tracks with mock vehicles [WBMD97], [BCM+98]. Some works do not report any kind of verification for the designed applications, but high automotive safety standards (e.g., ASIL, see Section 2.1.3) would require verification, at latest, during the implementation phase. The formal verification, as discussed in Section 2.1.3, is not performed in the studied literature.

Applications classification

In the following, a summary and classification of the reviewed literature on FCW and RECA applications is provided in a tabular form. Other applications (e.g., ACC

or Platooning) are mentioned only if combined with FCW or RECA. Applications that focus on early detection of a rear-end obstacle followed by a modification of a route plan to change lanes e.g., [CHCP08], [MPGO12], are not included. Algorithms that address acceleration of the leading vehicle for the sake of avoiding another vehicle driving into it, e.g., [CC09], [CGM12] are also out of scope. Although several decades of research is summarized here, the provided summary does not claim completeness. Other surveys, e.g., [vdH90], [SSH98], [KLP+99], [SM03], [Jan05], [MHDK09], [Ati10], [BR11], provide more complete and detailed information.

Table 2.2.1 summarizes reviewed rear-end collision avoidance applications listed chronologically. In the *Reference* column corresponding references are provided. In the *Application* column the type of the application is specifically outlined, e.g., whether it is an FCW or an RECA application. Whether an application is based on the distance or time approach is specified in the *Approach* column. Which technology is supporting the application is noted in the *Supporting technology* column. The *Driver characteristics* column summarizes how driver behavior is accounted in the application design. The last two columns summarize how reviewed applications accommodate the functional safety outlined in Section 2.1.3. The *Fail-safety* column indicates design features that counteract possible failures, e.g., inadequate response of the drivers. Finally, the *Verification* column outlines how an application is verified to be safe and to correctly operate.

Reference	Application	Approach	Supporting technology	Driver characteristics	Fail-safety	Verification
[KMH+93]	FCW	Distance-based	RADAR	Fixed values	N/A	N/A
[DBN+94]	RECA	Distance-based	RADAR	Fixed values	Automatic braking if driver does not respond	N/A
[AYHI96]	RECA	Distance-based	RADAR, camera	Fuzzy reasoning, as in [IHN95]	Automatic braking if driver does not respond	N/A
[WBMD97]	FCW	Distance-based	Technology independent	Fixed values	Driver notification of a failure and limitations	Qualification tests
[BCM+98]	FCW	Distance-based	RADAR	Fixed values	N/A	Simulations in a driving simulator
[SSH98]	RECA	Distance-based	N/A	Limited adjusting is possible	Automatic braking if driver does not respond	N/A
[BC98]	RECA	Time-based	RADAR	Implicit	Automatic braking if driver does not respond	N/A
[KLP+99]	FCW	Deceleration-based	Laser, RADAR, communication	Limited adjusting is possible	Driver notification	Objective test procedures
[GdSMH01]	FCW + CACC	N/A	IEEE 802.11p	N/A	CACC switches to ACC in the event of communication failure	N/A
[BKPP02]	FCW	Distance-based	RADAR	Limited adjusting is possible	N/A	Verification tests with the prototype
[DMSS04]	FCW	Time-based	Camera	Limited adjusting is possible	N/A	Tests on test tracks, collision speed reduction
[Gen05]	FCW + ACC	Distance-based	RADAR	Limited adjusting is possible	Driver notification	Prototype verification on component, subsystem, and system level
[KCF+05]	FCW	Time-based	RADAR	Limited adjusting is possible	N/A	Test track tests

Reference	Application	Approach	Supporting technology	Driver characteristics	Fail-safety	Verification
[AKG06]	FCW + ACC	Distance-based	RADAR	Limited adjusting is possible	N/A	Simulation, collisions are not avoided if ACC is off and LV brakes hard
[ZAG06]	RECA	Time-based	N/A	Implicit	Automatic braking if driver does not respond	N/A
[LBT+08]	FCW	Time-based	RADAR	Implicit	N/A	Field operational test
[LMZW08]	FCW	Deceleration-based	RADAR, LiDAR	Fixed values	N/A	N/A
[MMY09]	FSRA + RECA	Distance-based and Time-based	RADAR	Fixed values	Automatic braking if driver does not respond	Real vehicle tests
[TY10]	FCW	Time-based	IEEE 802.11p	Limited adjusting is possible	Compensation for communication delays	N/A
[BR11]	FCW	Distance-based	N/A	Fixed values	N/A	N/A
[ISO13a]	FCW	Distance-based	RADAR	Fixed values	Driver notification	Driving tests
[XQX14]	RECA	Distance-based	IEEE 802.11p	Fixed values	Automatic braking if driver does not respond; GNSS errors compensation	Simulation

Table 2.1: Classification of applications addressing rear-end collisions.
N/A – information is not available

2.2.2 Applications related to intersection management

An intersection represents a shared area where two or more roads meet or cross [RPM10]. Since the shared area can be occupied by vehicles traveling on different roads, their paths can cross and a mechanisms is required to manage the crossing order of vehicles. Currently, two types of intersection management approaches exist: traffic lights and traffic signs. Traffic lights are typically installed on large and busy intersections, whereas traffic signs regulate smaller intersections with less traffic flow [RPM10]. Irrespective of the intersection management approach, drivers should be able to safely cross an intersection as long as they all abide to the intersection crossing *protocol* (e.g., green light means "go", red light means "stop", and "right before left" ordering). Intersection management is required not only to allow safe intersection crossing but also to optimize and increase traffic efficiency.

In the following, first an overview is given that relates different types of applications that assist drivers in intersection crossing. Most of these applications are envisioned to be supported by wireless communication because of the non-line-of-sight conditions that occur on intersections under which sensor technologies reach their operational limits. Intersection management applications can be divided into those that are intended for intersections that are governed by traffic lights and those that are intended for intersections with traffic signs. Distinguishing of applications can also be done based on their goals—to improve traffic efficiency or traffic safety. At the end of the section, applications that manage intersection crossing on intersections governed by traffic signs are discussed in more detail, as outlined in Chapter 1. In particular, the related work is surveyed to examine how functional and safety requirements are addressed by proposed applications. A summary is given in a tabular form at the end of this section.

Overview

Figure 2.2 depicts an overview of driver assistance applications related to intersection crossing. Most of these applications are only envisioned and do not exist on the roads, as the intersection's non-line-of-sight environment is very challenging for sensors, whereas communication technology is still not on the market. Depicted applications are arranged according to *their primary goal* – whether it is improvement of traffic safety or traffic efficiency, and according to which *intersection management approach* is already in place – whether intersections are governed by traditional traffic lights or traffic signs.

Traditional traffic lights utilize fixed phase switching, which is not always efficient with respect to the actual traffic situation. There are multiple approaches in the literature that use additional information from sensors, e.g., induction loops, pedestrian pressing a crossing signal, and communication, to optimize traffic light switching in order to dynamically accommodate current traffic loads [GGD+07], [MBML11], [A. 13]. Optimized traffic lights are also called *Adaptive Traffic Lights*, and are depicted in Figure 2.2 as striving for both, traffic efficiency and traffic safety. A *Red Light Running* (RLR) [O. 09] application is envisioned to improve traffic safety for the cases when vehicles enter an intersection after the signal has turned red. Just as the Adaptive Traffic

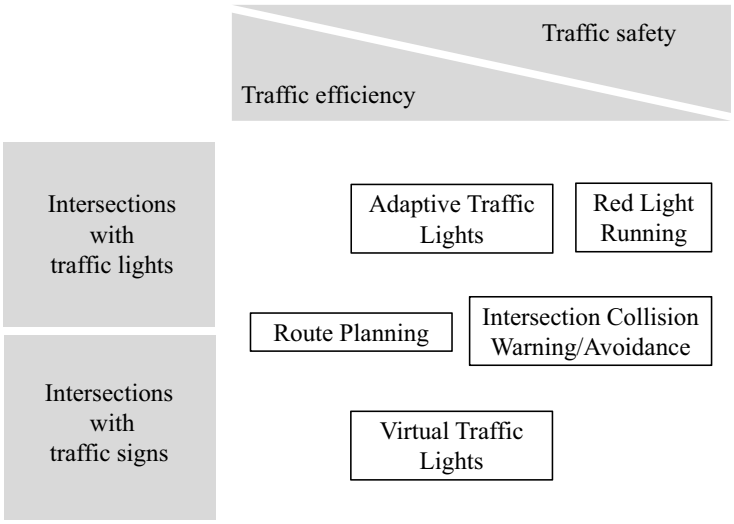


Figure 2.2: Overview of applications related to intersection management

Lights application, the Red Light Running is foreseen to adapt signal timing schemes to react to a predicted collision in real-time. Similar application which benefits traffic safety, irrespectively of whether intersection is equipped with traffic lights or traffic signs, is an *Intersection Collision Warning or Avoidance* [HH10], [HCCV13], [JBS+14]. This application focuses on avoiding or mitigating intersection collisions by warning the driver during intersection crossing maneuver in case a collision is predicted.

On the other side of the spectrum are applications that focus on optimizing traffic efficiency on a larger scale, e.g., *Route Planning*, that computes the optimum routing path over multiple intersections [WER+03]. As a general rule, Route Planning applications do not impose safety risks as only route recommendations are given and not the time when to travel certain road segments. Just as the Intersection Collision Warning or Avoidance, the Route Planning application can be beneficial on intersections that are equipped with traffic signals and traffic signs.

The last application shown in Figure 2.2 is an intersection crossing management application, called Virtual Traffic Lights (VTL), which name is adopted from [FFC+10]. Contrary to Adaptive Traffic Lights application, the VTL is foreseen for intersections that are governed by traffic signs rather than traffic lights. Since no physical traffic lights are in use, vehicles temporarily adopt the role of "virtual" traffic lights.

In this thesis the focus is put on intersection management approaches that are intended for intersections governed by traffic signs, see Chapter 1. The motivation behind researching approaches for such, also called, uncontrolled intersections, is the potential to increase traffic efficiency on such intersections to the level similar to the one achieved by (adaptive) traffic signals, *without* installing traffic light infrastructure, and thus avoiding costs associated with their maintenance. These intersection management approaches are also envisioned to make use of wireless communication. Wireless communication facilitates communication and coordination in a non-line-of-sight environment where most of the sensors would face their operational limits.

Naturally, approaches that do not rely on infrastructure traffic lights require that all vehicles were capable to communicate with each other and follow the same set of steps or the same protocol. Since current crossing order is only visible to the driver via an on-board display, inclusion of other traffic participants like bicyclists and pedestrians represent an additional challenge.

As discussed in Chapter 1, this thesis focuses on application that fall into category of Virtual Traffic Lights that manage intersection crossing on intersections governed only with traffic signs. In the following, related work is surveyed for functional and safety requirements for such applications.

Application requirements

The authors of [LW06] and [TBS14] divide intersection management protocols into *centralized* and *decentralized*. In the current thesis the usage of additional infrastructure installed on intersections (not a traffic light) is understood as a centralized management approach. Within a decentralized approach no additional infrastructure is installed and intersection management is done purely by vehicles themselves. The intersection management protocols following centralized and decentralized management approaches are analyzed below in separate.

Centralized intersection management protocols rely on infrastructure installed on intersections. This infrastructure, typically a control unit or a computational server, takes over the role of a central authority that is regulating the intersection crossing order. One advantage of using infrastructure is an improved communication with all approaching vehicles, as it is typically installed in the middle of the intersection to which most of the vehicles would have a line-of-sight conditions to. In addition, vehicles' computational load is reduced. Although a central authority represents a single point of failure, it facilitates an unambiguous intersection crossing coordination. Vehicles communicate their speed, location, and direction; and a centralized control unit allocates intersection crossing timing or order.

Dresner et al. in [DS04] and [DS05] propose a reservation-based intersection management protocol that utilizes communication between vehicles and units installed on intersections, called *intersection managers*. An intersection is divided into a grid of n x n tiles, where each tile can be reserved by only one car at a time. Approaching vehicles send the intersection crossing requests to their intersection managers, and include information on their expected time of arrival at the intersection, together with the speed, direction, and vehicle's dimensions. An intersection manager responds with a confirmation if the simulated path of the vehicle does not interfere with other reservations, or with a rejection in the opposite case. Requests are handled in a First Come, First Served (FCFS) manner. In [DS04], a reservation-based approach is presented and in [DS05] a more detailed communication protocol is described, improved by considering turns, acceleration, and canceling of the reservations. Minimal communication between vehicles and intersection is considered desirable to prevent potential delays and packet losses and to keep computational load for intersection managers within limits. Dresner's approach is meant for autonomous vehicles where human behavior is excluded. The later work of [DS07] adds functionality to

account for human drivers with the help of existing traffic light infrastructure and a *light model* that communicates with the intersection manager, although human behavior is not explicitly modeled.

An alternative approach to the reservation-based scheme is to calculate the sequence of vehicles that are allowed to cross an intersection. The approach of [WATM09] relies on the center controller installed on an intersection that computes and controls the sequence of vehicles that cross an intersection. The protocol is based on the arrival time of each vehicle at the intersection and the vehicles positions, which are communicated by vehicles to the center controller. As the primary goal is to vacate intersection as soon as possible, the center controller computes the fastest sequence to vacate the intersection. As such computation is typically expensive, especially in a real-time, dynamic programming is suggested for optimal sequence computation. In [WATM09] the authors model an intersection with two incoming lanes and in [YDM09] the same model is extended to a 4-way intersection. In the work of [AATP+13] the vehicles negotiate the "right of way" by agreeing on the sequence of crossing vehicles in real-time using Timed Petri Nets with Multipliers (TPNM). The utilization of position markers are also foreseen on all intersecting roads to detect vehicles.

The intersection crossing of a vehicle sequence is similar to the intersection CACC approach of [ZR14]. In [ZR14] an intersection controller advises the CACC-equipped vehicles on optimal course of actions.

The authors of [LP11a] describe a very simplistic intersection management approach based on communication between vehicles and a traffic signal. The traffic light communicates its signal phase to vehicles and vehicles communicate with the traffic light. The goal of the work is a formal verification that an intersection management protocol results in a collision-free operation, rather than a complex protocol development.

In the work of [LP12] Intersection Control Agents manipulate acceleration and deceleration of vehicles to coordinate intersection crossing. Several methods for solving nonlinear optimization problems are used to determine whether the trajectories of vehicles would overlap. Between a pair of vehicles with overlapping trajectories, the one with the lower priority is stopped. Communication is assumed to be perfect.

Another centralized intersection management approach is proposed by [TBS14]. Although the title of the work refers to "decentralized traffic management", according to our classification, this approach falls into the category of centralized approaches as it relies on a special infrastructure called *autonomous control agents*, which are placed at each intersection. Decentralization therein refers to individual management of each intersection separately. The control units space the vehicles in such way, that they can cross intersections without a stop. Such a synchronization of the vehicles' arrival can slow down individual vehicles but facilitates continuous traffic flow.

One of the first **decentralized intersection management protocols** is introduced by [NRTT97]. In [NRTT97] an intersection is divided into critical regions, that represent regions where trajectories of different vehicles might intersect. For each region exactly one access permission exists that a vehicle must obtain before entering the region. The obtained permission must be released when leaving the region. When conflicting vehicles, whose trajectories overlap, want to access the same region at the

same time, a priority is given based on a weighting function of vehicle's velocity and idle time, thus aiming to achieve fairness. This approach assumes that the first vehicle to approach an intersection will generate a set of permissions and that there is always a vehicle to whom a released permission to access a region can be handed over.

In [LW06] approaching vehicles self-organize into groups of three and exchange their driving plans. A solution tree is generated to search for intersection crossing sequences, in which each node represents a particular driving plan. All driving plans that are not safe, i.e., may result in a collision, are pruned from the solution tree. Only vehicles whose trajectories do not overlap are allowed to pass the junction simultaneously. Consequently, the schedule that leads to the least travel time is chosen as the best one. Here too, communication is assumed to be perfect within a certain range.

Another decentralized approach presented in [VDS08] modifies the time reservation approach of [DS04] to operate without an infrastructure. In this approach vehicles broadcast and receive information to and from each other using two types of messages – Claim and Cancel. A Claim message reserves a time slot to cross an intersection, and a Cancel message releases any held reservations. The Claim message contains information on the vehicle ID, the arrival at the intersection, and the time when intersection would be left. The reservation conflicts, i.e., Claim messages with overlapping times requested for intersection crossing, are settled based on certain priority rules (e.g., fastest vehicle has a priority, or vehicle with lowest ID). Repeated broadcast of all messages is considered to be sufficient for an up-to-date information on each vehicle. All vehicles are assumed to be able to communicate and the communication within some range is assumed to be perfect. A vehicle cannot enter an intersection if its Claim message does not have a priority and intersection must be vacated at latest at the time indicated in the Claim.

The authors of [WATM09] propose a sequence-based decentralized approach where vehicles close to each other form a team and cross the intersection together. Vehicles, that are the closest to the intersection from each team, negotiate with each other about the priority to cross the intersection, which is given to the team closest to the intersection. The last vehicle in the prioritized team passes the authority to cross the intersection to the stopped vehicles.

A similar approach has been proposed by Ferreira et al. in 2010, namely, a decentralized protocol, called Virtual Traffic Lights that coordinates intersection crossing between vehicles [FFC+10]. Vehicles that are approaching an intersection from various roads are forming clusters. The closest vehicle to the intersection is called *the Cluster Leader* vehicle. The intersection *Leader* is then elected among those cluster leaders and is taking over a temporary role of a traffic light – *virtual traffic light*, while stopped at the intersection. The intersection leader is broadcasting the current traffic light phase and is regulating the intersection crossing order. After certain amount of time, the intersection leader hands over its role to another vehicle. Thus, approaching vehicles first need to listen whether the intersection leader exists or they need to elect a new one. Communication aspects are not the focus of the work presented in [FFC+10], and thus a perfect unit-disk communication is assumed.

The decentralized approach of [SBC11] is similar to the one of [VDS08]. The vehicles

send out requests to access the shared resources, i.e., the intersection sections, but only entering the intersection sections once acknowledgments from all other vehicles in the area are received. Vehicles are first required to listen to other requests, to deliver own requests, and to receive a feedback before arriving to the decision point, which is the last point at which a vehicle must start deceleration in order to avoid entering the intersection. The approach of [SBC11] settles conflicting requests in a FCFS manner and assumes that communication provides an ordered delivery and a bounded latency.

In [LLW+14] authors propose an intersection crossing scheme based on predefined traffic rules that prioritize intersection crossing maneuvers. If a collision is predicted, the vehicle with the lowest priority is required to decelerate, and let the vehicle with the higher priority cross. Although communication failures are not explicitly considered, no collisions were reported during the FOT.

Application requirements on receiving information

Intersection management without a conventional traffic light requires all road participants being able to communicate with each other. Moreover, a common view on intersection crossing order has to be established among multiple vehicles. Typically, in the related work, specific requirements on receiving information are not yet formalized.

Considered driver reaction behavior

The majority of studied works devised their approaches for autonomous vehicles [NRTT97], [DS04], [SBC11], [TBS14], thus, leaving the need to consider drivers' reaction by the application obsolete. The work of [FFC+10] assumes that intersection crossing instructions are shown to the human drivers over an on-board display, but the focus is not to ensure the application's suitability for all types of drivers.

Several works extend existing approaches for autonomous vehicles to co-exists with manually driven vehicles, e.g., [DS07], [dLF10], [OMV+12], [QGMF14], by utilizing existing traffic lights [DS07] or smartphones [NVT13]. By utilizing conventional traffic lights, authors of [DS07] make use of the traffic light property to be safe for most of the humans by integrating the yellow phase. Here too, the focus is to make intersection crossing schedule visible to human drivers but not on accommodating driver characteristics like different reaction times and braking intensities.

Integrated fail-safety features

Since most applications are foreseen for automated vehicles, the challenge of predicting human driver reaction is less prominent. But, the dependability on wireless communication requires appropriate fail-safety features to be integrated in the design of applications. E.g., the rebroadcast of messages is foreseen by [DS05] in order to compensate for the lost packets. And the approach proposed in [TBS14] requires larger inter-vehicle distances between vehicles to exclude collisions, which simultaneously degrades the traffic efficiency.

The main fail-safety feature of [DS04] and [DS05] is that no vehicle is allowed to enter an intersection without a confirmation from a central infrastructure that keeps track of all reservations. Thus, the lost packets result in an additional delay but not in a collision. The failures that can be introduced due to sensor errors were treated

with an inclusion of an additional space buffer around vehicles. The work of [DS08] extends the original centralized reservation protocol of [DS04] and [DS05] to safely deal with mechanical failures, e.g., a collision or a vehicle breakdown. The major assumption is that a vehicle can notify an intersection manager about the failure, so that the intersection manager would stop granting further reservations and can notify others. Simulated evaluation with one intentionally crashed vehicle inserted into an intersection showed that such safety measures result in a car crash reduction from approx. 90 to 1.5 cars crashed if no packets were lost. If the packets notifying of this collision were all lost, then the safety measures result in approx. 3 crashed vehicles. Additionally, the safety measures result in a collision velocity reduction. In [DS08] the authors describe a protocol to prevent vehicles from entering an intersection after a mechanical failure. And in [AFV+12] the authors describe a protocol to prevent a collision among vehicles that are already in the intersection where mechanical failure happens. Similarly in [SBC11], if a vehicle fails to coordinate its crossing of an intersection, it must come to a complete stop before the junction.

Verification methods

Managing intersection crossing according to some protocol or a set of rules requires *verification* of the design in order to eliminate possible design deadlocks or errors. Even a seldom and a little error in the logic of the protocol might eventually lead to a collision. As a consequence, most of the related work verifies their designs to some extend or the other. Most commonly, a simulation is performed, in which vehicles cross intersections running a specific coordination protocol and it is observed whether vehicle collisions take place or not. The drawback of such approach is that the protocol can be considered collision-free only for the simulated scenarios, which is only a small fraction of a possible set of circumstances. The verification using formal methods discussed in Section 2.1.3 provides the reliable guarantee that the verified design does not cause vehicle collisions in general. The challenge behind verifying intersection control protocols lies in the different nature of subsystems that need to be verified [LP11a], [NRTT97]. In particular, the movement of the vehicles is a continuous system which is interacting with a discrete system of the application's protocol, and with the discrete communication events.

The authors of [NRTT97] make use of a high level Petri Net analysis, namely a predicate/transition-net, to verify that their intersection crossing algorithm does not cause any collisions. By formally proving that there is never more than one car in every critical region, the algorithm safety is proven. This is modeled with the token-based principle known from computer networks. A vehicle requires to possess a token to access one critical part of an intersection, while there is exactly one token for each critical part existing. Once not needed anymore the token is released. In order to prove the safety of this approach, an occurrence graph or a state space of all possible states is build. No deadlocks were found in [NRTT97], i.e., there are no states with only incoming arcs and every state is found reachable.

Loos et al. in [LP11a] propose verification of a simple intersection crossing model with two crossing lanes and one traffic light. The focus of the paper is the verification

of a hybrid system with continuous vehicle movements and discrete protocol decisions but only an implicit model of communication. The verification proves that under no circumstances a car will occupy an intersection if the traffic light is red. This condition is expressed in quantified differential dynamic logic, which is a type of theorem proving, and is verified with the help of a tool called KeYmaera [PQ08].

In [AMB+12] Asplund et al. describe an automatic verification of a simplified protocol of [SBC11] using an SMT-lib language and a *Z*3 theorem prover [dMB08]. The safety invariant requires, among others, that vehicles crossing an intersection are allowed to do so, i.e., have acquired a resource and have enough time for crossing. The freedom from deadlocks has also been ensured. Nevertheless, strong assumptions on communication have not been eliminated for the current verification approach.

The complexity of the whole system that is to be verified, makes it very challenging to include all three aspects – discrete protocol, communication, and continuous movement of the vehicles. The discussed verification approaches make simplifying assumptions on communication and do not account for its unreliable nature.

In [VDS08] authors simulated the impact of imperfect communication on their decentralized protocol. When packet loss was kept at 40 % no vehicle collisions happened. Once packet loss rose between 40 % and 60 % the system begin experiencing safety failures, i.e., 5 out of 1200 simulated scenarios resulted in vehicle collisions.

Applications classification

In Table 2.2.2 we summarize intersection management approaches found in the literature as well as their assumed communication model and how they accommodate functional and safety requirements. In particular, the first three columns indicate a reference to each work, type of intersection crossing approach, and the supporting technology. The *Driver characteristics* column summarizes if and how driver reaction behavior is accounted in the design. The *Fail-safety* column indicates design features that counteract possible failures, e.g., communication packet drop or inadequate response of the drivers. The *Verification* column outlines how each approach is verified, e.g., with formal methods or with simulations.

Reference	Approach	Supporting technology	Driver characteristics	Fail-safety	Verification
[NRTT97]	Decentralized	Idealistic communication	AV assumed	N/A	Formal verification, using predicate/transition-nets
[DS04]	Centralized	Idealistic communication	AV assumed	Vehicles may not cross unless reservation is acknowledged by intersection manager	N/A
[DS05]	Centralized	Idealistic communication	AV assumed	Vehicles may not cross unless reservation is acknowledged by intersection manager	Checked for deadlocks
[DS08]	Centralized	Idealistic communication, packet losses are modeled	AV assumed	Vehicles may not cross unless reservation is acknowledged by intersection manager, no more reservations once a vehicle failure is detected, intersection manager informs other vehicles of the accident by repeated broadcast	Small amount of collisions even if packet loss is zero
[VDS08]	Decentralized	Idealistic communication, packet losses are modeled	AV assumed	N/A	Simulation, the higher the packet loss rate, the more collisions
[LW06]	Decentralized	Idealistic communication	Human behavior is not modeled	N/A	N/A
[WATM09]	Centralized and decentralized	N/A	N/A	N/A	Simulation
[YDM09]	Centralized	N/A	N/A	N/A	N/A
[FFC+10]	Decentralized	Idealistic communication	Human behavior not modeled	N/A	N/A
[LP11a]	Centralized	Idealistic communication	N/A	N/A	Formal verification

Reference	Approach	Supporting technology	Driver characteristics	Fail-safety	Verification
[SBC11]	Decentralized	Idealistic communication, ordered delivery and bounded latency guarantees	AV assumed	Vehicles are not allowed to enter intersection unless everyone acknowledge the crossing	Checked for deadlocks
[LP12]	Centralized	Idealistic communication	AV assumed	N/A	N/A
[TBS14]	Centralized	Idealistic communication	AV assumed	N/A	Simulation, no collision is detected

Table 2.2: Classification of intersection management applications.
N/A – information is not available. AV – Autonomous vehicle

2.3 Design decisions made in this thesis

In previous sections we outlined aspects that are important for the design of vehicular applications based on communication. In Section 2.2 we analyze the related work on two application use cases – the avoidance or mitigation of rear-end collisions and the increasing of traffic efficiency on intersections that are not equipped with conventional traffic lights, see Section 2.2.1 and Section 2.2.2 respectively. In particular, functional requirements of various applications are analyzed as well as how applications support functional safety and accommodate wide distribution of driver population. In the following we summarize major design aspects from related work and outline our decisions for the design of two safety-critical applications. Outlined design decisions allow us to answer the question stated in Chapter 1, namely, how to design a safety-critical application that is fail-safe against unreliability of vehicular communication and unpredictable driver behavior.

2.3.1 Rear-End Collision Avoidance Application

Design approaches of two applications, that actively address rear-end collisions, have been reviewed in Section 2.2.1. An FCW and an RECA assist a driver with a warning or an automatic braking features in case of a potential rear-end collision. Irrespective of the technology used to support these applications, all applications have a decision mechanism or an algorithm that evaluates the current traffic situation as safe or dangerous, and thus requiring a warning or an automatic braking. Related work distinguishes conditions that trigger the application's warning or automatic braking events as those that are distance-based or time-based. The benefits of a distance-based approach include a direct relation to the kinematic motion of the vehicles and a straightforward incorporation of the driver's reaction behavior. As it is challenging to correctly estimate the driver's reaction behavior, see Section 2.1.2, fixed mean or average values are often used. However, not all drivers are able to react to a warning that considers mean reaction time and mean braking intensity. This results in a warning that is considered a false positive warning for some drivers or a false negative warning for the others. There are various modifications and extensions to a simple distance-based approach, e.g., considering age and gender of the driver [Ati10]. While this addresses the problem of estimating driver reaction, it does not solve it completely. Other modifications include road friction and weather conditions in the calculation which simply increase the distance where a warning or automatic braking should be initiated.

A time-based approach is considered to be a natural indicator of how urgent dangerous situations are. The driver's reaction to a warning issued in accordance with a time-based approach is considered, only indirectly through the estimation of a time threshold triggering the warning. Multiple studies have been performed to find an appropriate threshold value, but the lack of knowledge on lead vehicle kinematic parameters, e.g., deceleration, degrades the benefit of utilizing the time-based approach [SM03].

The goal of the warning is also an important factor for the application's design,

whether it is to redirect driver attention or to warn of imminent collision that requires urgent reaction. Depending on the goal of the warning, the warning can be given rather early or late. Authors of [LMBR02b] provide design recommendations: they suggest estimating the driver's distraction (e.g., detecting the cellphone usage or analyzing driver eye gaze via a camera) that might be relevant for adjusting the warning thresholds. The graded warnings are also considered to provide greater safety and not to habituate drivers to collision warnings [WBMD97], [SSH98], [BKPP02], [LHH04].

Historically, rear-end collision avoidance applications have been using RADARs to sense the space in front of their host vehicle and to determine the lead vehicle's velocity and position. Few recent approaches utilize wireless communication and a fusion of several technologies. Due to the safety-critical nature of rear-end collision avoidance applications, certain safety requirements have to be met, see Section 2.1.3. Automatic braking can be considered as a fail-safe feature to prevent incorrect estimation of driver response to a warning. The immediate notification to the driver is considered to be a widely used practice against failures that are caused due to unreliability of information on the leading vehicle or other system failures. Although modern sensors are considered to be highly reliable in detecting obstacles, their limitations might be critical, especially for automatic braking decision, since false alarms for automatic braking is much more invasive as false alarms for warnings. Wireless communication is generally less reliable; transmitted packets can get lost, due to interference, fading, or packet collisions, and thus never be received. Not possessing the updated information on the leading vehicle might lead to false decisions for warnings and automatic braking.

The "correct balance" between false positives and false negatives is often mentioned as a requirement in the related work. Yet, this is rarely addressed in the design of the application. Typically, the automotive manufactures indicate 20 % as an allowed threshold for false positive rates without mentioning the tolerated values for false negative rates [F. 11b]. The strict safety requirements of such safety-critical applications as RECA would require the highest ASIL, cf. Section 2.1.3. Such considerations are typically not a part of the design but rather evaluation of the applications. Meanwhile, the non-zero false negative warning does not necessarily mean a vehicle collision, especially if the application's design integrates appropriate fail-safety features, see dependability concept in Section 2.1.3.

Most automotive products, integrate a fail-safety feature which consists of informing a driver when a system is experiencing a failure. Such feature is also described in the ISO 15623 [ISO13a], where a driver always remains responsible for a safe operation of the vehicle. The correct notification to the driver is very important when a system experiences a failure. This includes a "right" design of the human-machine interface and informing about system limitations in the manual or with caution stickers.

As the rear-end collision algorithms typically do not comprise of complex distributed coordination, the formal verification of the algorithms' design is not performed. Verification of the correct operation on the field test tracks or via simulations under the most relevant collision scenarios is performed instead.

Design decisions adopted in the thesis

As none of the reviewed applications address rear-end collisions purely with the help of wireless communication and in a fail-safe and a verified manner, we devise an application design based on findings acquired through the literature review summarized previously. Our goal is not to eliminate the use of sensor technologies but to investigate limits of communication in supporting of rear-end collision avoidance applications. An RECA application is designed with automatic braking that acts as a fail-safety measure preventing inadequate driver reaction to a warning. A distance-based approach is chosen to calculate triggering conditions for warnings and automatic braking events. The exact knowledge of the driver's reaction to a warning is not safety critical because automatic braking is integrated. However, since wireless communication is less reliable than sensor technologies and notifying the driver of failures may not be sufficient, additional fail-safe features are required. Thus, the design of RECA application that is based on wireless communication should include an additional fail-safe feature to counteract unreliability of communication without relying on the driver. More specifically, during the non-reception of an expected communication packet, an uncertainty about the vehicle environment arise. Thus, a fail-safe application is to assume the worst case situation change during the uncertainty period.

The default application's communication requirements, e.g., update frequency of 10 Hz, can be either insufficient for an application or can congest the radio channel unnecessarily, depending on the traffic situation. We decided to perform a detailed application requirement analysis, including analysis of the requirements on receiving the information, rather than relying on the default requirements on communication.

The road friction and weather conditions are ignored for the application design for two reasons: first, according to rear-end collision statistics, most of the collisions happen during the day-time under normal weather conditions [tra12], and second, the inclusion of road friction and weather conditions simply increases the calculated kinematic distance. Future work can focus on including all aspects that have even a minor impact on the calculation of the kinematic warning and automatic braking distances.

For the design of application no differentiation between various levels of warning is done as this requires extensive driver psychology studies and contributes to driver's comfort, rather than to traffic safety. Formal verification, test track tests, and simulations, are not performed, since the goal is to design a fail-safe application that is irrespective of probable errors caused by human drivers and communication. Hence, an analytical study under the most typical collision scenarios is considered sufficient.

From now on, the application designed in this thesis is referred to as a Rear-End Collision Avoidance (RECA) unless specifically stated otherwise. Chapter 4 describes the rear-end collision avoidance application design in more details.

2.3.2 Virtual Traffic Lights Application

Section 2.2.2 examines proposed applications that are intended to manage intersections governed with only traffic signs, i.e., no conventional traffic light infrastructure is present. Summarizing related work, the following statements can be outlined: inter-

section management applications can be divided as those that follow a centralized approach and those that follow a decentralized approach. Applications that follow the centralized approach imply the usage of some infrastructure installed on intersections, which is not necessarily a traffic light but a control unit or a computing server that communicates with all approaching vehicles and perhaps neighboring intersections. Decentralized approach implies that all vehicles follow the same protocol to self-organize each other for intersection crossing without the help of additional infrastructure. Dedicated infrastructure installed on intersections could act beneficially by facilitating better connectivity with approaching vehicles as well as taking off computational load from vehicles. However, the investment for installation and maintenance of such infrastructure unit can be comparable with the one arising from conventional traffic lights. Additionally, although one central authority regulating intersection crossing eliminates the ambiguity on the crossing order, similarly as conventional traffic lights do, one central authority also represents a single point of failure. As outlined in Chapter 1 the goal of this thesis is to design an intersection management application without the use of additional infrastructure.

Decentralized approaches suggested in the literature follow two main patterns: either each vehicle reserves a crossing time and space on its own, or vehicles form groups and group leaders arrange a crossing sequence for each group. Individual time reservations are challenging, as vehicles must strictly respect their reservations which can lead to collisions if vehicles fail to do so. Realizing ordered intersection crossing by utilizing vehicle sequences relaxes the strict time and space constraints of the reservation-based approach. However, computing the optimum sequence might result in a high computational load and potential delays.

In general, design approaches in the literature make strong assumptions on communication. They either assume idealistic communication within a certain communication radius, e.g., [FFC+10], [TBS14], or they introduce packet losses and evaluate resulting system behavior [DS08]. Repeated broadcasting of the messages considered to be sufficient for eventual message delivery and the resulting delay considered non-detrimental for the protocol's safety [VDS08].

Examined intersection management protocols are mostly envisioned for autonomous vehicles, so it is not necessary for the protocol design to accommodate various human driver characteristics mentioned in Section 2.1.2. The protocols that are intended for human drivers or aim to accommodate both autonomous vehicles and human drivers, typically focus on making intersection crossing policy "visible" to the human driver without modeling realistic driver behavior [DS07], [NVT13].

The centralized protocols of [DS04], [DS05], [DS08], and [SBC11] integrate a fail-safety feature that requires vehicles to receive an acknowledgment from the central authority when they are allowed to enter the intersection. The driver acceptance is then implicit, which in the case of packet loss results in additional travel time delay but not in unsafe intersection entrance. As idealistic communication and autonomous vehicles are often assumed, fail-safe features against unreliability of wireless communication and unpredictability of a human driver are not part of the protocol design.

Due to the safety critical nature of intersection management, the protocols that man-

age intersection crossing need to be verified not to cause any collisions. Verification methods used in the related work range from simulations to verification using formal methods. The challenge of formal verification consists of the need to verify different subsystems, i.e., discrete protocol decisions, discrete communication events, and continuous driving dynamics. Verification through simulation, although allows to integrate complex models, e.g., fuel consumption estimation model [NRTT97], only checks that no collision happens in a particular scenario setup that was simulated, but the actual set of scenario possibilities can be much larger. Thus, formal verification is often considered to be exhaustive proof for the collision-free nature of an individual protocol. Even the complex formal verification methods considered in the literature, make strong assumptions on reliability of communication [NRTT97], [LP11a].

Design decisions adopted in the thesis

Although plenty of different approaches that coordinate intersection crossing are discussed in the literature, they are not suitable to fully answer the research questions stated in this thesis, cf. Chapter 1. In particular, a decentralized protocol is required to avoid the use of additional infrastructure. In addition, the protocol should be safe irrespective of adversities caused by realistic communication environment and inadequate driver response. Appropriate fail-safe features should be integrated, as idealistic communication and autonomous vehicles are not assumed in this thesis. The approach suggested by Ferreira et al. [FFC+10], called Virtual Traffic Lights, provides needed foundation for this thesis. The VTL approach does not rely on additional infrastructure; the role of the traffic light is temporarily assigned to one of the vehicles that is stopped at the intersection resulting in a quasi-centralized intersection management. The published work of Ferreira et al. [FFC+10] does not provide much design details, rather that it is based on the requirement of conventional traffic lights. To a certain extent, the presented protocol is a "proof of concept" as idealistic unit disc communication as well as an obedient and predictable driver are being assumed.

The protocol designed for this thesis, although is based on the idea of Ferreira et al., needs to be designed anew in order to integrate fail-safe features against unreliability of communication and unpredictability of human drivers. In addition, application design needs analysis of requirements, not only to determine what application should do but also to estimate necessary communication parameters to support the designed application. Application requirements analysis is based on requirements of conventional traffic lights, as outlined by Ferreira et al. in [FFC+10].

Due to anticipated complexity of the resulting protocol, a formal verification is required to ensure that the designed protocol is safe. The chosen verification method is called model checking. Model checking can facilitate formal verification in all three essential system components: wireless communication, vehicle movement and the protocol itself, cf. Section 2.1.3.

Several approaches in the related work aim for local or global efficiency and for local or global fairness. We do not address global efficiency or any type of fairness in the designed protocol. For each intersection crossing the protocol accounts for local efficiency, i.e., the roads with larger number of vehicles would have a crossing

priority. Global efficiency requires a specific communication protocol between multiple intersections and is left out of scope. Local or global fairness can be integrated in the designed approach by adding weighting factors when assigning priorities for the crossing roads, and is left for the future work.

From now on, the approach designed in this thesis is referred to as Virtual Traffic Lights (VTL) unless specifically stated otherwise. Chapter 5 describes the design of the VTL application in more detail.

2.3.3 Assumptions and out of scope

Within the scope of this thesis several assumptions have been made. First of all, all vehicles are assumed to possess an inter-vehicle communication capability and are able to send and receive information. In addition, all vehicles are assumed to possess a Global Navigation Satellite System receiver for obtaining its own positioning information with at least lane-level precision[3]. All vehicles possess digital maps. Driver assistance applications, developed in this thesis, are assumed to have access to the host's in-vehicle sensors that deliver information on the ID, position, acceleration, and speed of the host vehicle. Information on the host vehicle is assumed to be accurate and error-free. The applications are assumed to make use of Cooperative Awareness Messages that are periodically broadcast by all vehicles. All vehicles have the same physical dimensions and represent a typical passenger car. No other traffic participants, like bicyclists or pedestrians are considered.

The following aspects are left out of scope: the human-machine interface, in particular how the driver assistance information is presented to the driver which is partially provided in [J.L07] and in the work-in-progress SAE J2400. This thesis deals only with the development of two applications on the design level. The implementation, production, and maintenance are left out of scope.

[3]Considering proliferation of various satellite navigation systems and relative error in localization (vehicles in the vicinity tend to experience similar localization errors), assumption for lane-level localization precision is not unrealistic.

3
Performance evaluation methodology

Performance of a communication-based system depends on multiple influencing factors, e.g., radio channel conditions, transmission parameters of participating vehicles, as well as on the application itself. The impact of these factors on the network performance has been extensively studied in the related work. Multiple analytical and simulation models can be used to estimate which network performance can be expected when certain influencing factors contribute. However, a detailed sensitivity analysis that evaluates and quantifies the impact of each individual influencing factor is missing. It is important to know the impact of individual influencing factors because information about these factors is often inaccurate. The system designers need to know which errors in the network performance estimation should be expected. If information on the influencing factors is erroneous either extra efforts should be invested to obtain accurate information or appropriate countermeasures integrated, especially when errors in the network performance estimation cannot be tolerated by safety-critical applications. In addition, it is not only desirable to determine how certain communication parameters lead to certain application performance and as a consequence to certain traffic performance, but also to translate application and traffic requirements into optimal communication parameters. In such a way, adjustment of communication parameters can be done taking application requirements into the account. These two open issues – the lack of a detailed sensitivity analysis that evaluates and quantifies impact of influencing factors on the network performance and the lack of appropriate bidirectional connection between communication parameters and application or traffic performance are formulated and elaborated in Section 3.1.

Section 3.2 provides an overview of existing performance evaluation methods for sensor- and communication-based systems. It is essential to first study how performance is evaluated for sensor-based systems in order to be able to build on top of this knowledge and adopt the most relevant methods for evaluation of communication-

based systems. The main difference in performance evaluation of these two systems is that system designers of a sensor-based system have little real-time control on sensor performance, whereas communication performance can be adjusted by, among other factors, communication parameters. Section 3.2 is intended to provide an overview of existing methods, with no claim on completeness.

The rest of the chapter presents contributions of this thesis to evaluate performance of communication-based driver assistance systems. Open issues outlined in Section 3.1 are addressed in Section 3.3, which describes *a sensitivity analysis* of the impact of radio channel and network parameters on network performance, and in Section 3.4, where a bidirectional connection between network and application layers with the help of *an awareness principle* is presented. Moreover, the ultimate goal of performance evaluation of communication-based systems is not to evaluate the network performance nor application performance but to determine the impact a communication-based system has on road traffic. Section 3.5 combines the knowledge of the previous sections and presents a general methodology framework. This methodology framework can be followed to answer the research question stated in Chapter 1 on determining the scalability of IEEE 802.11p communication to reliably support applications and to evaluate the impact of fail-safe applications on road traffic.

Parts of the work presented in this chapter have been previously published in [AGH11] and [AMSETM11].

3.1 Overview and open issues

Performance evaluation in the area of vehicular networks can be performed by means of evaluation via simulation [MHDK09], evaluation in a driving simulator [JLC08], [LMBR02a] or simulation with hardware-in-the-loop [GPSV06], [OHS00], small-scale field operational tests, typically with a mock-up vehicle [SM03], and large-scale, long-term field operational tests that spread over several months or even years [NSH+06], [NLS+11], and [ADM+12]. In addition, performance evaluation requires identification of relevant scenarios where a specific application should provide assistance to the driver. Crash statistics are often used to identify the most relevant scenarios [HLA+12]. The ISO standards that define functional requirements for certain driver assistance systems also include *objective tests* that describe relevant scenarios which should be used for performance evaluation of a system, see [ISO13a], [ISO13b].

Performance of a driver assistance system can be evaluated on, at least, three different levels: *technology*, *application*, and *traffic*[1]. These three levels are graphically depicted in Figure 3.1. The technology level addresses the performance of a specific technology used to support driver assistance applications. Technological performance evaluation of vehicular communication networks, typically, only covers the lower OSI layers. In particular, the network layer performance influenced by radio channel conditions and transmission parameters, such as transmission rate (Tx. rate) and transmission

[1]Another level is *a driver level* at which driver acceptance, typically in the form of a follow-up survey, is evaluated. Common metrics are ease of use, ease of learning, perceived value, and driving performance. The explicit driver acceptance evaluation is left out of scope of this thesis.

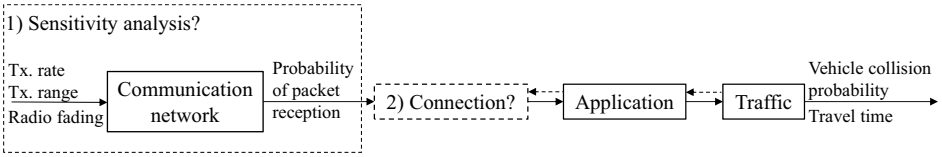


Figure 3.1: Open issues in performance evaluation of communication-based driver assistance applications: 1) The lack of sensitivity analysis that determines how factors such as radio fading, transmission range (Tx. range), and transmission rate (Tx. rate) impact the network performance, e.g., probability of packet reception. 2) The lack of bidirectional connection between network layer and application, or traffic layers

range[2] (Tx. range), is evaluated. Metrics that are used to evaluate performance on the technology level, e.g., *packet delay* or *probability of packet reception*, are typical network layer performance metrics.

Application level evaluation of a driver assistance system allows identification of the limits and flaws of the application's logic in a realistic environment and optimization of the application's parameters, e.g., warning time optimization based on the feedback from drivers. Some verification methods described in Section 2.1.3 are similar to the application level evaluation methods.

Traffic level evaluation is performed to predict potential benefits a system might have, i.e., the impact a driver assistance system might have on road traffic. Typical measures of benefit can be related to traffic safety and efficiency, e.g., the overall number of vehicle collisions, as well as injury and fatality rates, vehicle speeds and speed variability, as well as the amount of travel time delay and the maximum number of vehicles per hour passing a certain point [BMY05].

Most of the related work on performance evaluation of communication-based driver assistance systems is focused on *the technology level*. The relationship between the influencing factors, e.g., transmission parameters and the network layer metrics, e.g., probability of packet reception, has been extensively studied. There are exist multiple analytical and simulation models that can be used to estimate what network performance can be expected if certain combination of influencing factors is chosen. However, up to now, no study exists that performs sensitivity analysis on how each of the influencing factors contribute individually. In addition, the information on the influencing factors is often inaccurate. Information detection and estimation might cause extra efforts, e.g., additional data broadcast. However, the result of information detection can be only partially beneficial, e.g., if the network is congested, additional packet traffic will worsen the situation even more. *A sensitivity analysis* can provide information on when accurate information is important as well as can quantify the expected errors in network performance estimation if information on the influencing factors is inaccurate. A detailed sensitivity analysis is essential for designing a system that is functional under realistic and inaccurate information. This is depicted in Figure 3.1 and marked as the missing **1) Sensitivity analysis**.

[2]Transmission range represents a distance that can be successfully reached with a certain transmission power under the assumption of deterministic Two-Ray Ground propagation model.

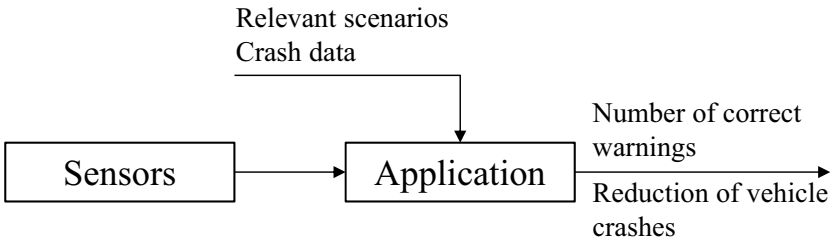


Figure 3.2: Typical process of performance evaluation for a sensor-based system

Furthermore, one cannot judge the application's performance, let alone the driver assistance system's impact on traffic, based solely on network layer metrics. It is possible to evaluate how certain communication parameters impact the application performance and as a consequence traffic, but an appropriate bidirectional connection can allow to propagate application and traffic level requirements to optimally adjust communication parameters. This is illustrated in Figure 3.1 and marked as the missing **2) bidirectional connection**. In the related work multiple approaches exist that analyze performance evaluation on the "pre-application" level, which is moving one step "up" from technology level but still giving only limited insight on application itself and on road traffic. In order to utilize these "pre-application" level metrics for adjusting of the communication parameters, it should be possible to express application requirements in the form of these metrics.

The two open issues that we address within the scope of performance evaluation of communication-based driver assistance systems are the lack of detailed sensitivity analysis and the missing bidirectional connection between network and application layers. This is graphically noted with the doted lines in Figure 3.1.

3.2 Related work

3.2.1 Evaluation methods for sensor-based systems

Performance evaluation of sensor-based systems, based on the related work found in the literature, mostly address evaluation of application logic and impact on road traffic. Quite often sensors are used as off-the-shelf products and driver assistance system designers have little to no real-time influence on the sensors' parameters.

Although sensor-based applications are sometimes evaluated on the application level, with such metrics as number of correctly given warnings, the prime interest lies in the evaluation on traffic level, i.e. evaluation of impact that driver assistance systems have on traffic safety and efficiency. A safety benefit evaluation is typically performed with the help of crash statistic data which is fed into a simulator with an implemented safety system, as presented in [LT06], [MHD08], [MHDK09], [ADM+12], [KG12].

The typical performance evaluation process for sensor-based systems is displayed in Figure 3.2. Sensors gather information about the neighborhood, either actively or passively, and feed this information to the application. In addition, relevant test

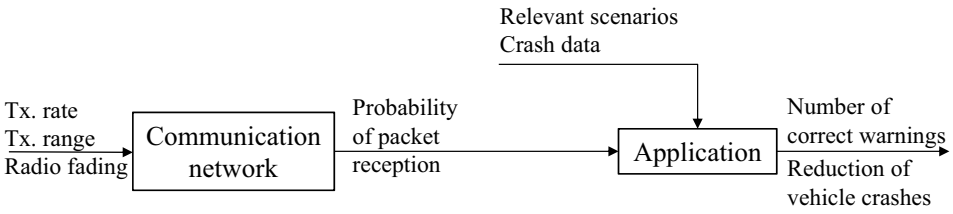


Figure 3.3: Typical process of performance evaluation for a communication-based system. Major focus is put on understanding communication network performance. The lack of bidirectional connection between network and application layers is depicted with an arrow from communication to application, but not vice versa

scenarios or crash data are used. Depending on whether relevant test scenarios or crash data are used, the evaluation can be carried out on the application level, with metrics like *number of correct warnings* or on the traffic level, with metrics like *reduction of vehicle crashes*. The safety benefit of a sensor-based rear-end collision avoidance system has been quantified as a reduction of driver fatality or injure rates by up to 50 %, depending on the type of the system and traffic scenario, e.g., [KG12], [ADM+12]. The benefit of sensor-based Adaptive Signal Control has been quantified in the form of travel time delay reduction by up to 40 %, e.g., [BMY05]. This confirms the great potential and usefulness of two selected applications, although based on sensor technology, to positively impact traffic safety and efficiency.

3.2.2 Evaluation methods for communication-based systems

In contrast to sensor-based systems, performance evaluation of communication-based systems, as still being actively researched and developed, strongly focuses on performance evaluation of supporting technology, i.e., communication. Communication-based systems require at least one more vehicle that would communicate with the host vehicle. In addition, transmission parameters as well as current radio channel conditions have a large influence on the overall performance.

In Figure 3.3 we show a typical performance evaluation process of a communication-based system. Initially, the performance evaluation methods of generic wireless communication networks have been adopted for vehicular networks. For this reason, a lot of work is focusing on evaluating vehicular communication networks on the network layer with metrics like probability of packet reception (PPR) or end-to-end packet delay. This is depicted on the left side of Figure 3.3. Studying of this relations allows to adjust transmission parameters depending on the desired outcome, e.g., if it is required to achieve a PPR of above 90 %, the transmitting parameters can be adjusted accordingly. However, the network layer metrics provide little insight on whether application warns the driver on time (application level evaluation) or whether the traffic collision could be avoided (traffic level evaluation) [EGH+06], [BK06].

Minor part of related work is focusing on evaluating on the application layer, depicted on the right side of Figure 3.3. Performance evaluation of an application can be described with such metrics as number of correctly given notifications or false

positive/false negative error rates. Although it can be determined, which transmission parameters and which probability of packet reception resulted in certain application performance, the lack of appropriate connection between network and application layers prohibits adjustment of transmission parameters to achieve desired application performance. In particular, zero false negative rates required for some application does not necessarily translate in 100 % PPR requirements. Similarly, the default update frequency of, e.g., 10 Hz [XSJC03], [The05], [Ins09], might be excessive, as unnecessarily congesting the channel, or not sufficient for reliable operation of an application.

More interestingly than application performance itself is the application's impact on the road traffic. As application can perform perfectly (its requirements fully supported by communication) but traffic can still be unsafe or inefficient (due to e.g., faulty application design). However, the evaluation from traffic perspective is complicated by several factors, e.g., collision situations are very rare and the success of an application can be dependent on the adequate reaction of the driver. Nevertheless, depending on how an application is designed, the application performance metrics can already give insights on the traffic performance. For example, if an application is designed to avoid collisions by automatic braking, satisfying the application requirement to perform correct automatic braking can describe not only application performance but also the general traffic safety. In addition, the impact on the traffic level can be determined as for the sensor-based evaluation – by utilizing crash statistic data.

In the following, the related work on relations between input parameters influencing communication network performance and their output metrics is described in detail. Next, an outline of the related work on approaches to connect communication network with application layers is presented. A short conclusion summarizes best practices and their drawbacks for performance evaluation of driver assistance applications.

Understanding communication network performance

The methods to evaluate vehicular networks evolved from evaluation of generic wireless communication networks. Initial focus was on the network perspective evaluation, in particular, it has been studied how typical network performance metrics, e.g., probability of packet reception, packet error rate, channel busy ratio, and average packet delay, are influenced by such factors as transmission parameters and radio channel conditions. High mobility of vehicles, which leads to fast-fading radio channel conditions, shadowing due to obstacles, as well as decentralized coordination of medium access, pose additional challenges specific to vehicular networks. As a result of extensive research, a lot of knowledge has been acquired in the understanding of this relationship, e.g., various congestion control mechanisms have been suggested that allow adaptation of transmission parameters to avoid congestion in the radio channel [GSKB07], [TMMSH09], [FHSK10], [FHSK11], [Ins11], [TJHD13], and [BKR13].

The network performance evaluation studies mostly comprise of either simulation campaigns [YEY+04], [ZSO+05], [BTD06] or empirical measurements [HXRK09], [BSK10]. Due to high computational load of some simulation experiments and their time-consumption as well as the expenses behind the measurement campaigns, the network performance has been often described with analytical models to aid further

evaluation. In the following, we describe some of the existing models used to describe vehicular communication network performance.

The *packet error model* presented in [ZSO+05] takes into account the radio propagation in different scenarios and effects of such parameters as modulation mode, coding rate, and packet length. The drawbacks of the model include abstraction from differentiation of line-of-sight and non-line-of-sight cases and not addressing the effects of vehicle motion.

The empirical model of Killat et al. [KH09] utilizes a large set of traces gathered during simulation of static nodes that are transmitting periodic beacons. The model outputs *probability of packet reception* based on four input parameters: the distance between sender and receiver, transmission range, transmission rate, and the number of communicating vehicles. Several other parameters, such as the size of the packets, data rate, and radio fading conditions, are also included, although not variable. The empirical model has also been validated against simulation results. If it is desired to maintain certain PPR, this model allows to determine what transmission parameters should be used, depending on the number of communicating vehicles. Although very practical, the model has some limitations, e.g., radio conditions are expressed only with a Nakagami distribution and a relaxed fast-fading parameter of $m = 3$, which is rather pessimistic at short distances, i.e., distances smaller than 100 m. The default packet size is fixed at 400 bytes, the data rate is fixed at 3 Mbps, the ranges of varying parameters are also limited, e.g., transmission rate can vary between 1–10 Hz, transmission range between 100 and 500 m, and the number of communicating vehicles should not exceed 500 veh/km (vehicles per kilometer).

The *local broadcast capacity model* of Schmidt-Eisenlohr et al. [SEH10] estimates the amount of data that each node may transmit per second such that at each receiver in a certain *awareness range p* percent of all messages is received successfully. The authors deal with the challenge of multiple senders and multiple receivers, all sharing one communication channel.

The models that describe *packet delay*, e.g., [GNK05], [ARM09], [GHMK14], can utilize analytical or simulation studies and typically include rebroadcasting of messages and different mobility models. Groenevelt et al. in [GNK05] suggested a stochastic model that predicts message delay in mobile ad hoc networks, where nodes rebroadcast messages and move according to several mobility models (random waypoint, random direction, and random walker mobility models). The work of [ARM09] simulated realistic vehicle movement together with various forwarding schemes. The resulting message delay traces have been fitted to a heavy-tailed distribution. The authors of [GHMK14] proposed an analytical model that allows to predict packet end-to-end delay based on such factors as data packet loss probabilities, retransmission times, and back-off timers.

The authors of [vERH12] presented an analytical model to estimate the 802.11 Distributed Coordination Function (DCF) performance. The analytical results are validated against simulations and report a good match for modeling of such parameters as *probability of successful reception*, *service time*, and *saturation of the channel*.

Connecting communication network and application performances

Connecting the performance of communication network to application performance is essential not only to evaluate whether communication is adequate to support applications successfully and subsequently positively impact on the traffic, but also to provide insights on how to adjust transmission parameters in order to fulfill applications requirements.

The periodic status messages broadcast by all vehicles are also called awareness messages; upon reception of these messages vehicles acquire *mutual awareness*. The protocols controlling transmission frequency and range are also referred to as *awareness control* protocols. In contrast to congestion control protocols which adjust transmission parameters based on the network layer metrics, e.g., channel busy ratio, awareness control protocols adjust transmission parameters based on *awareness metrics*, which base on mutual awareness and have potential to link network and application layers. Some approaches to link network and application layers or awareness metrics are discussed below.

The research community has undertaken various steps to extend network performance metrics towards the application layer. One of the first steps was the work of Bai et al. [BK06] that introduced a notion of *application level reliability* of a DSRC communication. In other words, the authors introduced an analytical model to relate the *T-window reliability metric*, also called application level reliability, with network performance metrics, also called communication reliability. The T-window reliability is defined as "the probability of successfully receiving at least one single packet from neighbor vehicles during the tolerance time window T". This probability is related to communication level reliability or probability of packet reception as Binomial probability for Bernoulli trials. The application level reliability, although it refers to the application, based on our definition, represents an awareness metric, which should be further linked to a concrete application.

The work of Gozalvez et al. [GS07] extends the application level reliability of [BK06] by accounting for the distance, named *critical distance* at which a certain level, e.g., 99 %, of application reliability should be achieved. The critical distance is the kinematic distance that is required by a vehicle to stop assuming constant deceleration. Thus, an application is required to be aware of the traffic situation (in [GS07] during an intersection approach) before crossing the distance which allows safe deceleration. In addition, a protocol called OPportunistic-driven adaptive RAdio resource Management (OPRAM) has been suggested, which adapts the transmission parameters based on the vehicle's position and the calculated critical distance. In later work of Sepulcre et al. [SGHH10] the authors utilize the same concept for a Lane Change Assistance application and define the application reliability metric as the probability of receiving at least one CAM before reaching a warning distance, a distance where application should warn the driver. The application requirements are then expressed as the warning distance and probability of at least 99 % that at least one CAM is received. The authors also argue that the same application reliability results from receiving a CAM at slightly larger or at much larger distance than the warning distance. As different transmission parameters combination can fulfill the application requirements, the

authors suggest to use the one minimizing the channel load to avoid congestion.

The analytical bounds for a maximum acceptable *message delivery latency*, for a collision warning application, and the corresponding minimum required retransmission frequency have been discussed in [Nek09]. The proposed model utilizes equations of motion of affected vehicles and outputs the maximum delivery packet latency for which a collision can be avoided. The required rebroadcast frequency is then calculated based on the similar concept of application reliability provided in [BK06]. The authors evaluated several scenarios with various vehicle speeds and road grip coefficients. The authors of [NPN11] proposed similar model that characterizes delay requirements to avoid a collision.

The latency measure is also addressed in [EGH+06] in the form of a packet *inter-reception time (IRT)* metric, that is defined as the time elapsed between two successive successful reception events. The simulation study shows the development of IRT in one specific simulation scenario. The adaptation of transmission parameters are studied to further minimize the IRT values. Similar time metrics is used by [KGRK11] in the form of *information age*. An application layer rate control algorithm is suggested to minimize the observed information age.

The transmission power control algorithm of [GSKB07] includes mechanisms to fulfill safety application requirements that are presented in the form of a *target range*. If some vehicles receive beacons outside the target range, the sender reduces the transmission power, in the opposite case, the transmission power is increased. In such a manner the presented protocol also avoids unnecessary channel congestion. The authors of [LP11b] utilized similar concept called *region of interest*, which can be defined by different applications. The goal is to adjust transmission parameters in such a way that the number of *invisible neighbors* inside the region of interest is minimized. A neighbor is considered invisible if no packets were received from this neighbor for a specific time interval. The simulations showed that combination of large transmission ranges and small packet generation rates lead to the lowest number of invisible neighbors. The impact of modulation scheme, vehicle density, and propagation models is also discussed.

The works of [RSK07], [SLS+10], and [HFSK10] use a *position accuracy* metric for which transmission parameters are adjusted to achieve certain level of position accuracy. The communication scheme presented in [RSK07] triggers transmissions of packets when the position error, that other vehicles predict of the transmitter, is above a certain threshold. The premise is that all vehicles run the same position prediction algorithm. Such approach allows controlling transmission rate based on the desired position accuracy by different applications. This neighbor tracking principle was further extended in [HFSK10] by considering transmission range adaptation. The authors of [SLS+10] pursue a goal to reduce channel load and provide the best possible position information accuracy for safety applications. Average and maximum position errors are quantified based on the beacon rate and the vehicle's velocity. The results confirm that high velocities and low transmission rates result in larger position errors compared to when higher transmission rates are used. In addition, high transmission rates have restricted impact on increasing position accuracy as velocity

increases. The authors discuss adaptation of transmission parameters based on own vehicle movement or on the movement of neighboring vehicles in consideration with the channel load. It is also stated that safety applications need to define their requirements on position accuracy for particular situations. Similarly, ETSI defines a message generation rule which accounts change in position of a host vehicle as well as radio channel state [AAE+14].

In [SLL+10] authors define an awareness metric called *awareness quantile*, which is defined "as the relation between knowledge of vehicles that is stored in a vehicle's neighbor table and the knowledge of vehicles that should be stored". In order to satisfy a desired awareness quantile, similarly as with the concept of invisible neighbors [LP11b], the adjustment of transmission parameters is foreseen.

Discussion

Research community has accumulated a lot of valuable knowledge regarding performance, behavior, and limits of vehicular communication networks. Various analytical and empirical models describing network performance aid further development of communication-based driver assistance systems. However, as it has been numerously pointed out, network level performance is insufficient to judge on application performance or on the traffic impact [EGH+06], [BK06], [AGH11]. The congestion control protocols, although keep channel congestion under control, may have detrimental impact on application performance and consecutively on the traffic [SGAK14]. Thus, awareness control protocols or metrics extending beyond standard network performance metrics have been proposed, e.g., target range [GSKB07] and position accuracy [SLS+10], [HFSK10]. In these works, applications are assumed to be able to express their requirements using these metrics, e.g., target range of 100 m, which might not always be practical and realistic. With this respect, several works integrate concrete application designs in order to be able to provide insights on application performance or the impact on traffic [vEWKH09], [SGAK14].

The analytical, empirical, and simulation models allow to predict network performance based on the initial conditions, and in combination with appropriate awareness metrics can aid further application performance evaluation. However, in reality initial parameters are not always perfectly known. For an application designer it is important to know the sensitivity of accurate information regarding the input factors to design an application that can safely function in realistic environment. This aspect is addressed within the scope of this thesis and is described in the following Section 3.3.

Also, within the scope of this thesis, *an awareness principle* is generalized to connect network and application layers. The interrelation of network parameter configuration and *awareness* are discussed in detail. If an application can express its requirements with the help of the awareness principle, the required transmission parameters can be determined; in Chapter 4 we show how awareness principle can be used to translate RECA application requirements to communication requirements. The awareness principle is discussed in Section 3.4.

3.3 Sensitivity analysis

The main assumption taken by network performance models is that the initial parameters, such as the number of transmitting vehicles, their transmission range and rate, as well as radio fading conditions, are known. In reality, without central coordination, this information is either limited, inaccurate, or not known at all. Application designers need to be able to estimate network performance, in order to either adjust transmission parameters or to integrate fall-back mechanisms in the application design, in case network performance degrades beyond what can be tolerated by safely-critical applications. In particular, depending on the radio channel conditions, opposite measures are necessary, e.g., to reduce channel congestion, the transmission range, rate, or both, need to be reduced. However, in the case of bad radio channel conditions, the transmission range and rate are typically increased. This section answers the question of how accurate the information on network and radio channel conditions needs to be and in which situations this information might be critical or not. In particular, additional efforts to gain perfect information might be unjustified, so the expected error on estimating network performance should be quantified. A sensitivity analysis can reveal which conditions have the largest impact on the network performance and for this reason need to be estimated with higher care. Similarly, conditions that have least impact on the network performance do not require extra efforts in detecting and estimating them.

Parts of this section have been previously published under the title "Accurate knowledge of radio channel and network conditions – When does it matter?" in [AMSETM11].

Sensitivity analysis (SA) "studies the relationship between information flowing in and out of the model [...] it is the study of how the variation in the output of a model (numerical or otherwise) can be apportioned, qualitatively or quantitatively, to different sources of variation, and of how the given model depends upon the information fed into it" [SCS00]. The majority of the SA methods are based on derivatives: "the derivative $\frac{dY_j}{dX_i}$ of an output Y_j versus an input X_i can be thought of as a mathematical definition of the sensitivity of Y_j versus X_i" [SRA+08].

If the number of input parameters increases, the sensitivity analysis becomes challenging. Statistical toolboxes, such as Sampling and Sensitivity Analysis Tools (SaSAT) [HRW08], are developed to ease the analysis for models with moderate to large number of input parameters. The SaSAT is built in and utilizes algorithms contained in the MATLAB® Statistics Toolbox [Thea]. The tool offers several methods, e.g., the factor prioritization by reduction of variance, which is a statistical method for ranking the importance of variables that contribute to particular outcomes. The objective of this method is to identify the input factor which, if determined, would lead to the greatest reduction in the variance of the output variable of interest. The ranking of importance is thus nothing else as ranking of factors that lead to greatest reduction of the variance of the output. The factor prioritization methods assigns the sensitivity index to values that range between 0 and 1. The higher the index is, the more important the parameter is. Such a sensitivity index is easy to interpret, and the method is independent of the model type.

Input factors	Range (units)
Vehicle density	[20:20:180] (vehicle/km)
Transmission range	[100:100:1000] (meter)
Packet size	[100:100:900] (bytes)
Packet generation rate	[2:2:14] (Hz)
Sender-receiver distance	[0:5:500] (meter)
Radio channel conditions	TRG; Nak-3; Nak-1

Table 3.1: Input factors

In the scope of this thesis, we perform a sensitivity analysis with the factor prioritization method by SaSAT [HRW08] and with a generic method based on derivatives. In our sensitivity analysis we investigate which input factors and under which conditions influence the network performance metric, called probability of packet reception (PPR), the most. The analysis is based on the data gained via simulations of vehicular networks performed in NS-2 by Schmidt-Eisenlohr [SE10]. The simulation study of [SE10] calculates the probability of packet reception in multiple scenarios with varying input configurations. The varying input configurations are: vehicle density, transmission range, packet size, packet generation rate (PGR, similar to transmission rate), sender-receiver distance, and radio channel conditions. The scenario layout represents a highway segment with evenly distributed vehicles. Each vehicle broadcasts periodic awareness messages with the most robust data rate of 6 Mbps [JCD08]. The radio conditions varies between non-fading environment modeled by Two-Ray Ground (TRG) model and fast-fading environment modeled by Nakagami-m with $m \in \{1, 3\}$ (Nak-1, Nak-3). Other input factors values cover most of the realistic possible ranges and are summarized in Table 3.1, where the minimum value, the interval, and the maximum values are provided, separated by a colon.

Sensitivity analysis based on the factor prioritization method

Although only limited number of variables is considered for sensitivity analysis of network performance, the SaSAT is used with its factor prioritization method as it allows to perform fast statistical analysis. The input parameters, summarized in Table 3.1, as well as the corresponding output in the form of the probability of packet reception, have been fed to the SaSAT. The tool assigns sensitivity indexes to the input factors depending on their influence on the output. Table 3.2 summarizes the resulting sensitivity indexes provided by SaSAT. As expected, the distance between sender and receiver and the transmission range are the most important or influential factors for the PPR, i.e., have the highest index values. This has been observed for all three radio channel conditions, as seen in Table 3.2.

Sensitivity analysis based on the factor prioritization method allows to see relative importance of each input condition but it does not reveal exact sensitivity to a variation or indicates which situations lead to the highest sensitivity.

Factors	Sensitivity index		
	TRG	Nak-3	Nak-1
Vehicle density	0.128691	0.103160	0.074259
Transmit range	0.304099	0.326273	0.343989
Packet size	0.076241	0.059083	0.047871
Packet generation rate	0.112486	0.082571	0.072684
Sender-receiver distance	0.378481	0.428911	0.461194

Table 3.2: Sensitivity analysis based on the factor prioritization method

Sensitivity analysis based on derivatives

In order to determine sensitivity to input variations and situations that lead to the highest sensitivity, we perform a detailed analysis individually with respect to each input factor. The distance between sender and receiver is taken out of the direct sensitivity analysis. From the perspective of broadcast applications, due to multiple receivers, there is no one distance that could be estimated. It is more important to know the vehicle density in the surrounding and their packet generation rate.

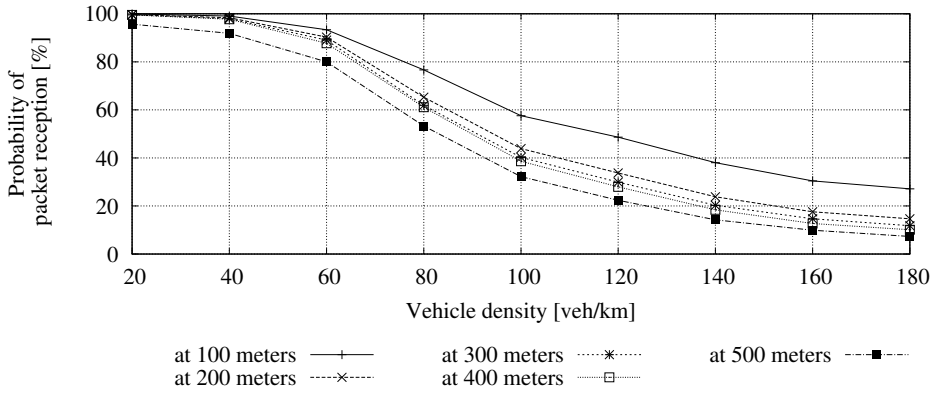
For the detailed sensitivity analysis one input factor is varied at a time (shown on the x-axis) while the remaining factors are kept fixed in order to identify the most sensitive areas for the PPR (shown on the y-axis). The distance between sender and receiver is considered only indirectly when the impact of variation of one input factor, other than the distance between sender and receiver, is observed for the PPR at various distances between sender and receiver. The PPR at following sender-receiver distances is observed – 100, 200, 300, 400, and 500 m. The ranges and intervals of varying input factors are the same as summarized in Table 3.1. For each two successive points in each PPR curve the slope of a secant line is calculated. The absolute value of the calculated slope is then depicted in a figure indicating the average change rate but not the direction of the slope. The resulting change range of PPR curves can be interpreted as *a variation on the x-axis by* 1 *unit leads to an absolute change of the PPR by n percentage points (% points)*. The change rate value of zero can be interpreted as "no sensitivity is observable" and any value greater than zero as "the PPR is sensitive to a variation of 1 unit on the x-axis by *n* percentage points". Such approach allows not only to identify the most sensitive situations but also to quantify the amount of increase/decrease of PPR due to variations of the input factors.

Based on the methodology presented above, the detailed sensitivity analysis is performed on all possible input configurations. Two cases can be identified:

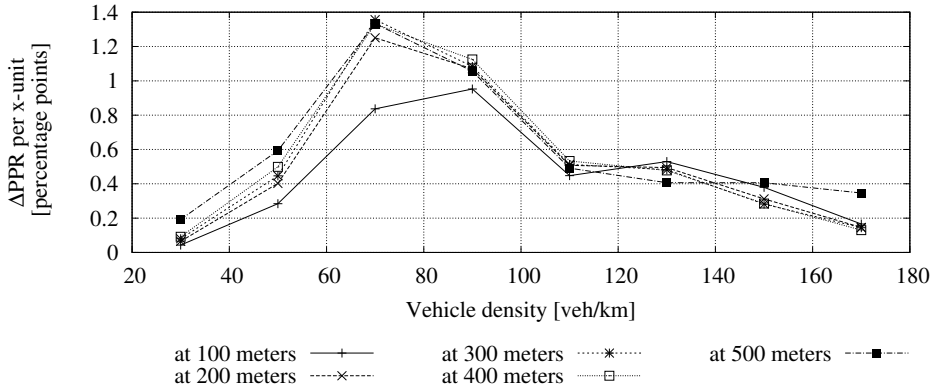
Case 1: High sensitivity is observable under high network load conditions and mainly independent from the distance between sender and receiver.

Case 2: High sensitivity is observable even under low network load conditions if the distance between sender and receiver is close to the maximum communication range.

In the following, first the results for each input factor, except radio channel conditions, that are based on the TRG radio model are presented. In addition, the two cases described above are demonstrated. Finally, the results of sensitivity analysis



(a) PPR vs. Vehicle density



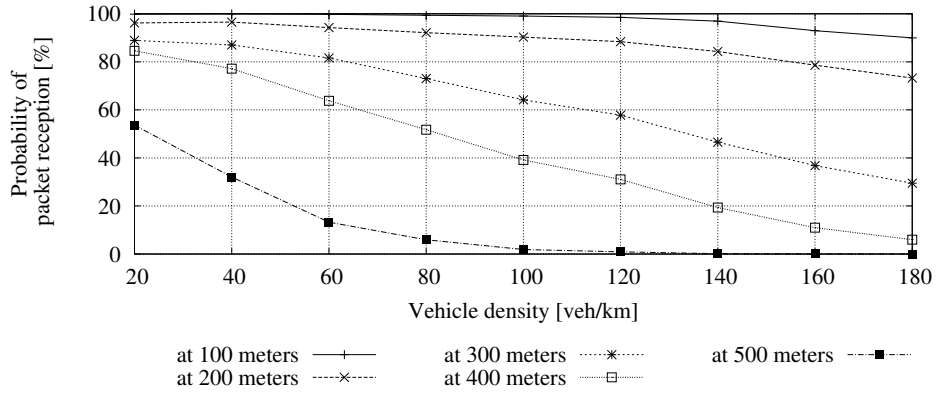
(b) Change rate of PPR vs. Vehicle density

Figure 3.4: Sensitivity analysis with respect to vehicle density – Case 1. Packet size is 500 bytes, Tx. range is 1000 m, PGR is 10 Hz, min. network load is 1.6 Mbps, max. network load is 14.4 Mbps

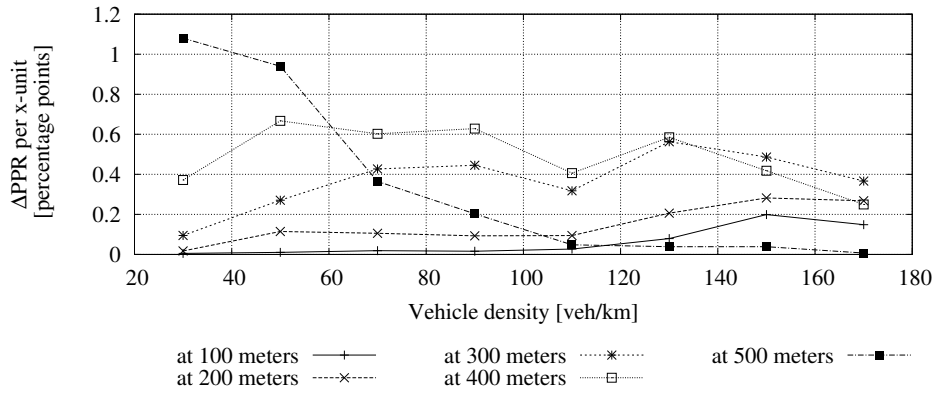
with respect to different radio propagation models, namely, TRG and Nakagami-m with $m \in \{1, 3\}$ are compared with each other.

Sensitivity analysis with respect to vehicle density. The sensitivity analysis with respect to vehicular density and the identified Case 1 are shown in Figure 3.4. To show the conditions of Case 1, i.e., the high network load, the packet size is fixed to 500 bytes, the transmit range to 1000 m, and the PGR to 10 Hz. Together with a vehicle density varied between 20 veh/km and 180 veh/km these configurations translate to network load values between approx. 1.6 Mbps and 14.4 Mbps. As can be seen in Figure 3.4(a), the PPR starts at approx. 100 % for all distances between sender and receiver and drops significantly as soon as the vehicle density exceeds 60 veh/km. In Figure 3.4(b) the change rates of the PPR curves are shown. The highest change of rate occurs between 60 and 80 veh/km, which in terms of network load corresponds to approx. 5.6 Mbps. Moreover, the change rates of curves behave similar for various distances in this scenario configuration.

From the change rate curve one can also quantify the impact of having an imprecise



(a) PPR vs. Vehicle density

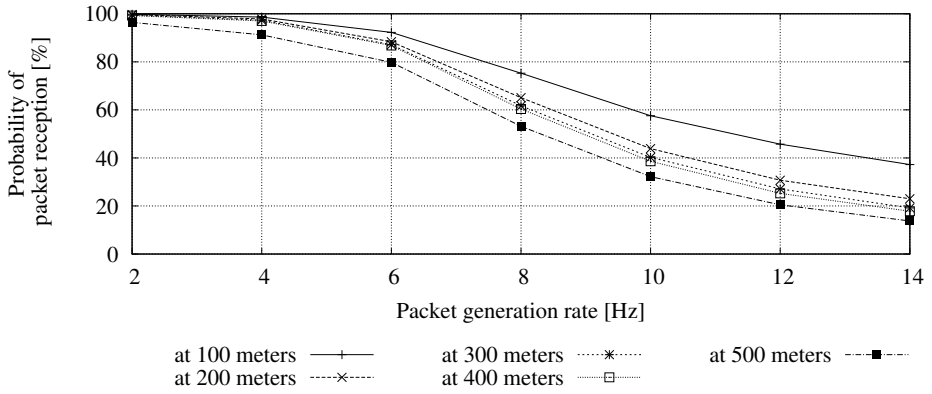


(b) Change rate of PPR vs. Vehicle density

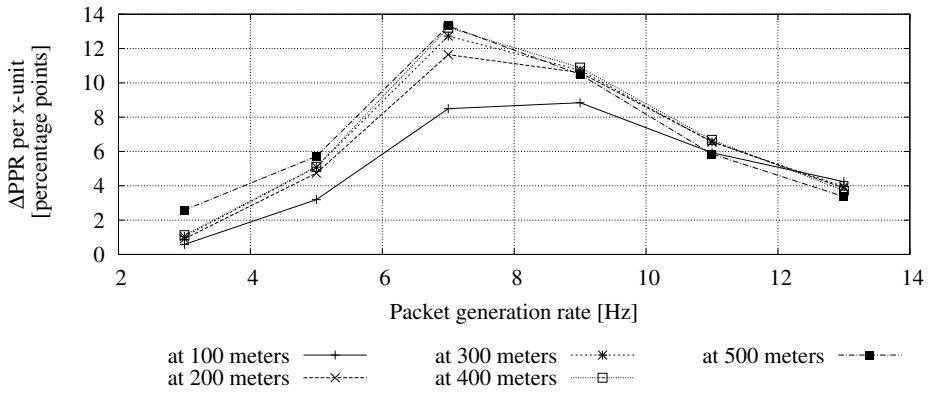
Figure 3.5: Sensitivity analysis with respect to vehicle density – Case 2. Packet size is 500 bytes, Tx. range is 500 m, PGR is 6 Hz, min. network load is 0.48 Mbps, max. network load is 4.32 Mbps

knowledge, or uncertainty, about the vehicle density. For example, the min. change rate of 0.04 *% points* can be observed for distance of 100 m between sender and receiver and a density between 20 and 40 veh/km. That means that the variation of vehicle density by 1 veh/km would result in the variation of the PPR of roughly 0.04 *% points*. The max. change rate of 1.36 *% points* can be observed for a distance of 300 m and a density between 60 and 80 veh/km. Under the assumption that the density information is provided with a max. error of 10 %, a deviation in the PPR estimation of 0.12 *% points* for the min. change rate case and 9.52 *% points* for the max. change rate case has to be expected. While it should be negligible whether the PPR will be 99.99 % or 99.87 % as for the min. change rate case, it might be significant whether the PPR will be 89.00 % or 79.48 % as for the max. change rate case (the change rate and the PPR values are taken from the corresponding points in Figure 3.4).

The Case 2 is presented in Figure 3.5(a) and Figure 3.5(b). In the underlying scenario the network load is never saturated, even for high vehicle density. However, the limited transmission range of 500 m introduces sensitivity, especially at distances



(a) PPR vs. Packet generation rate



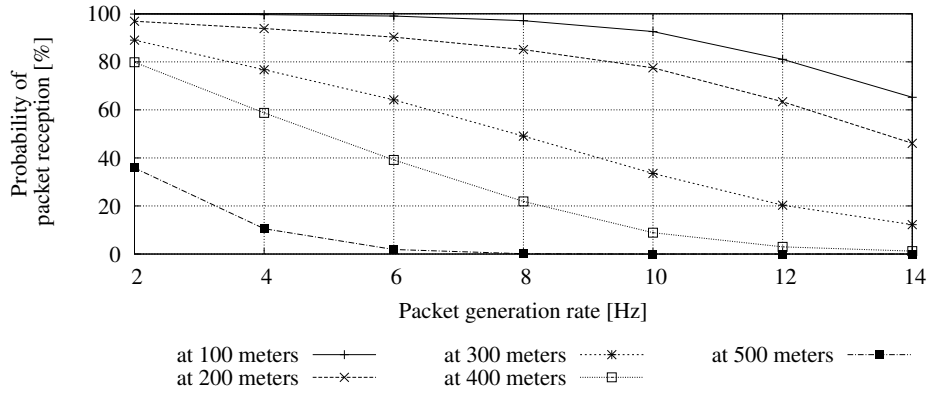
(b) Change rate of PPR vs. Packet generation rate

Figure 3.6: Sensitivity analysis with respect to the packet generation rate – Case 1. Packet size is 500 bytes, Tx. range is 1000 m, vehicle density is 100 veh/km, min. network load is 1.6 Mbps, max. network load is 11.2 Mbps

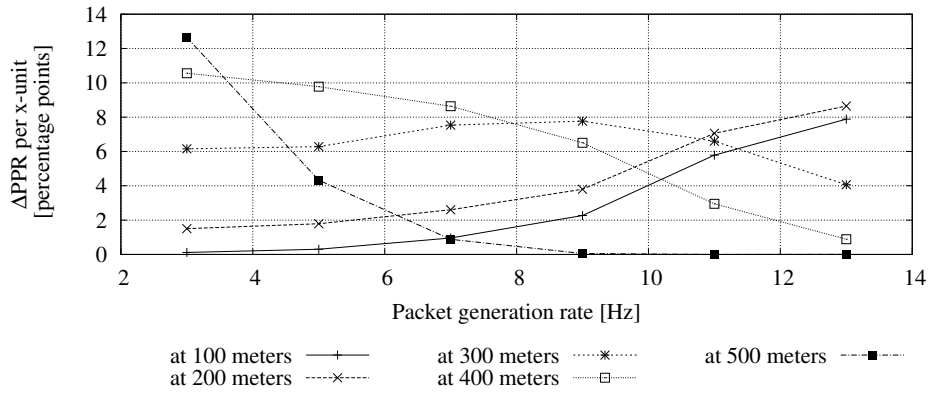
between sender and receiver above 200 m. As expected, only a small sensitivity can be observed for distances up to 200 m. Thus, the accurate estimation of vehicle density for short distances is not crucial, if the transmission range is significantly larger. The closer it gets to the border of transmission range the more prominent is the sensitivity with respect to the density and an accurate estimation might be required.

Sensitivity analysis with respect to the packet generation rate. Sensitivity analysis for the packet generation rate variation is illustrated in Figure 3.6 and Figure 3.7. For the Case 1 of sensitivity under a high network load the same values for the transmit range and packet size are fixed as for the vehicle density analysis. In its turn, the vehicle density was fixed to 100 veh/km, which leads to the network load of 1.6 Mbps for the min. PGR and up to 11.2 Mbps for the max. PGR, see, e.g., Figure 3.6(a).

The min. change rate of 0.58 *% points* is observed at the distance of 100 m and PGR between 2 and 4 Hz, as shown in Figure 3.6(b). The max. change rate of 13.31 *% points* can be seen at the distance of 500 m and for the PGR between 6 and 8 Hz. An error in the PGR estimation of 10 % will result in 0.17 *% points* of the PPR deviation for the



(a) PPR vs. Packet generation rate



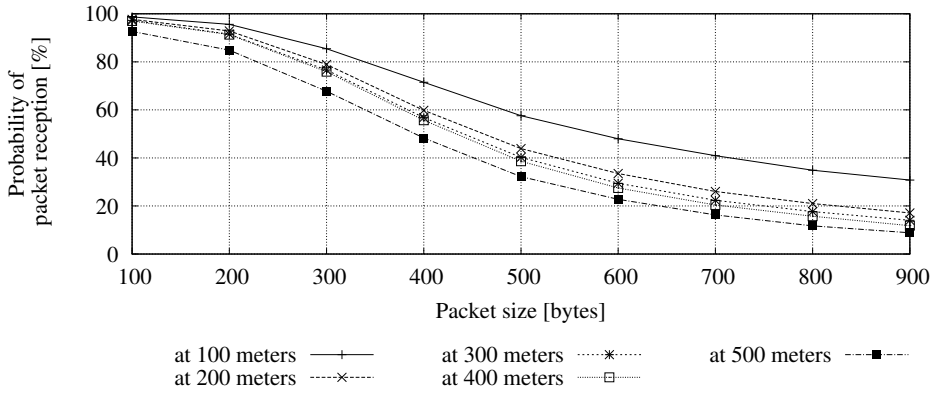
(b) Change rate of PPR vs. Packet generation rate

Figure 3.7: Sensitivity analysis with respect to the packet generation rate – Case 2. Packet size is 500 bytes, Tx. range is 500 m, vehicle density is 100 veh/km, min. network load is 0.8 Mbps, max. network load is 5.6 Mbps

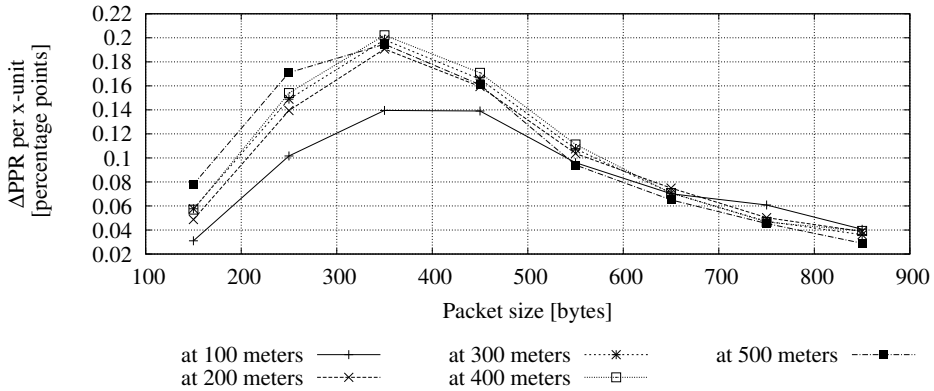
min. change rate case and in 9.32 *% points* of PPR variation for the max. change rate case. These translate into the small PPR deviation from 99.66 % to 99.49 % for the min. change rate case and larger PPR deviation from 79.80 % to 70.48 % for the max. change rate case, indicating the significance of the correct estimation of the PGR under higher network load even if the transmission range is large.

The Case 2 with a moderate network load is illustrated in Figure 3.7. One can observe high sensitivity for the distances that are close to the transmission range (400 and 500 m) even for a low network load. The sensitivity for short distances (up to 200 m) is negligible for the low PGR but increases significantly as the PGR increases above 8 Hz.

Sensitivity analysis with respect to the packet size. The sensitivity analysis with respect to the packet size is inline with those already shown (considering the shape of the slopes) and is presented in Figure 3.8 and Figure 3.9. In Case 1 of high network load, the PPR starts close to 100 % for small packet sizes and decreases for larger packet sizes. Higher reception probabilities are observed for smaller packet sizes as network load is smaller. The smallest change rate of 0.03 *% points* is determined



(a) PPR vs. Packet size



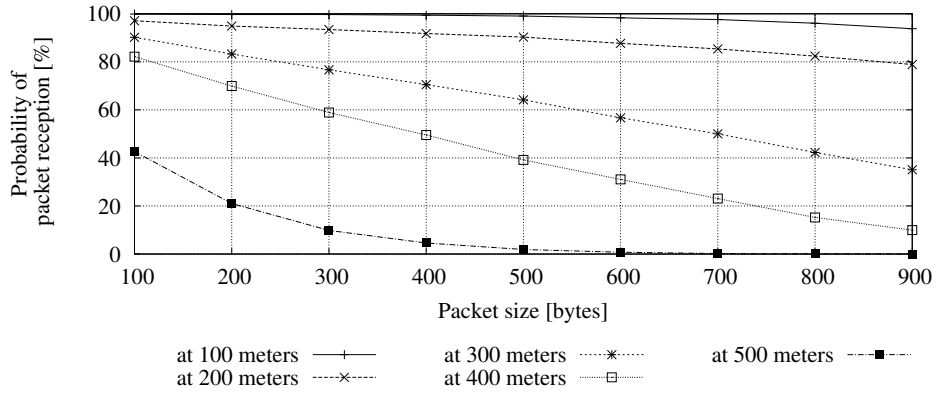
(b) Change rate of PPR vs. Packet size

Figure 3.8: Sensitivity analysis with respect to the packet size – Case 1. Tx. range is 1000 m, vehicle density is 100 veh/km, PGR is 10 Hz, min. channel load is 0.96 Mbps, max. channel load is 8.64 Mbps

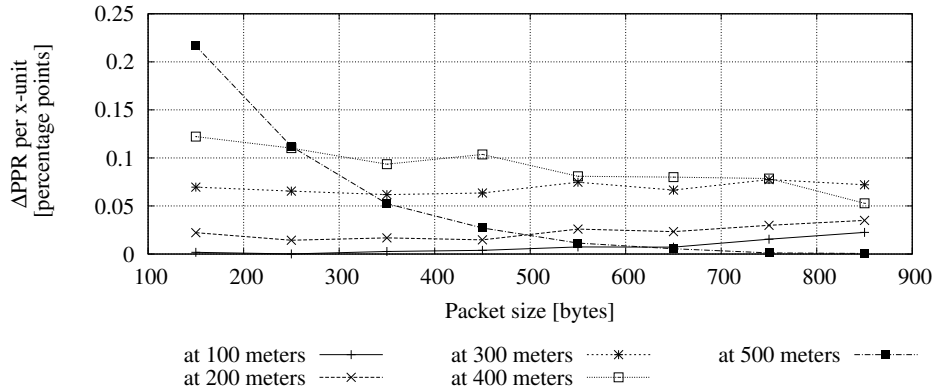
for packet size between 100 and 200 bytes and for the distance between sender and receiver of 100 m, whereas the highest change rate of 0.20 *% points* is observed for packet sizes between 300 and 400 byte and for the distance of 400 m. The analysis with respect to the 10 % deviation from the actual packet size lead to a change of PPR of 0.45 *% points* at minimum and of 7.00 *% points* at maximum. The conditions that lead to the max. change rate might have considerable impact on the expected PPR and consequently on envisioned applications.

In Case 2, i.e., when channel load is low, it can be observed that sensitivity with respect to packet size is increasing with distance between sender and receiver. For each individual distance, however, the sensitivity remains nearly constant over all packet sizes. In consequence, packet size does not have to be determined precisely in case of low channel load.

Sensitivity analysis with respect to the transmission range. In the sensitivity study with respect to the transmission range, cf. Figure 3.10, the scenario configurations are fixed to 500 bytes for the packet size and PGR of 10 Hz. This, together with a



(a) PPR vs. Packet size

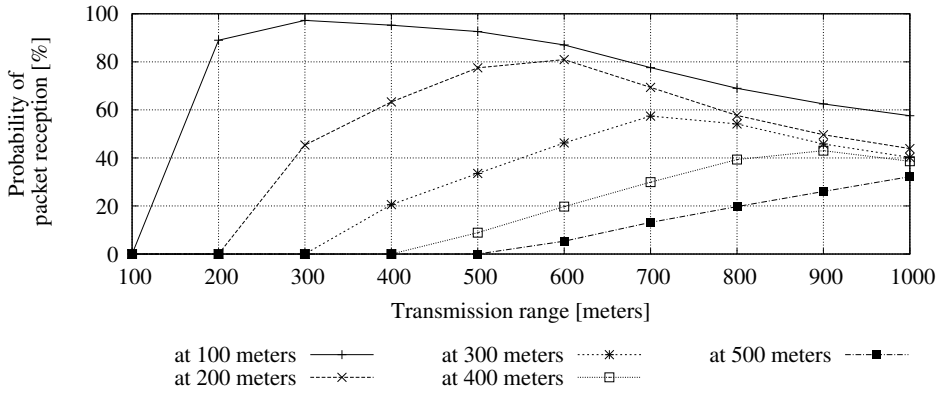


(b) Change rate of PPR vs. Packet size

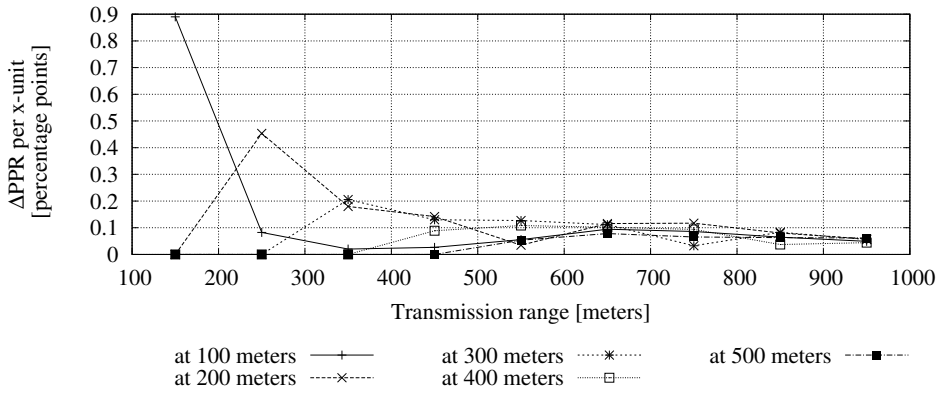
Figure 3.9: Sensitivity analysis with respect to the packet size – Case 2. Tx. range is 500 m, vehicle density is 100 veh/km, PGR is 6 Hz, min. channel load is 0.48 Mbps, max. channel load is 4.32 Mbps

fixed vehicle density of 100 veh/km and the varying transmission range, translates to network load between 0.8 Mbps and 8 Mbps.

As can be seen in Figure 3.10(a), the PPR raises above zero as soon as the transmission range is greater than the considered distance between sender and receiver. Since the load in the network is smallest for the lowest transmission range configuration, the PPR is most sensitive for small transmission range selections and small distances between sender and receiver, which is the identified Case 1. Furthermore, it can be noted that the PPR increases with an increase of the transmit range. However, the PPR reaches a maximum (with respect to transmission range) at some point and decreases again afterward. In other words, an optimal transmit power selection exists and one can not simply increase the transmit power to further increase the PPR. This observation is expected, since an increase of the transmit power leads to an increase of the network load, and once the network reaches its saturation point, performance decreases again. Note also, that this saturation point is not equal for all distances between sender and receiver, thus, one cannot simply tune control algorithms to



(a) PPR vs. Transmission range



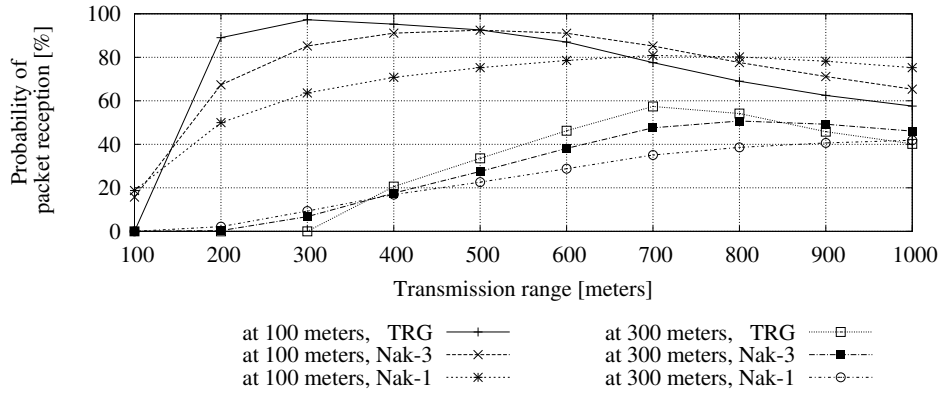
(b) Change rate of PPR vs. Transmission range

Figure 3.10: Sensitivity analysis with respect to the transmission range – Case 1. Packet size is 500 bytes, vehicle density is 100 veh/km, PGR is 10 Hz, min. network load is 0.8 Mbps, max. network load is 8 Mbps

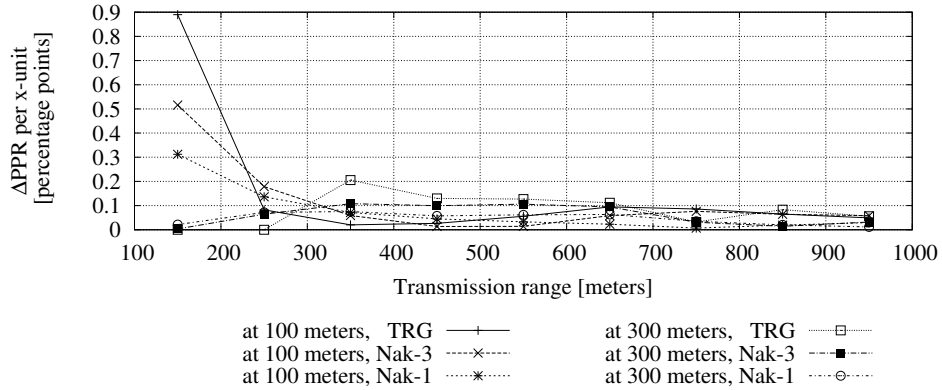
limit the network congestion to a fixed value.

Figure 3.10(b) quantifies the sensitivity as follows: for the min. change rate of 0.03 *% points* which can be observed at a distance of 200 m and a transmit range between 700 and 800 m, a max. error of 10 % in the detection of the average transmission power by neighboring vehicles, the error of the estimated PPR will be 2.25 *% points*. Likewise, at the max. change rate of 0.89 *% points*, at a distance between sender and receiver of 100 m and a transmit power between 100 and 200 m, a detection error of 10 %, will result in the 13.35 *% points* error of the estimated PPR.

Sensitivity analysis with respect to the radio channel conditions. Due to similar behavior, only the example of varying transmission range is shown to demonstrate how different radio channel conditions impact the estimated PPR. The identical setup as in Figure 3.10 is used, but only a distance of 100 and 300 m between sender and receiver is shown. A non-fading radio channel (TRG) and a fading channel with either a small fading intensity (Nakagami-m, $m = 3$) or a large intensity (Nakagami-m, $m = 1$) are depicted.



(a) PPR vs. Transmission range



(b) Change rate of PPR vs. Transmission range

Figure 3.11: Sensitivity analysis with respect to the radio channel conditions. Packet size is 500 bytes, vehicle density is 100 veh/km, PGR is 10 Hz, min. network load is 0.8 Mbps, max. network load is 8 Mbps

In Figure 3.11 two main observations can be seen. First, the sensitivity of the average PPR with respect to the transmission range is reduced if the channel is fading (compared to the non-fading case) and even further reduced if fading intensity increases. Second, a significant sensitivity of the PPR with respect to the radio channel condition itself can be observed. In the case of a 300 m transmit range and a distance of 100 m between sender and receiver, the difference in PPR between a non-fading channel and a very fast fading channel is approx. 33.6 %, and approx. 21.6 % between small and large fading intensities. In comparison to the cases where the sensitivity was studied with respect to the network related parameters, e.g. transmission range, vehicle density, or PGR, no "rule of thumb" has been observed that can be used as a guideline for application developers. For instance, it is not possible to make a statement similar to "under high network load conditions, the sensitivity of the PPR is significant to the radio channel conditions". Instead, the radio channel conditions should always be detected or estimated very accurately, in order to derive proper estimates of the average PPR.

Discussion

Based on the performed sensitivity analysis the following guidelines can be given:

- There is no need to know the network conditions in every situation.
- Network conditions should be detected whenever the two identified cases occur, namely:
  - whenever the network is on the transition from non-saturated to saturated state
  - whenever the considered distance between sender and receiver is close to the communication range
- If radio channel conditions change from non-fading to fading the sensitivity of the probability of packet reception with respect to the network related parameters, such as packet generation rate, packet size or vehicle density, decreases.
- The radio channel conditions need to be detected in every situation.

These guidelines are based solely on the probability of packet reception metric, which characterizes network layer performance, and it is not clear whether these guidelines are still valid for other metrics, including application layer metrics. Furthermore, the results reflect only a highway scenario with uniform vehicle density distribution. As such, it is not straightforward whether this results can be transferred to different scenarios, e.g., urban or non-uniform vehicle distribution. Nevertheless, the results can be used to characterize the implications of inaccurately estimating network and radio channel conditions. Such quantification helps application and protocol designers to consider and deal with the deviations of such network layer metric as the probability of packet reception.

Vehicles themselves can aid detection and estimation of network and radio conditions, e.g., vehicles can include own transmission parameters into the periodic beacon messages, together with own observation, e.g., observed channel busy time, vehicle density, path loss, or its packet reception statistics. In such a manner, algorithms obtain a detailed feedback of each vehicle's behavior, together with information on how they perceive the current situation. Depending on application requirements, this network feedback can also be propagated over multi-hop. In addition, the knowledge on network and radio conditions may help scheduling transmission of non-safety messages by, e.g., postponing the transmission until better conditions. More information on methods to detect and estimate network and radio conditions is provided in [AMSETM11].

The identified sensitivity analysis is not further utilized within this thesis, we assume that network and radio channel information is always accurate and given, see Section 2.3.3.

3.4 Connecting network and application layers

Typical awareness metrics describe neighborhood information acquired through periodic CAM exchange. In the current section we elaborate on the idea of using *an awareness principle* to connect application and network layers. In particular, we perform a detailed analysis of the relation between defined awareness and a network performance metric, called probability of packet reception. In such a manner, if an application can express its requirements with the help of the awareness principle, the transmission parameters that will satisfy the application's requirements can be determined. The feasibility of translating the application's requirements to awareness parameters is presented in Chapter 4 on the example of Rear-End Collision Avoidance application.

Parts of the work presented in this section have been previously published in a conference paper, titled "VANET: Is 95 % probability of packet reception safe?" [AGH11].

Understanding the awareness principle

The awareness definition used in this thesis is based on the binomial probability of awareness message reception and is similar to the application reliability of [BK06] and [GS07], and the neighborhood awareness defined in [MSEK+08]. In a binomial experiment, also called Bernoulli trials, there are two mutually exclusive outputs – "success" and "failure". As both events happen with certain probability, the binomial probability of an event, "success" or "failure", occurring *exactly*, *at least*, or *at most*, n times, can be calculated. The reception of an awareness message can also be seen as a Bernoulli trial: the message is either received – "success" or not – "failure". This is similar to application reliability presented in [BK06], [GS07], and [SGHH10], where a probability of successfully receiving at least one packet during a time window is calculated, whereas authors of [GS07] and [SGHH10] also define a critical distance at which certain application reliability needs to be achieved. In addition, the authors of [MSEK+08] defined neighborhood awareness as a probability of having received at least one beacon within the past second. Since the awareness metric should be easily utilized by most of the applications, we specify the awareness definition as following:

Definition 3.1. ***The awareness probability*** *P_A is the probability of successfully receiving at least n packets in the time window T. The distance at which awareness probability is defined is referred to as awareness range R_A.*

The awareness probability P_A is calculated as in the equation below:

$$P_A = \sum_{n}^{k} \binom{k}{n} p^n (1-p)^{k-n}, \tag{3.1}$$

where p is the probability of packet reception (PPR) between sender and receiver at a certain distance, and k is the amount of packets that were sent in the time window T with a certain transmission rate. Just as the probability of packet reception, awareness probability is related to the distance between sender and receiver, i.e., its awareness range.

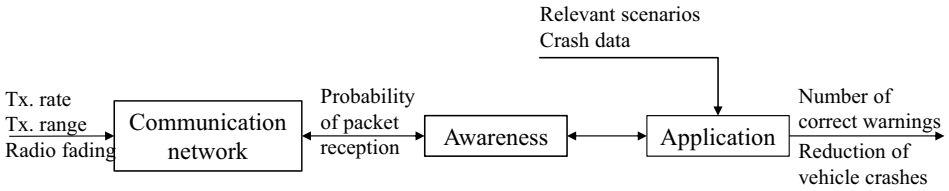


Figure 3.12: Awareness as a link between network and application layers. The bidirectional arrows between application and communication network via awareness represent the capability of awareness to translate application requirements to the required transmission parameters (contrary to Figure 3.3)

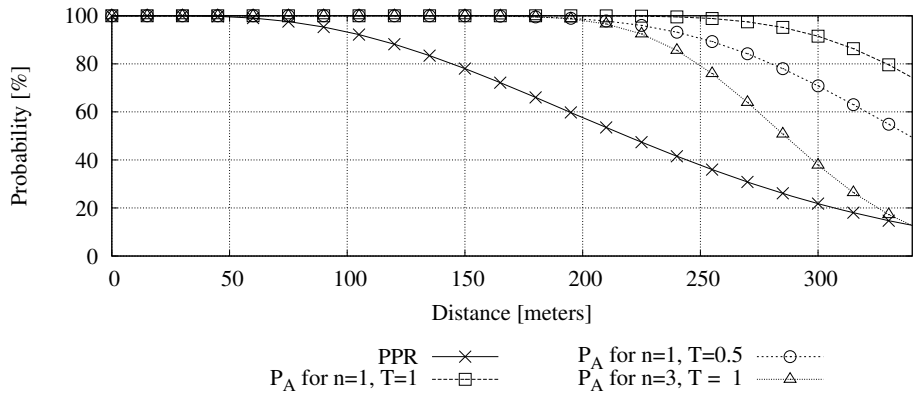
In such a manner, awareness is "located" between network and application layers, as depicted in Figure 3.12. The use of awareness allows not only to evaluate communication network on "pre-application" level but also to propagate application requirements to communication. In particular, it becomes possible to make application-aware adjustment of transmission parameters. Probability of packet reception is clearly not suitable to represent application requirements – if only one packet is sent and received, the resulting 100 % probability of packet reception will not guarantee successful operation of a safety-critical application over the course of time.

In the following, the relationship between probability of packet reception and awareness is further analyzed. Figure 3.13 graphically depicts the differences between probability of packet reception and probability of awareness. The curves correspond to a scenario with a vehicle density of 60 veh/km, a transmission range of 300 m, and transmission rates varying between 2, 6, and 10 Hz. The PPR curve is obtained with the help of the empirical communication model of Killat et al. [KH09][3] and the awareness probability for different number of packets n and time window T is based on Equation 3.1.

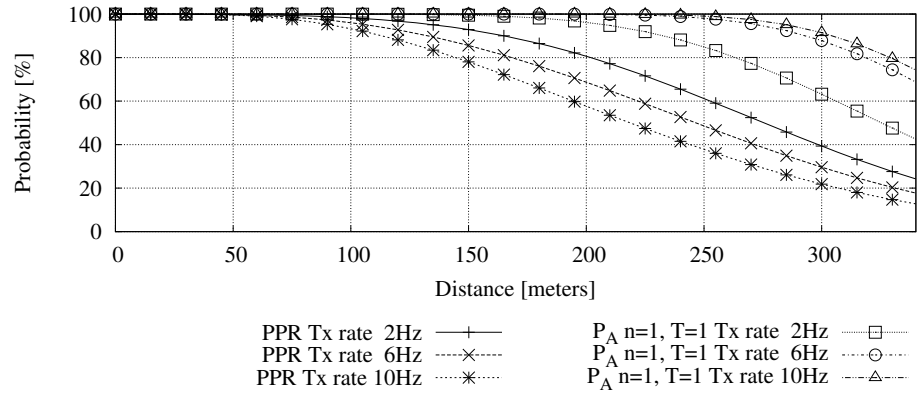
As can be seen in Figure 3.13(a) the PPR degrades very fast at distances above 60 m, whereas the awareness probability P_A, for different time windows T and numbers of packets n, is close to 100 % up to the distance of 200 m. Naturally, if the number of packets n is large and the time window T is small the resulting awareness range is smaller than for the case when n and T are more relaxed. This shows that in situations when the PPR is not favorable (e.g., less than 70 %) nodes still might have nearly perfect (in this case greater than or equal to 99 %) awareness.

In Figure 3.13(b) one can observe PPR curves for different transmission rates and the corresponding awareness probability P_A, for the number of packets $n = 1$ and time window $T = 1$. In the considered scenario an increase of transmission rate degrades PPR but, nevertheless, increases the awareness range. For example, the awareness probability P_A for all configurations is nearly the same and approximately or greater than 99 % for the awareness range of up to 150 meters, whereas PPR is rapidly degrading at distances above 50 m. This observation, inline with [SGHH10], shows that the same awareness range and the same requirements on the number of

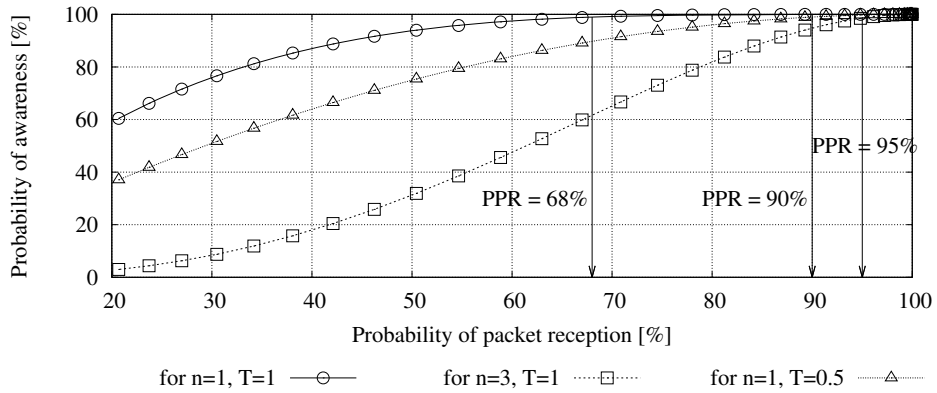
[3]The values for probability of packet reception can also be taken from other analytical, empirical, or simulation models, e.g., [SE10].



(a) The PPR and the corresponding P_A with various T and n parameters. Vehicle density is 60 veh/km, transmission rate is 10 Hz, transmission range is 300 m



(b) The PPR for various transmission rates and the corresponding P_A. Vehicle density is 60 veh/km, transmission range is 300 m, transmission rates are 2, 6, and 10 Hz



(c) Required PPR for various awareness parameters. Vehicle density is 60 veh/km, transmission rate is 4 Hz, transmission range is 300 m

Figure 3.13: Correlation of awareness probability and PPR under different network conditions and awareness parameters

received packets n in the time window T can be achieved by different transmission rates and thus with different PPR values.

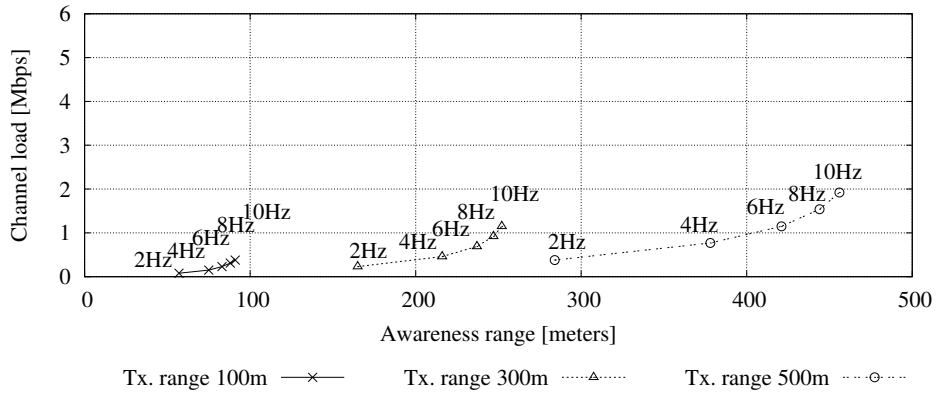
Figure 3.13(c) depicts more prominently the correlation between PPR and awareness calculated according to Equation 3.1. In order to achieve an awareness probability P_A of greater than or equal to 99 %, the required packet reception probability has to be about 68 %, 90 %, and 95 %, for cases when the number of packets n and time window T parameters are varied as following: $n = 1$ and $T = 1$; $n = 1$ and $T = 0.5$; $n = 3$ and $T = 1$, respectively. Hence, depending on the awareness requirements, a PPR of as low as 68 % can be sufficient or a PPR as high as 95 % can be necessary.

In order to further understand the relationship between transmission parameters and awareness, two scenarios with different vehicle densities have been studied. Figure 3.14 shows the maximum awareness range (shown on the x-axis) which can be achieved for various configurations of transmission range and transmission rate as well as the corresponding channel load (shown on the y-axis). The required awareness parameters are set as following: the number of packets is $n = 1$ in a time window $T = 1$, and the desired awareness probability P_A is greater than or equal to 99 %. The two vehicle density configurations are 60 veh/km and 180 veh/km. Just as for the previous figure the PPR values have been obtained with the help of the empirical communication model of Killat [KH09] and the awareness range calculated analytically based on Equation 3.1. It can be seen that in the scenario with low vehicular density, cf. Figure 3.14(a), an increase in transmission range and transmission rate increases the awareness range with only a slight increase of the channel load. However in the case of the scenario with higher vehicular density, the awareness range no longer consequently continues to increase with an increase of transmission rate and even degrades leading to a higher increase of the channel load. In particular, in Figure 3.14(b) for a transmission range of 500 m and transmission rate of 10 Hz, the awareness range is not increased but even lower (at approx. 170 m) than for lower transmission rates of 2–4 Hz (approx. 280 m), even though the channel occupancy is below the limit of 6 Mbps. A lower transmission range of 300 m leads to a similar awareness range (of approx. 170 m) but with much smaller load on the channel.

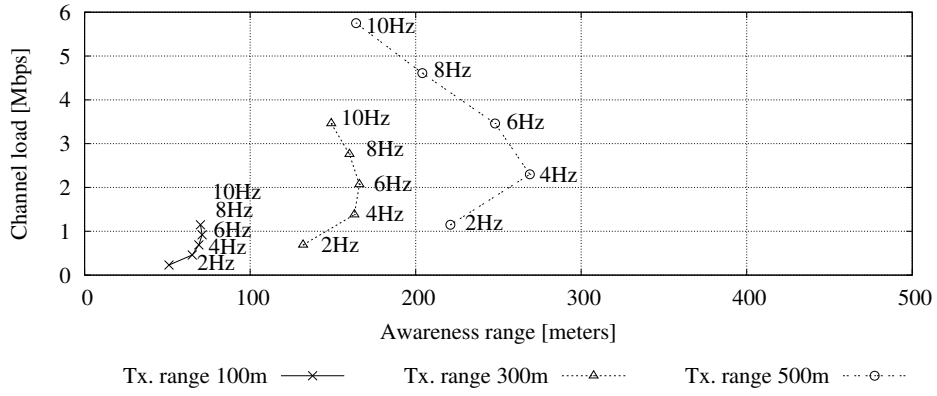
If the radio fading is more severe, i.e., follows a Nakagami-m distribution with $m = 1$, the awareness range degrades similarly with the increase of the channel load[4]. The same scenario configurations as in Figure 3.14 but with severe fading conditions result in approx. 2/3 of awareness range when radio conditions follow Nakagami-m with $m = 3$.

As a general trend for the studied transmission ranges there is no further increase in the awareness range due to the increase of transmission rate after the channel load is greater than 2.5–3 Mbps. Thus, there exists an optimum communication configuration with respect to the channel load that results in a certain maximum range where vehicles are aware of each other (with certain awareness probability P_A). This depicts the limits of communication – if the vehicle density is high the resulting channel load is prohibiting the fulfillment of the awareness requirements.

[4]based on the simulation results of [SE10].



(a) Vehicular density 60 veh/km



(b) Vehicular density 180 veh/km

Figure 3.14: Maximum awareness range vs. channel load. Transmission rate is between 2 and 10 Hz, fading is following Nakagami-m with $m = 3$, $P_A \geq 99$ %, $T = 1$, and $n = 1$

Discussion

In the current section we defined an awareness metric called awareness probability based on the binomial probability and existing definitions of awareness. Awareness probability P_A defines the probability of receiving at least n number of packets in the time window T at a certain distance, called awareness range R_A. In addition, we analyzed the relation between probability of packet reception and awareness probability. We showed that there exists an optimum configuration of transmission rate and range for a maximum awareness range and a minimum channel load.

Several limitations regarding spatial and temporal variation of the analytical representation of probability of packet reception must not be ignored. First, the reception probability of succeeding packets is assumed to be independent of each other and second, the probability of packet reception is assumed to stay constant within the time window T. In future work it is important to address such aspects as, e.g., burst errors.

Discussed awareness can be further fortified with "environmental awareness" that gives information on, e.g., radio conditions. For example, not receiving any beacons

could be mistakenly interpreted as absence of vehicles in the neighborhood but in combination with good environmental awareness (e.g., that indicates severe fading) it means that beacons are simply lost and neighborhood awareness is no longer reliable.

Although the awareness principle makes it feasible to adjust transmission parameters based on application requirements, it is assumed that application requirements to receive information can be expressed with the help of awareness parameters. In Chapter 4 we demonstrate how the awareness principle is utilized to determine minimum required transmission parameters to satisfy information reception requirements of a fail-safe Rear-End Collision Avoidance application. In realistic communication conditions, e.g., in the presence of burst errors, requirement of applications to receive information cannot always be satisfied even if the "right" transmission parameters have been used. For this reason, we chose to design the two applications in a fail-safe manner, which allow applications to function safely, even when their requirements on information reception cannot be satisfied. Consequently, the awareness principle can be used for the straightforward determination of optimal transmission parameters, but the application's safety should be "backed up" by the fail-safety features.

3.5 Evaluation methodology

We present a general evaluation methodology that can be followed to determine whether IEEE 802.11p communication can scale and reliably support safety-critical applications. In addition, resulting impact of applications on road traffic can be determined. Clearly more elaborated and thorough evaluations, e.g., with driving simulators or test track evaluations, are required before any of the safety-critical applications can be introduced into an actual vehicle. However, the presented methodology already gives initial indications and sheds light on possible impact a communication-based application might have on the traffic.

Performance evaluation of the designed applications can be performed analytically or via simulation analysis. The choice can be dependent on the complexity of the application as well as on the availability of appropriate models with a desired level of realism. The decision on evaluation environment is followed by identification of common scenarios in which an application is intended to function, e.g., the pre-crash scenarios.

Below are the possible steps that can be followed for evaluation of a fail-safe communication-based application:

1. **Determining minimum transmission parameters:** This step is required as an *initial reference* for further evaluation process. Application requirements can provide feedback on the minimum transmission parameters that should be used. The prerequisite for this is that application requirements can be mapped to transmission parameters. The awareness principle provides one alternative for such mapping, cf. Section 3.4.

2. **Determining the number of communicating vehicles:** The initial estimate on the number of communicating vehicles should be determined. For this, either

a realistic range can be chosen or an initial number which can be increased or decreased, depending on whether IEEE 802.11p communication manages to support all communicating vehicles or not.

3. **Determining the reliability of IEEE 802.11p communication to support an application:** Whether vehicular communication is capable of supporting an application requires considering of application requirements. In such a way, it can be determined if and how application requirements can be satisfied by communication. For each of the two safety-critical applications we exemplary define *communication reliability*, as discussed in the corresponding chapters.

4. **Determining the scalability of IEEE 802.11p communication:** If application requirements could be supported for the initial number of communicating vehicles with sufficient reliability, the number of communicating vehicles could be scaled up and the previous step repeated. Otherwise, the number of communicating vehicles could be decreased until sufficient reliability is achieved. In such a way, the scalability of IEEE 802.11p communication could be determined.

5. **Determining the impact on road traffic:** In order to determine the impact a communication-based application can have on road traffic we compare application's traffic level performance to situations with no communication-based application in place. Traffic impact can be measured on traffic safety and on traffic efficiency. Since the two considered applications are designed fail-safe, it is expected that they will not result in unsafe situations, at least, not due to the addressed failures. The resulting communication reliability only implicitly describes traffic safety (see dependability concept in Section 2.1.3). Traffic efficiency can be measured with metrics like *vehicle density* or *average travel time*. These metrics can be compared to traffic efficiency which currently occurs on roads with no communication-based applications in use.

Discussion

This chapter deals with three aspects relevant for performance evaluation of communication-based applications covered in Section 3.3, Section 3.4, and Section 3.5, along with defined open issues and related work covered in Section 3.1 and Section 3.2. In particular, Section 3.3 describes sensitivity analysis of information that is used to estimate network performance, i.e., sensitivity of information on the number of communicating vehicles or radio channel fading to estimate the probability of packet reception. We identified conditions when accurate network information is important and quantified possible errors in the network estimation which are caused by inaccurate information. This knowledge can be used by application designers to either better estimate the network information or to integrate additional mechanisms avoiding unsafe operation under inaccurate information. Although the insights acquired through this sensitivity analysis are essential for the overall design and evaluation of communication-based applications, their exact practical use is left for the future work.

Section 3.4 describes the awareness principle which provides a direct connection between network and application layers and helps to translate the application requirements to the necessary transmission parameters. In particular, the relation between awareness and network layers has been defined and analyzed. The connection between application and awareness is discussed on the example of Rear-End Collision Avoidance application in Chapter 4.

Lastly, the evaluation methodology described in Section 3.5 provides general steps which can aid in the evaluation of communication-based applications. It is possible to apply different models and metrics to the presented methodology when specific applications or impact need to be evaluated. The presented methodology allows a simple and straightforward approach to estimate capability of IEEE 802.11p communication to support vehicular applications. In such a way, we could provide an answer to the research questions stated in Chapter 1, namely, "how to determine the scalability of IEEE 802.11p communication to support applications and to evaluate the impact of a fail-safe application on road traffic".

Chapter 4 and Chapter 5 adopt this general evaluation methodology and evaluate the performance of two concrete applications, Rear-End Collision Avoidance and Virtual Traffic Lights applications.

4
Rear-End Collision Avoidance application

The current chapter consists of two main parts. The first part described in Section 4.1 deals with the design of a Rear-End Collision Avoidance application. Prior to presenting the resulting application logic, the performed requirements analysis and our method to integrate fail-safety features are described. In the second part of this chapter, in Section 4.2, the designed RECA application is evaluated based on the methodology presented earlier in Section 3.5. The outcome of the evaluation quantifies reliability of IEEE 802.11p communication to support Rear-End Collision Avoidance application and the resulting traffic efficiency.

Parts of the work presented in this chapter have been previously published in [AMJ+13], [AMH14], and [AMH15].

4.1 Application design

Our primary goal for application design is to design an application that is fail-safe against two major potential failure sources—unpredictable driver behavior and unreliable vehicular communication. Hence, appropriate countermeasures have to be integrated in the design. Prior to the actual design we analyze application requirements, i.e., what an application should do and what it requires to perform its functions. We perform our application's requirements analysis with a zero false positive and a zero false negative constraints. In such a way, we want to determine requirements of *an ideal application*. Naturally, satisfying both zero false positive and zero false negative errors is not possible, and a compromise or a balance between false positive and false negative is often sought instead. Although some automobile manufactures indicate allowed false positive rate to be less than 20 %, false negative rate is hardly ever mentioned [F. 11b]. Instead, safety standards, like ISO 26262 [ISO11], indicate

observable failure occurrence rate as *x allowed errors per operation hour* for different systems. Requirements analysis of an ideal application allows to balance false positive and false negative errors by quantifying the "gains and losses" and to relate to the technical feasibility of satisfying the requirements. In addition, the RECA application is envisioned to function in dynamic traffic environment, which can rapidly and unexpectedly change. The requirement analysis with zero false positive and zero false negative constraints allows to exclude unsafe situations due to unexpected traffic changes. Prior to describing the requirements analysis the relevant pre-crash conditions describing typical rear-end collision scenarios are sketched. The relevant pre-crash conditions are conditions that most frequently occur prior to a rear-end collision; the pre-crash conditions identify the situations in which an application should provide safety benefit.

Based on the pre-crash scenarios, on the requirements analysis, and on the fail-safety features, the detailed fail-safe application design is provided in the second part of this section.

4.1.1 RECA application outline

A rear-end collision is a collision between two or more vehicles that are driving in the same lane and direction, one after each other. Rear-end collisions "occur when the lead vehicle stops suddenly or unexpectedly and/or when the trailing driver follows too closely for the prevailing speeds and environmental conditions" [RPM10]. Depending on the country, between 20 % and 30 % of all vehicle collisions account for rear-end collisions, see [KMH+93], [KOUK97], [WBMD97], [T. 11], [tra12], [H. 13]. The crash statistics categorize situations that precede rear-end collisions as following:

- situations when the leading vehicle is stopped (LVS),
- situations when the leading vehicle is decelerating (LVD), and
- situations when the leading vehicle is moving with a constant speed (LVM).

According to [HPY+14] almost 60 % of all rear-end crashes involving light vehicles were preceded by a Leading Vehicle Stopped scenario. Approximately 25 % of all rear-end crashes were preceded by a Leading Vehicle Decelerating scenario, and approximately 12 % were preceded by a Leading Vehicle Moving scenario.

A Rear-End Collision Avoidance application that is envisioned to be supported by vehicular communication is heavily based on the analogous applications that are developed for sensor technologies, see [ISO13a], [ISO13b], and Section 2.2.1. An RECA application provides warnings to the driver if there is a rear-end collision danger and can apply an automatic braking if the driver does not respond to the warning. A typical scenario for the RECA application is depicted in Figure 4.1. The RECA application of the host vehicle is activated when *at least one beacon is received from a vehicle moving in the same lane and direction*, i.e., from a leading vehicle. Application warns its driver in case the approach to the LV is too fast, i.e., the rear-end collision is impending. In case the driver does not respond to the warning, the system brakes automatically.

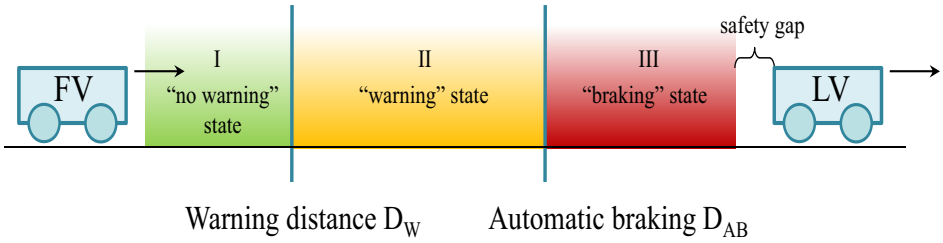


Figure 4.1: An example scenario for a Rear-End Collision Avoidance application. FV – following vehicle, LV – leading vehicle

Hence, the RECA application calculates two distances – *a warning distance* D_W and *an automatic braking distance* D_{AB}, at which application either presents a warning or performs automatic braking. The two distances also mark the change of the application's states, between "no warning", "warning", and "braking", see Figure 4.1.

4.1.2 Requirements analysis of an ideal application with zero false positives and negatives

An ideal RECA application always warns and automatically brakes exactly when it is needed, not earlier, not later, i.e., the application commits **zero false positive** (FP) and **zero false negative** (FN) errors. A false positive error (also called Type I error, false/nuisance alarm, or false positive) is an incorrect rejection of a null hypothesis. In other words, a false positive error happens when e.g., a certain condition, like a disease or a fire, has been detected present, when in fact it is not present. A failure to reject a null hypothesis when it is false is called a false negative error (also called Type II error, missed alarm, or false negative). In such a way, a false negative is a failure to detect the presence of a certain condition, when it is in fact present. In the perspective of an RECA application presenting a warning or triggering automatic braking when there is no collision threat is considered a false positive. Whereas a failure to detect the rear-end collision threat, and thus a failure to present a warning or to trigger automatic braking, constitutes a false negative.

Both types of errors are undesirable for an RECA application – false positives compromise the driver's trust in the system so that the system can be even switched off and false negatives may lead to unsafe situations. Typically, eliminating both types of errors is unrealistic, hence error minimizing strategies, depending on which error type has the most undesirable effect, are performed [Wis97]. The automotive industry defines acceptable false positive ratio for warnings, which is a ratio between the number of false positive warnings and the total number of warning events, to be less than 20 % [F. 11b]. A typical acceptable false negative ratio is usually not defined.

False positives and false negatives are influenced not only by the detection of the driving situation and the application's operation but also by the driver. A warning considered to be a false positive warning by an aggressive driver could result in a false negative warning for a more defensive driver, see Section 2.1.2. Hence, finding the

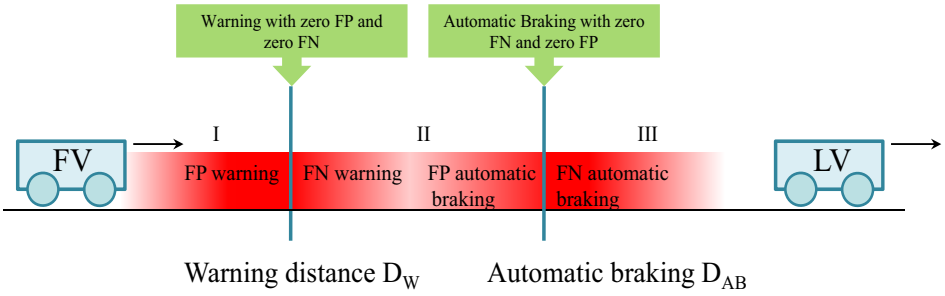


Figure 4.2: False Positive (FP) and False Negative (FN) errors

optimal balance between false positives and false negatives requires a multidisciplinary approach and is out of scope of this thesis.

Even though achieving zero false positive and zero false negative error rate is not realistic and, as it will be shown later, not necessarily essential, it is first necessary to analyze what is needed for the perfect functioning of an application in order to decide how to arrange the tradeoff. This section starts with determining the application's requirements in the context of information reception in order to achieve ideal application performance with zero false positives and zero false negatives. In the end of the section, the strict requirement of zero FP and zero FN is relaxed, and the resulting tradeoff is quantified.

Understanding zero false positive and zero false negative

In Figure 4.2 we illustrate false positive and false negative alarms of a Rear-End Collision Avoidance application. For zero false positive and zero false negative the RECA application should only warn at warning distance D_W and automatically brake only at automatic braking distance D_{AB}. If a warning or an automatic braking happens before the calculated distances, this is a false positive, if a warning or an automatic braking happens after, this is a false negative.

Consider an example depicted in Figure 4.3. In Figure 4.3(a) an RECA system of an FV receives an update from an LV and determines the two critical distances D_W and D_{AB}, as well as its operation states (I–III). If the system does not receive further updates and simply presents the warning at initially calculated warning distance D_W, it might cause a false warning, as there is an uncertainty about the position of the LV since the last update. For example, the LV might accelerate and a new warning distance D_W will now be shifted away from the FV, the FV will stay longer in the "no warning" state, see Figure 4.3(b). The related work mostly does not consider this possibility of false positive warnings. If the LV reduces its speed, the initially calculated warning distance D_W will be shifted closer to the FV and the time that the FV will spend in the "no warning" state will become shorter, see Figure 4.3(c). Not taking this into account might produce false negative errors. Similar reasoning holds for the automatic braking distance D_{AB}.

The realistic traffic situation can change dynamically and the application's state changes can happen fast. In order for an RECA application to perform with zero false positive and zero false negative, application is required to know not only how

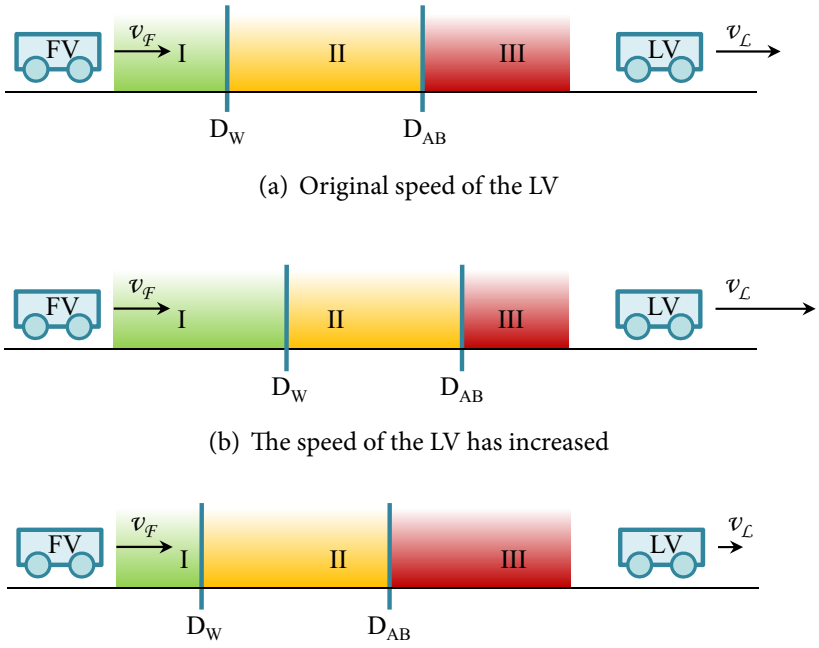


Figure 4.3: Implications of relative speed change

large are the warning and automatic braking distances are, but also how fast the FV can cross one of these distances and change application's state. Consequently, as the application in question is envisioned to be based on vehicular communication, application requirements on receiving information can be determined.

Requirements analysis

In order to determine the requirements of an ideal RECA application on information reception, the following question is answered: How far do vehicles need to communicate and when do vehicles need to receive the next update from other vehicles so that the application could successfully perform, both with zero false positives and zero false negatives? In the following, the determination of how far and how often vehicles should communicate is addressed separately.

How far should vehicles communicate? Vehicles should be able to communicate at the distance that is at least as large as the distance at which a warning should be given. According to the definition in Section 4.1.1 a warning is given to a driver if he is approaching the leading vehicle too fast, i.e., the speed of the following vehicle v_F is larger than the speed of the leading vehicle v_L. The warning distance D_W is the distance at which a driver of a following vehicle moving with the speed v_F should receive a warning, so that after his reaction time t_R, he starts applying deceleration a_F in order to come to the same speed as the leading vehicle v_L without colliding with it. A safety gap D_{gap}, i.e., the inter-vehicle distance between LV and FV which should not be crossed and the distance traveled during the system delay t_{sys} should also be taken into account.

In order to calculate a warning distance two basic kinematic formulas are used:

$$v_{final} = v_{initial} + a \cdot t \tag{4.1}$$

which determines the final speed v_{final} of a vehicle if a constant acceleration a (positive or negative) is applied during the time t to the initial speed $v_{initial}$, and:

$$d(t) = v_{initial} \cdot t + \frac{1}{2} a \cdot t^2 \tag{4.2}$$

which determines the distance d traveled during the time t if a constant acceleration a is applied to the initial speed $v_{initial}$.

Based on Equation 4.1 the time t is calculated that is necessary for the FV to apply a constant acceleration a_F to its initial speed v_F in order to achieve the same speed v_L as the LV:

$$t = \frac{v_F - v_L}{a_L - a_F} \tag{4.3}$$

where $v_F > v_L$ and $a_F < 0$. In case of LVS: $v_L, a_L = 0$, for LVM: $v_L > 0, a_L = 0$, for LVD: $v_L > 0, a_L < 0$.

According to Equation 4.2 the distance traveled by FV and LV in time t, $d_F(t)$ and $d_L(t)$ respectively, can be calculated by substituting Equation 4.3 into Equation 4.2. The inter-vehicle distance at which FV should start applying deceleration is equal to the distance difference that both vehicles travel:

$$d_F(t) - d_L(t) \tag{4.4}$$

Thus, by adding the safety gap D_{gap} and the distance traveled during reaction time t_R and the system delay t_{sys} to Equation 4.4 the warning distance is calculated as following:

$$D_W = \frac{(v_F - v_L)^2}{2(a_L - a_F)} + (v_F - v_L) \cdot (t_R + t_{sys}) + D_{gap} \tag{4.5}$$

for situations when the leading vehicle is stopped or moving with constant speed, where $v_F > v_L$, $a_L = 0$. And as following:

$$D_W = \left(\frac{v_F^2}{-2a_F} + v_F \cdot (t_R + t_{sys})\right) - \frac{v_L^2}{-2a_L} + D_{gap} \tag{4.6}$$

for situations when the leading vehicle is decelerating, where $a_L < 0$.

For example: An FV receives an update from an LV with following information: the leading vehicle's speed is $v_L = 80 km/h$ and its acceleration is $a_L = 0$ m/s², which corresponds to the LVM case. Combining it with information about own speed $v_F = 160$ km/h the FV calculates a warning distance $D_W \approx 62$ m (considered comfortable deceleration $a_F = -4$ m/s²). For simplicity and readability we assume reaction time t_R, system delay t_{sys}, and safety gap D_{gap} to be equal to zero. Thus, vehicles should communicate at the distance of at least 62 m. This is similar to the related

work approaches when a critical distance is identified as an orientation for communication range.

How often should vehicles communicate? Once the RECA application of an FV determines its current state by receiving an update at some inter-vehicle distance D, the time until a possible state change determines the allowed delay t_{update} to the next update. The state change will happen earlier if the speed difference Δv is increased, e.g., due to the speed change of the LV, of the FV, or both. The RECA application at the FV is aware of its own speed but the speed of the LV might have changed—increased, decreased, or stayed constant—from the moment the last update was received by the FV. The RECA application has to assume the worst case for the speed change of the LV—deceleration. *The goal is to determine the maximum time t_{update} during which vehicles might change their speed, in realistic boundaries, and the FV's RECA application will still be in the same state as at the reception of the last update.* As soon as the new inter-vehicle distance D^{new} equals the new warning distance D_W^{new}, the next update is needed to make sure no false positive nor false negative warnings are committed.

In other words, the time until the next update t_{update} has to be calculated, for which the following equation is true:

$$D^{new} = D_W^{new} \tag{4.7}$$

The new inter-vehicle distance D^{new} accounts for the distances that both FV and LV traveled during t_{update} from the moment the last update is received at D.

$$D^{new} := D - d_F(t_{update}) + d_L(t_{update}) \tag{4.8}$$

The new warning distance D_W^{new} is calculated as in Equation 4.5, but considers that the speed of vehicles can change. By substituting D_W^{new} and D^{new} into Equation 4.7 and solving the quadratic equation, the t_{update} can be calculated for various distances D, i.e., the time when the next update is needed for each distance D.

In the following possible states and the subsequent state changes are considered separately:

"State change I–II" represents the scenario when the last update is received in the "no warning" state. The worst case assumption that FV needs to assume, in order to prevent false negatives, as depicted in Figure 4.3(c), is that the speed of LV might decrease by a rate of a_L'.

The results for various distances D are depicted in Figure 4.4 and can be interpreted as follows: at each inter-vehicle distance, on the x-axis, the time t_{update}, on the y-axis, depicts how long the leading vehicle can change its speed, in realistic boundaries, and the FV still be in the same state as at the time of reception of the last update from the LV. After the time interval t_{update} has passed, there is a chance that the FV will change its state and Equation 4.7 becomes true; the next update is needed to avoid false warnings. For simplicity, reaction time t_R, system delay t_{sys}, and safety gap D_{gap} are assumed to be zero. Naturally, the closer the FV gets to the critical distance the more frequently it needs a next update to make sure that the current state is identified correctly. The required update frequency increases to infinity if correct classification is desired at a very close distances (e.g., millimeter precision).

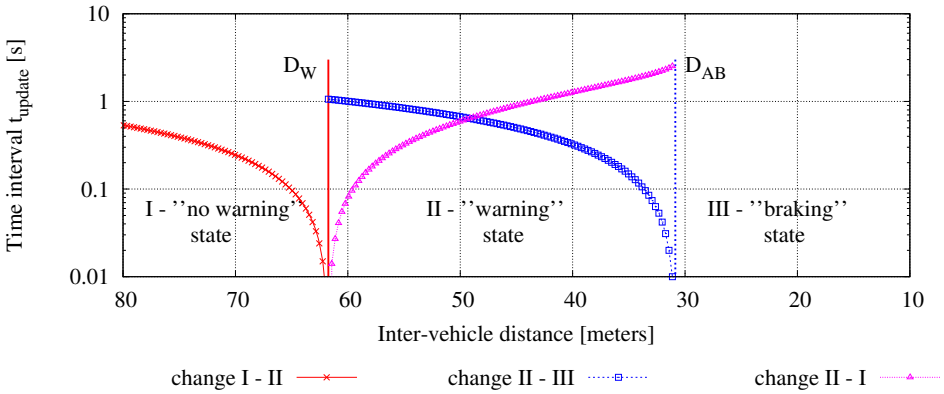


Figure 4.4: Time interval t_{update} when the FV needs to receive the next update from the LV for zero false positives and zero false negatives errors. Relative speed is $\Delta v = 22.22$ m/s (80 km/h), the worst case assumption for LV's speed change rate is $a'_L = -2$ m/s² (for the state change "I–II" and "II–III") and $a'_L = 4$ m/s² (for the state change "II–I"). For the warning distance D_W, $a_F = -4$ m/s², for the automatic braking distance D_{AB}, $a_F = -8$ m/s²; $t_R = 0$, $t_{sys} = 0$, $D_{gap} = 0$

This is rather unrealistic to fulfill and not even necessary, as such precision for the warning will not be noticed by a driver.

"State change II–III" represents the scenario when the last update is received in the "warning" state and the warning was already issued at the warning distance D_W. The RECA determines when the FV needs an automatic braking if the FV's driver does not respond to the warning by braking. The "State change II–III" is similar to "State change I–II", but the automatic distance D_{AB} is used as an orientation distance rather than the warning distance D_W. The automatic braking distance D_{AB} is calculated as in Equation 4.5, with t_R equals to zero and a_F equals to the maximum achievable deceleration of the FV, not of the FV's driver. Figure 4.4 also depicts the time interval t_{update} when a next packet has to be received for the RECA application to correctly determine when to automatically brake.

"State change II–I" represents the scenario when the last update is received in the "warning" state and the warning was already issued at D_W. The RECA determines when the warning will no longer be needed, as the driver of the FV will brake hard enough and will change from the "warning" state back to the "no warning" state. The application of the FV should again assume the worst case situation, i.e., that the LV's speed increases as the FV brakes. Corresponding time intervals t_{update} are depicted in Figure 4.4.

State III: represents the scenario when the last update is received in the "braking" state. The RECA system of the FV is already braking with the maximum achievable deceleration. From this point on DENMs are sent out. Any realistic acceleration of the LV, under this setting, will not lead to the case of the FV going back to the "warning" state. If the RECA application assumes softer deceleration values for the automatic braking state, it might happen that FV will leave the "braking" state and go back to the "warning" state. The time for the next update can then be calculated,

in analogy, as going back to the "no warning" state – "State change II–I".

The most critical areas, where updates are needed frequently are around the two critical distances of D_W and D_{AB}, when there is a possibility that the application's states will be changed. The minimum time until the next update t_{update} as well as the communication at the distance at least as large as the warning distance D_W are required by the RECA application that operates with zero false positives and zero false negatives.

In the following we discuss two example cases with the help of Figure 4.4:

Example case 1: the last update is received by the FV at the distance of $D = 70$ m. Based on the following received information about the leading vehicle: $v_L = 80$ km/h, $a_L = 0$ m/s² and information about own speed $v_F = 160$ km/h, the calculated warning distance D_W is 61.73 m. If both vehicles continue with constant speed, the FV will reach this warning distance D_W in approximately 0.37 s. However, if the LV will decelerate, by e.g., $a'_L = -2$ m/s², the new warning distance will be at $D^{new} = 64.5$ m and FV will reach it already in $t_{update} = 0.245$ s. Thus, the next update is needed already in 0.245 s after receiving the last update at $D = 70$ m. If after 0.245 s the next update is received and it is clear that LV did not change its speed, there is no need to present a warning and no false negatives will be made.

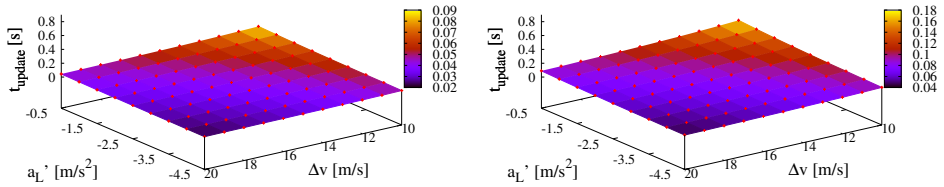
Example case 2: at the inter-vehicle distance of 50–60 m, close to the warning distance D_W, the FV needs more frequent updates as it might leave the "warning" state. The closer the FV gets to the automatic braking distance D_{AB}, e.g., inter-vehicle distance of 30–45 m, the longer the FV needs to get back to the "no warning" state. Thus, here, the RECA application should determine the next update reception according to the calculation of reaching automatic braking distance D_{AB} – "State change II–III", rather than changing back to the "no warning" state.

4.1.3 Relaxation of strict requirements

Satisfying the t_{update} requirements at each inter-vehicle distance is not realistic neither for communication nor for any sensor-based technology but also not necessary. If the warning is given five meters or less, prior to the actual calculated warning distance, the driver might not perceive this as a false positive warning, due to perception failure [WOH04]. In such a way, it not only becomes feasible to fulfill the t_{update} requirement, but also to exclude false negatives and to control the size of the area where false positives could happen.

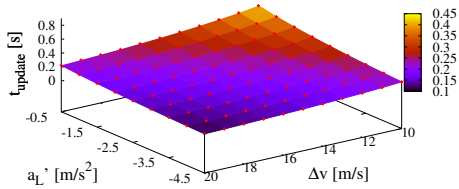
In the following, we demonstrate how relaxing the zero false positive constraint effects the t_{update} requirement based on the realistic traffic configurations, in particular, the speed differences $\Delta v = v_F - v_L$ between LV and FV, and various rates a'_L by which speed of LV could change.

A tolerance region is defined as a geographical region prior to the distance at which the state change should occur, e.g., prior to the warning distance D_W. Thus, one border of the tolerance region is adjacent to the warning distance D_W and the other border is a geographical distance at which the application maintains the required t_{update} and at which the current RECA application state is identified correctly. A warning (or automatic braking) can be triggered at the outer border or during this tolerance

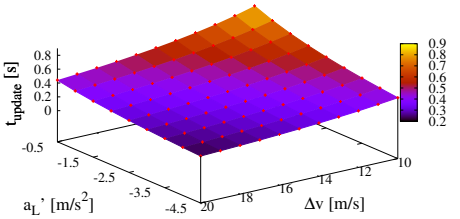


(a) Tolerance region of 1 meter

(b) Tolerance region of 2 meters



(c) Tolerance region of 5 meters



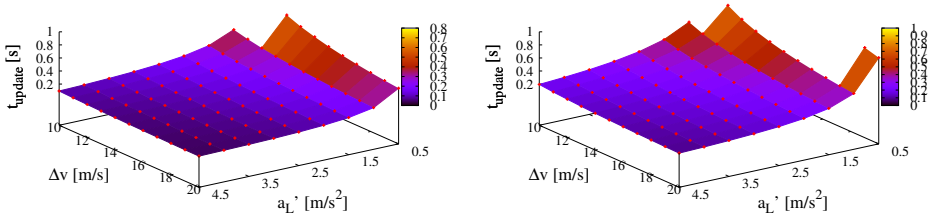
(d) Tolerance region of 10 meters

Figure 4.5: The impact of various tolerance regions on the time interval t_{update} for the "State change I–II"

region, which allows the possibility for false alarms. The false negative errors can be excluded by allowing only false positive errors, although at controllable small regions. In the following we investigate how the time interval t_{update} changes under four tolerance region sizes—1 m, 2 m, 5 m, and 10 m. Note that for large speed differences $\Delta v > 10$ m/s it is not necessary to expect successful operation of application with a small tolerance region < 1 m—no adequate driver wishes to drive close to the front vehicle with a large speed difference. Whereas for small speed differences $\Delta v < 10$ m/s, smaller tolerance regions are realistic and even desirable, as vehicles moving with similar speed tend to drive closer to each other.

In case of "*State change I–II*" the RECA application of the FV is in the "no warning" state, i.e., the situation is safe. The smaller the speed difference Δv is, the longer it takes for the FV to change to the "warning" state. Figure 4.5 presents how time interval t_{update} varies for different speed differences Δv and assumed deceleration of the LV $a_L{}'$. The shown ranges for Δv and a'_L represent the most interesting and realistic configurations. The speed difference $\Delta v < 10 m/s$ represents less problematic scenarios, as vehicles drive with similar velocities, whereas $\Delta v > 20$ m/s represents more challenging but rather seldom cases for state I. For the assumed speed change rate of LV values do not exceed $a'_L = -4.5$ m/s² as it is assumed that high deceleration values should trigger transmission of DENMs.

Figure 4.5(a) shows an unrealistic set up: firstly, large speed differences Δv with tolerance region of 1 m would result in up to 72 km/h speed difference at the inter-vehicle distance of ≈ 51 m (D_W + 1 m, where $D_{gap} = 0$, $t_R = 0$, $t_{sys} = 0$); secondly,



(a) Tolerance region of 1 meter (b) Tolerance region of 2 meters

Figure 4.6: The impact of various tolerance regions on the time interval t_{update} for the "State change II–I"

small time interval $t_{update} \approx 0.02$ s requires unrealistic reception rate of 50 Hz. The tolerance region of ≤ 1 m is relevant and realistic for speed differences $\Delta v \leq 10$ m/s.

If the last update was received at the tolerance region of 10 m, by giving the warning at the initially calculated warning distance D_W (or within the tolerance region), the system may commit a false warning, as the actual warning distance D_W could be increased by ≈ 5 m in the worst case (for speed difference $\Delta v = 20$ m/s and $a_L{}' = -4.5$ m/s²). It is out of scope to investigate whether receiving a warning 5 m earlier could be perceived as unnecessary by a driver. A false negative error can be avoided by assuming that the LV's speed has decreased.

In general, large speed difference Δv results in small time interval t_{update}. As a rule of thumb, if speed difference Δv is doubled, the time interval t_{update} is reduced by ≈ 50 %. Similar, if the speed difference Δa_L is large, the time interval t_{update} is small. If the assumed acceleration of the LV a'_L is doubled, then the time interval t_{update} is reduced by $\approx$ 20–25 %.

In case of *"State change II–I"* the FV is already in the "warning" state. Figure 4.6 shows results for tolerance regions of 1 and 2 m (tolerance region of 1 m means $D_W - 1$ m). Results for tolerance of 5 and 10 m are absent since under these configurations ($\Delta v = [10$ to 20 m/s²] and $D_{gap} = 0$, $t_R = 0$, $t_{sys} = 0$), the distance between warning distance D_W and automatic braking distance D_{AB} is sometimes ≤ 5 or 10 m. Typical realistic values for the vehicle's acceleration are $a'_L \leq 4.5$ m/s². For some speed differences Δv and small $\Delta a_L = 0.5$ m/s² the values for time interval t_{update} are missing. It is either impossible to get back to the "no warning" state under these conditions or it takes longer than 1 s and it is more probable that the FV will enter the "braking" state instead.

As a rule of thumb, if speed difference Δv is doubled, the time interval t_{update} is reduced by ≈ 60 %. If the assumed acceleration of the LV a'_L is doubled, then the time interval t_{update} is reduced by $\approx$ 50–60 %.

Smaller time intervals, $t_{update} < 0.1$ s, might also be unrealistic due to the limitations imposed by the accuracy and precision of current GNSS receivers. The position updates transmitted more often than 0.1 s will not differ much, if at all. Off-the-shelf

automotive GNSS receivers update with the frequency of 1 Hz, at most 5 Hz [XMKS04]. Some techniques, e.g., Kalman filter, in combination with additional information from in-vehicle sensors could be used to estimate position of the vehicle between the GNSS updates, although those estimations are subjected to errors.

If a reaction time of $t_R = 1$ s is introduced, then the warning distance D_W is increased (similarly, the system delay t_{sys} and safety gap D_{gap} will increase the warning distance D_W) and the required time interval t_{update} will be decreased on average by 10–30 % depending on the relative speed Δv.

4.1.4 Design of a fail-safe RECA application

We use the performed requirements analysis and the requirement to integrate fail-safe features in order to design a fail-safe Rear-End Collision Avoidance application. As described in Section 2.1.3 a fail-safe system is a system that *incorporates features for automatically counteracting the effect of an anticipated possible source of failure*. We identified two main sources of failure that can deteriorate application safety:

- *unreliable vehicular communication*
- *unpredictable driver behavior*

In the following we focus on application logic. Aspects related to fail-safety features that deal with, e.g., hardware failure or other external effects, are left out of the scope. The unreliability of vehicular communication and unpredictability of driver behavior can also be seen as an uncertainty that an RECA application can have either regarding the state of the leading vehicle, or as an uncertainty regarding own vehicle (or rather its own driver), respectively. To counteract these sources of failure, or uncertainties, the fail-safe application's logic has to assume *the worst case situation change* during the packet inter-reception time, see Figure 4.3(c), and that a driver fails to adequately react to the warning. Thus, the designed RECA application controls inter-vehicle distance (IVD) between own and leading vehicles by providing warnings to the driver or by taking over the vehicle control.

In the following the two failure sources and mechanisms counteracting their effect are addressed separately.

Addressing unpredictable driver behavior

A Rear-End Collision Avoidance application lets the driver to control vehicle as long as it is safely possible: if a rear-end collision possibility is detected the RECA application first warns its driver to take appropriate actions. If the driver does not adequately respond to the warning the vehicle control is taken over by the application and the vehicle brakes automatically. In such a way, *automatic braking provides fail-safe feature against unpredictable driver behavior*.

The RECA application still has to decide when to present a warning and when it is required to brake automatically. The decision to present a warning is based on

driver behavior characteristics, i.e., his reaction time and braking intensity. These values can only be estimated and never accurately known.

Multiple works exist that analyze driver behavior characteristics, see Section 2.1.2. The information is available in the form of either mean values for reaction time and braking intensity or presented as a distribution with a mean and a variance. If an application gives a warning considering the reaction time and braking intensity of "the average driver" (utilizing mean values), all drivers that are "worse" than the average driver require automatic braking in order to avoid rear-end collisions. If application had to account for "the worst driver" (maximum reaction time and minimum braking intensity) the majority of the drivers would consider such warning a false positive, although false negatives would be avoided. Such approach would also result in large inter-vehicle distances, and hence, inefficient traffic. A warning that accounts for "the best driver" will be more efficient with respect to traffic efficiency, but will require automatic braking for most of the drivers. In addition, the driver's characteristic values can be given only for a limited driver population and the maximum reaction time and minimum braking intensity, theoretically, are not limited. Drivers can be distracted or even sick and fail to react to the warning. The automatic braking does not need to account for driver reaction time and can use stronger deceleration up to a maximum physically possible braking intensity. We define *an automation level* that refers to the share of drivers that require automatic braking, and thus are deprived of vehicle control.

On one side, the application can leave the vehicle control to the driver and would never have to automatically brake but have to deal with highly inefficient traffic. On the other side, application can be highly efficient with respect to the traffic, and be a fully automatic system with no control left for the driver. The RECA application can control this tradeoff with varying automation level. Automatic braking allows to increase traffic efficiency, i.e., reduce inter-vehicle distance without compromising fail-safety features of a rear-end collision avoidance system.

Addressing unreliable vehicular communication

The RECA application requires to possess an up-to-date awareness picture in order to decide which action to perform. The unreliability of vehicular communication makes it impossible to guarantee a reception of the next message update. Meanwhile the traffic situation can change drastically during the packet inter-reception time (IRT)[1]. A fail-safe application needs to account for the packet IRT and be able to deal with unreliable reception of future packets.

One option to ensure fail-safety, in spite of uncertainty introduced during inter-reception time, is to assume worst case situation change that is possible from the moment the last message from the leading vehicle was received. In the case of a rear-end collision avoidance this means assuming maximum physically possible deceleration of the leading vehicle right after the last message was received. Accounting

[1]In this thesis, we assume that the transmission delay is negligible when compared to the packet inter-reception time.

for such improbable scenario might seem excessive, but as it is possible, it has to be considered to ensure fail-safety.

The worst case assumptions are as follows:

- if a leading vehicle is stopped, the worst case assumption is that it is still at stop. No assumption is made that the leading vehicle drives backwards.
- if a leading vehicle is moving with constant speed, the worst case assumption is that the leading vehicle starts deceleration with maximum physically possible deceleration.
- if a leading vehicle is decelerating, the worst case assumption is that leading vehicle's deceleration has changed to a maximum physically possible deceleration.

Resulting application logic

The RECA application's logic is based on the layout presented in [ISO13a] and [AMJ+13], see Figure 4.1, but is refined for the fail-safety objective.

The matching flowchart of the Rear-End Collision Avoidance application is shown in Figure 4.7. Based on the input of own information and information received from the LV, the application determines inter-vehicle distance and performs calculation of warning and automatic braking distances. During the packet inter-reception time the application estimates the speed and position of the LV with the worst case assumption. Depending on D_W and D_{AB} as well as on the current inter-vehicle distance the corresponding application's state is set. Such calculation happens periodically determined by the sampling time of the periodic timer but can also be triggered upon reception of a new packet from the LV.

The presented application logic, although shown for a two-vehicle case, is also fail-safe for three- and more vehicle cases. Since application assumes the worst case situation change, i.e., the maximum physically possible deceleration of the leading vehicle, it does not matter whether the leading vehicle brakes on its own or because of another vehicle in front of it.

Implications of assuming the worst case

The RECA application assumes maximum physically possible deceleration of the leading vehicle, to account for the uncertainty about the leading vehicle's speed during the packet inter-reception time. The larger the IRT is, the larger the assumed relative speed between LV and FV becomes, and thus the warning and automatic braking distances increase. Every time the FV receives a new packet, the calculation of D_W and D_{AB} is updated with the actual values for the LV's speed and acceleration. If the warning is given when the time elapsed from the last packet reception is not zero, the warning might be perceived as a nuisance alert, especially if the LV did not actually decelerate or even accelerated after the last packet reception. Figure 4.8 exemplary shows the development of the warning and braking distances calculated for the average driver (mean reaction time t_r = 1.3 s and mean braking intensity a_{FV} = −0.6 g) over

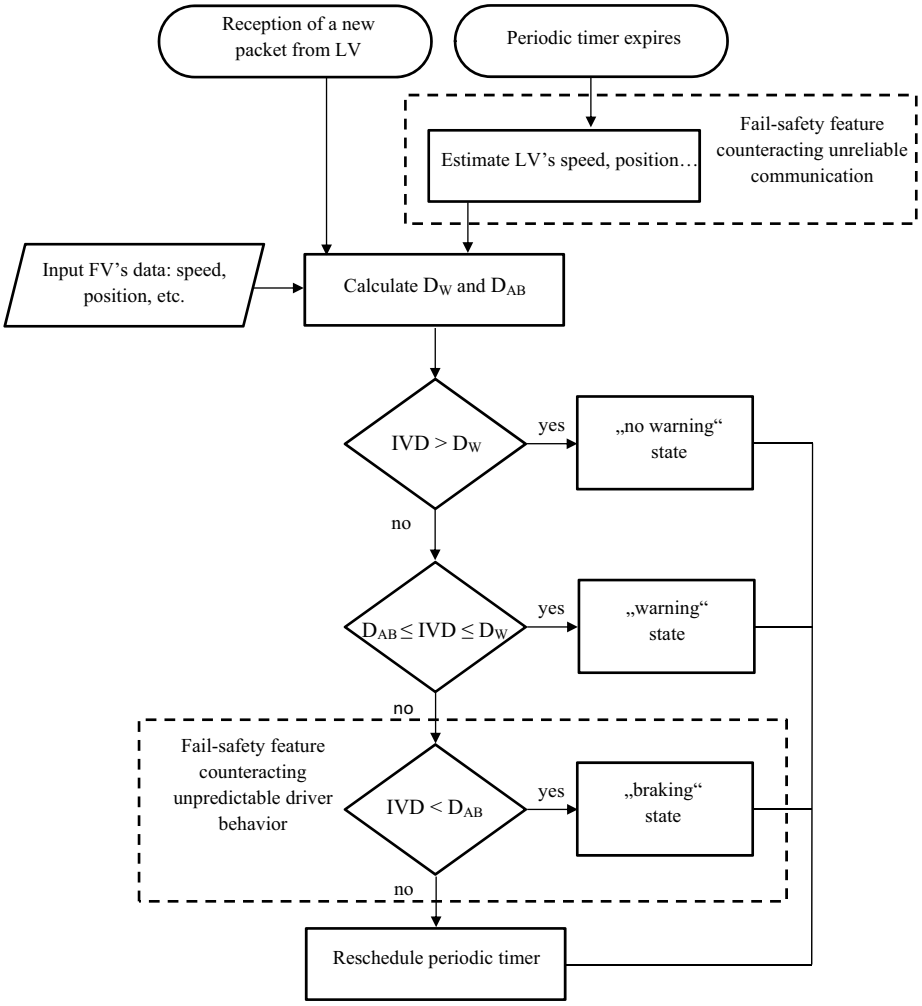
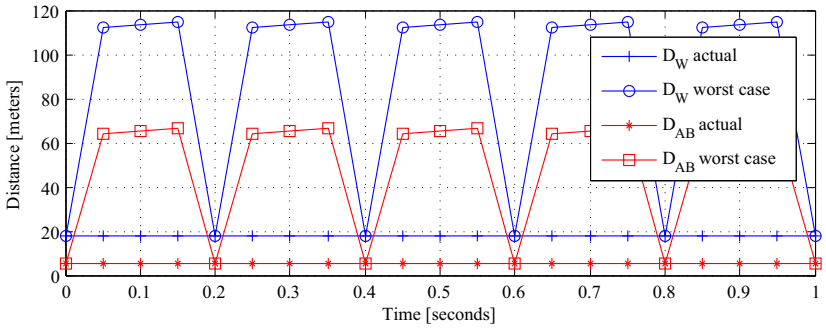
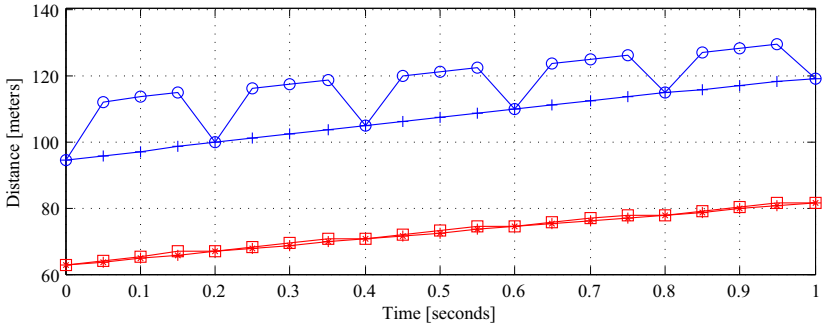


Figure 4.7: Flowchart of a fail-safe Rear-End Collision Avoidance application running on the FV. FV – following vehicle, LV – leading vehicle, IVD – inter-vehicle distance, D_W – warning distance, D_{AB} – automatic braking distance. The periodic timer refers to the internal sampling time of the application

time. The IRT is assumed to be 0.2 s, hence, packets are received at t = 0 s, 0.2 s, 0.4 s, and so on. Figure 4.8(a) depicts the scenario where the LV is moving with constant speed of v_L = 100 km/h, the FV's speed is v_F = 130 km/h. At every time sample, here every 0.05 s, the application performs warning and automatic braking distance calculations. The large increase in D_W or D_{AB} at times right after a packet is received, e.g., at t = 0.05 s, 0.25 s, 0.45 s, etc. is due to the worst case assumption. In particular, with the worst case assumption the D_W or D_{AB} are calculated as if the leading vehicle has started maximum physically possible deceleration during the time periods in which no new packets are received. Thus, the larger the IRT is the larger is the possibility for application to give a nuisance warning. Naturally, if the application assumes "the best driver" the difference between assumed worst case and actual distances is the smallest, whereas difference is at its largest when "the



(a) The leading vehicle is moving with a_{LV} = 0 g



(b) The leading vehicle is decelerating with a_{LV} = −0.6 g

Figure 4.8: Implications of the worst case assumption for the warning distance D_W and automatic braking distance D_{AB} based on information updates every IRT = 0.2 s. Deceleration of the leading vehicle a_{LV} stays the same, as reflected in calculation of actual distances. At time = 0 s the FV's speed is v_F = 130 km/h and the LV's speed is v_L = 100 km/h

worst driver" is assumed. E.g., if the FV crosses the assumed worst case warning distance at t = 0.1 s, the warning given by application is almost 100 m too early (if LV's acceleration stays the same). In this case the use of a RADAR technology would be beneficial to minimize nuisance alert rates, especially, when the FV is close to the warning or automatic braking distances. Although even high reception rates (larger than 50 Hz) do not eliminate the difference between assumed and actual distances completely, when aiming for a fail-safe operation.

Figure 4.8(b) illustrates results for a leading vehicle decelerating scenario. The original speed of LV is v_L = 100 km/h, LV's deceleration is a_{LV} = −0.6 g, and FV's speed is v_F = 130 km/h. The difference between assumed worst case and actual case (assuming acceleration stays the same) is less prominent and almost negligible for D_{AB} when compared to the LVM scenario. The small difference is due to the changing value of a_{LV} parameter in Equation 2.2. When a new packet is just received, application uses the value reported by the LV (in this case a_{LV} = −0.6 g). In between packet receptions the worst case is assumed a_{LV} = −0.8 g. Thus, the stronger the original deceleration of the LV is, the smaller the area where RECA application could make nuisance alerts.

4.1.5 Discussion

In the current section we first outlined an RECA application with its states and major functionalities. The RECA application requires calculation of two critical distances: a warning distance and an automatic braking distance, which separate "no warning", "warning", and "braking" states. We performed an analysis to determine when and where RECA application requires reception of information, if zero false positive and zero false negative constraints are aimed for. Such strict requirements were set in order to determine what an ideal RECA application requires and to identify the potential challenges for technologies in supporting of this application. We demonstrated that the most load on supporting technology occurs when application can change its states, i.e., on the borders with the warning distance and the automatic braking distance. The application requirements analysis has shown that it is not always possible, nor meaningful, to fulfill requirements with zero false positives and zero false negatives under realistic traffic and communication conditions. A "tolerance region" has been introduced to relax the strict requirements and make it feasible for communication to satisfy application needs although the possibility of false positive and false negative is admitted. We state that false negatives can be excluded by always allowing false positives, although the area where errors can happen is limited and perhaps not noticeable for human drivers.

Section 3.4 describes a solution to connect networking and application layers; it has been assumed that application can express its requirements in the form of awareness parameters. In this section, RECA application requirements have been analyzed and it has been found that RECA application requirements can be expressed in the form of the awareness parameters: the time interval t_{update} (at a certain tolerance region), witch can be matched to the time window T, the warning distance D_W, which can be matched to the awareness range R_A, and the number of packets $n = 1$. The resulting awareness probability P_A, with which awareness parameters or application requirements can be supported, can be used to represent communication reliability.

We utilized the empirical communication model of Killat et al. [KH09] to determine necessary transmission parameters that are required to satisfy the awareness parameters, and thus the application's requirements. The resulting number of communicating vehicles for which the RECA application is simultaneously supported can also be determined.

Several reports, e.g., [The05] and [Ins09], define minimum communication requirements for various applications. Typically transmission rate is stated as 10 Hz and transmission range as 100–300 m. These requirements are not justified from the application perspective and can be insufficient or excessive. In addition, the adjustment of transmission parameters by congestion control mechanisms may have a detrimental impact on the application level [SGAK14]. Depending on the traffic situation, the presented application requirements analysis provides guidelines for setting the minimum transmission rate and range. Such application awareness information can be used to control the channel congestion with minimal or no negative impact on the functioning of the RECA application. In addition, with the help of the performed requirements analysis, we identified regions where information fusion

or reliance on sensor-based technologies, due to their capabilities to update with higher frequency, can be especially beneficial.

The presented fail-safe design does not directly deteriorate traffic efficiency, as the RECA application can control the inter-vehicle distances through the automation level which impacts the amount of drivers from whom the vehicle control is taken away. In addition, by increasing the automation level, the possibility for nuisance alerts is decreased. The future work could also introduce gradual automatic braking to smoothen or reduce the effect of false positive automatic braking.

The exact evaluation of the RECA application, in particular the impact of automation level on the traffic efficiency as well as whether IEEE 802.11p communication can reliably support the application, is described in the next section.

4.2 Evaluation of a fail-safe RECA application

4.2.1 Applied methodology

The general performance evaluation methodology is presented in Section 3.5. Here the same methodology is applied to evaluate the performance of the fail-safe RECA application.

For the current performance evaluation we choose to perform an analytical study based on the awareness principle and the empirical communication model described in Section 3.4 and [KH09]. We choose not to perform a detailed simulation study because the empirical communication model is already based on the results of the detailed simulation study[2] and the questions stated for our evaluation can be answered analytically. Evaluation via simulations can benefit such applications like Adaptive Cruise Control, where a "smooth" operation over time is desired.

The empirical communication model of [KH09], also summarized in Section 3.2.2, calculates the probability of packet reception based on the number of vehicles in the neighborhood, their beacon rate, and their transmission range. Radio fading is modeled with Nakagami-m distribution with $m = 3$.

In Section 4.1.1 we outlined the most frequent pre-crash scenarios that precede rear-end collisions and in which application should be evaluated; these are Leading Vehicle Stopped, Leading Vehicle Decelerating, and Leading Vehicle Moving scenarios. A light passenger car is assumed for evaluation with a length of 5.5 m.

Adapting the general evaluation methodology, described in Section 3.5, the following steps have been carried out:

1. **Determining minimum transmission parameters:** The RECA application's requirements on receiving information, determined in Section 4.1, are expressed with a minimum distance at which vehicles should communicate, i.e., a warning distance D_W, and a minimum frequency, i.e., a time interval t_{update}, at which

[2] In addition, due to the assumed radio propagation conditions, the probabilities of packet reception, as calculated by the empirical model, are pessimistic at short distances, and hence lower than what can be expected in reality.

updates should be received. Recall, the time interval t_{update} represents *an allowed time interval* when an RECA application assumes the worst case speed change of the leading vehicle but will not change its state (e.g., from no warning to warning), hence a reception of a new packet is required not until after the time interval t_{update}. The closer the following vehicle is to the warning distance D_W or to the automatic braking distance D_{AB}, the smaller the time interval t_{update} becomes, so that *a tolerance region* is required at which the required time interval t_{update} can be used as an orientation for transmission parameters.

The awareness principle, described in Section 3.4, can be used to translate the application's requirements on receiving information into the transmission parameters. In particular, the warning distance D_W can be matched to the awareness range R_A and the time interval t_{update} (at some tolerance region) to the time window T, with number of packets $n = 1$. Similarly, as shown in Section 3.4, the empirical model of Killat et al. [KH09] can be used to determine transmission parameters that satisfy the awareness parameters with some awareness probability.

However, the following difficulties arise: there is no single warning distance that "fits" all drivers and realistic communication conditions can lead to situations when an update after the time interval t_{update} might not be received, even if analytically calculated as a probable event. Those are the exact two failure sources that has been addressed during the design, and in the following we explain how the fail-safety features affect our evaluation:

– Fail-safety feature against unpredictable driver behavior requires the application to perform automatic braking in case the driver does not adequately respond to a warning. Since the driver's reaction time and braking intensity varies among the driver population there is no single optimal value for the warning distance D_W. In order to quantify and visualize this, the "driver profiles" are generated[3]. Five hundred samples out of the reaction time distribution and five hundred samples out of the braking intensity distribution are taken, see [BKPP02] and Figure 4.9. Each sample out of one distribution is matched with five hundred samples out of the other distribution, totaling of 250000 "driver profiles".

  For each pre-crash scenario (LVS, LVD, and LVM) and for each of 250000 drivers, *the individual warning distance* is calculated according to formulas given in Section 4.1. The individual warning distance is the distance which each driver requires to react with his reaction time and his braking intensity in order to avoid a rear-end collision. The cumulative distribution function (CDF) of warning distances for all generated drivers

[3] According to [BKPP02] driver's reaction time can be modeled with a log-normal distribution with $mean = 1.3$ s and $deviation = 0.74$ s, and braking intensity with a truncated Gaussian distribution with $mean = -0.6$ g and $deviation = 0.1$ g, truncated by $max = -0.8$ g and $min - 0.3$ g, where g is the g-force and is equivalent to ≈ 9.8 m/s^2. No correlation between braking intensity and reaction time values is typically assumed.

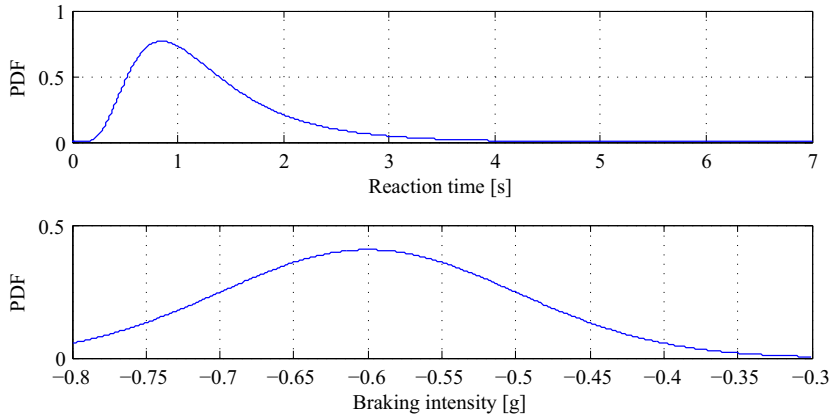


Figure 4.9: Driver behavior characteristics as reported by Brunson et al. [BKPP02]

shows the warning distances that provide enough time for a certain share of driver population to react. The fail-safety feature also determines an automation level that controls which share of drivers require automatic braking, and thus is deprived of the vehicle control. The CDF of warning distances allows to determine when to give a warning and when to automatically brake. This influences the amount of drivers that have to give up vehicle control to application or can independently react to the warning. If application gives a warning that allows enough time to react to e.g., 90 % of all drivers, then for 10 % of all drivers automatic braking is needed. The 10 % of all drivers that require automatic braking determine the automation level of 0.1. If application increases the level of automation, the warning can be given later in time when IVD is smaller.

– Fail-safety feature against unreliable communication requires the application to assume the worst case situation change for the leading vehicle. Due to probabilistic and unreliable nature of vehicular communication, the reception of packets, even if timely sent out, cannot be always guaranteed. As a consequence, various packet inter-reception times should be expected. A fail-safe application assumes the worst case change of the warning and automatic braking distances during packet inter-reception times. The maximum warning distance D_W and automatic braking distance D_{AB} that result due to various IRTs are calculated during evaluation, instead of their evolution over time, as in Figure 4.8. The considered inter-reception time values are 0.1 s, 0.2 s, 1 s, 2 s, as well as IRT = 0 s, representing the best case. In a way, an IRT experienced by the RECA application can be matched to the time interval t_{update} at certain tolerance region that is defined in requirement analysis, where the tolerance region is resulted from the size of the IRT.

In such a way, the CDF of 250000 warning distances (maximum warning distances for different IRT) or rather the automation level, determines the

minimum distance at which communication should take place. This distance is matched to the awareness range R_A. In addition, we use a varying time window T during which at least $n = 1$ packets should be received. We choose to use a time window T during which awareness is observed, rather than selecting a tolerance region and the corresponding time interval t_{update}, as those are dependent on the individual driver characteristics. With the help of empirical communication model [KH09] we calculate required transmission parameters that satisfy awareness parameters.

2. **Determining the number of communicating vehicles:** The number of communicating vehicles is estimated in the form of *vehicle density*, measured with vehicles per kilometer per lane (veh/km/lane). In particular, the warning distance relates to the inter-vehicle distance that is possible between two vehicles at a given relative speed before the RECA application warns or automatically brakes. This inter-vehicle distance can be used to calculate maximum vehicle density. For example, if a warning distance is equal to 194.5 m, accounting the average vehicle length of 5.5 m, the possible vehicle density for a fail-safe rear-end collision avoidance application is 5 veh/km/lane. If IEEE 802.11p communication is found capable of supporting this initially estimated vehicle density, the density can be scaled up by increasing the number of lanes, or scaled down otherwise. In addition, the chosen automation level influences the distance where a warning is given, and thus impacts the vehicle density. This provides a degree of freedom to application: the vehicle density can be increased if for more drivers the vehicle control is taken away without compromising traffic safety. The automatic braking distance marks the smallest inter-vehicle distance for a given relative speed and can be used to calculate the maximum vehicle density that can take place with a fail-safe RECA application, irrespective of whether IEEE 802.11p communication can support it or not.

3. **Determining the reliability of IEEE 802.11p communication to support an application:** The following definition of communication reliability is accepted for the rear-end collision avoidance application:

   Definition 4.1. ***Communication reliability, also denoted as awareness probability*** P_A***,*** *is the probability with which the application's requirements, expressed in awareness parameters, i.e., reception of at least n packets in a time window T at an awareness range* R_A*, are supported.*

   We use the empirical communication model of [KH09] to obtain probability of packet reception values. With the obtained probability of packet reception we calculate the awareness probability P_A or communication reliability, with which the application's requirements can be satisfied for the vehicle density estimated in the previous step.

4. **Determining the scalability of IEEE 802.11p communication:** If IEEE 802.11p communication can achieve "a sufficient level" of awareness probability or communication reliability, the number of communicating vehicles can be gradually

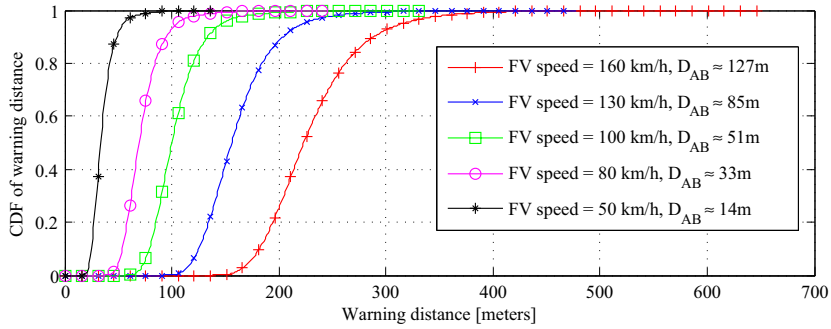
LoS	Description	Max. vehicle density [veh/km/ln]	Average speed [km/h]
A	complete free flow	7	100.0
B	free flow	11	100.0
C	marked influence of other vehicles presence	16	98.4
D	traffic congestion	22	91.5
E	near capacity, unstable level	25	88.0

Table 4.1: The multi-lane highway Level of Services for a free-flow speed of 100km/h, according to Exhibit 21-2 in [UB10]

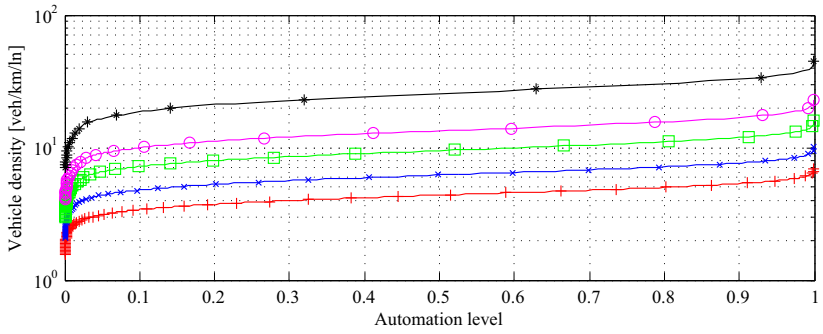
increased – scaling up to a multi-lane highway with 2-, 4-, 6-, and 8-lanes, and repeating the previous step. If this is not the case, the vehicle density or the automation level can be decreased and the previous step repeated.

5. **Determining the impact on the road traffic:** The resulting vehicle density is compared with Level of Services (LoS) provided by the Highway Capacity Manual [UB10]. The Highway Capacity Manual [UB10] provides characterization of the performance of portions of the transportation system, e.g., multi-lane highways, with the help of Level of Services. Table 4.1 summarizes the LoS classes for a multi-lane highway with their corresponding maximum vehicle density (and average vehicle speed) for a free-flow speed (speed at low vehicle density) of 100 km/h. Lower free-flow speeds of 70–90 km/h result in similar traffic density with slightly higher vehicle density for LoS E (up to 28 veh/km/ln). These LoS can be used not only to estimate the traffic efficiency impact of an application, but can also be used by applications to serve as an orientation for the maximum desired vehicle density. Interestingly, it might be counter-productive to increase automation level in order to accommodate higher vehicle densities, not only due to increasing congestion in the communication channel but also due to deterioration of the LoS.

   In addition, resulting traffic efficiency is compared against road traffic regulations in Germany. The German road traffic regulations indicate that the minimum inter-vehicle distance should be large enough to ensure enough space to react for a sudden braking of the LV. The penalty for following with a small IVD starts when speed is larger than 80 km/h and IVD equals to 1/4 of the speedometer value in meters (e.g., for speed of 80 km/h, at IVD less than 20 m). If the automatic braking distance is much smaller than the distance subjected to penalties, the drivers might have negative driving experience, in addition to penalties. Note, that with the advancement of automated driving, it becomes a reality to ensure fail-safe operation at IVD much smaller than what is required from human drivers.



(a) The CDF of warning distances for 250000 drivers



(b) Vehicle density for various automation levels

Figure 4.10: Leading Vehicle Stopped scenario

4.2.2 Results

Leading Vehicle Stopped scenario

If the LV is stopped and the worst case assumption between packet reception is that the LV is still at stop, then the duration of packet inter-reception time does not play a role in calculation of warning or automatic braking distances. The fail-safe application assumes that the LV is stopped as long as information stating otherwise is not received.

Figure 4.10(a) shows the CDF of all warning distances for the generated driver profiles assuming different speeds for the FV. Due to the heavy-tail distribution of driver's reaction time, the distribution of warning distances has also a heavy tail. As can be seen, most of the drivers share a similar warning distance, and the warning distance decreases with decreasing speed of the FV. If the FV's speed is 80 km/h and the application adjusts its warning distance to accommodate 99 % of all the drivers, then application warns at IVD of 143 m. In this case, 99 % of all drivers will manage to come to a stop on their own without colliding with the LV but 1 % of the drivers requires an automatic braking in order to avoid the collision. If application supports automation level of 0.99 (enough time to react for 1 % of all the drivers) then warning can be given at IVD of 47 m. The increase of automation level allows higher traffic densities but can make communication more challenging. Even though more vehicles

Probability of awareness (P_A)	Time window T				
	0.2s	0.4s	0.6s	0.8s	1.0s
99.9999%	0 m	6 m	18 m	26 m	31 m
99.999%	0 m	12 m	23 m	30 m	35 m
99.99%	1 m	18 m	28 m	35 m	39 m
99.9%	6 m	26 m	35 m	41 m	45 m

Table 4.2: Achieved awareness ranges in the LVS scenario with 8 lanes, the FV speed is 50 km/h. In comparison, the warning distance is 16 m and the automatic braking distance is 14 m

are communicating, their communication range requirement is decreased since the corresponding warning distance is smaller, i.e., more vehicles should communicate at smaller distances. If the FV's speed is 80 km/h, the vehicle density increases from 6.73 veh/km/lane to 19.05 veh/km/lane when automation level is increased from 0.01 to 0.99, as seen in Figure 4.10(b). Such increase in automation level also change the LoS of the multi-lane highway from level A to level D, i.e., free-flow conditions are changed to congestion, cf. Table 4.1. Even if high vehicle densities are feasible from IEEE 802.11p communication perspective, it might not be desirable from highway LoS or driver experience's perspectives. However, this analysis gives an insight on the maximum vehicle density, with IVDs much smaller than what human drivers can safely handle, that is possible by fail-safe RECA application, irrespective of whether IEEE 802.11p communication can support that or not. The corresponding automatic braking distances are stated in the legend of Figure 4.10 for different FV's speed.

With respect to the question whether IEEE 802.11p is able to fulfill the RECA application requirements, it is best to take a look at the most demanding scenario: FV's speed is 50 km/h and automation level of 1. In this scenario, a warning distance of 16 m is necessary, which translates to a vehicle density of 46.5 veh/km/lane; in addition, the vehicle density is scaled-up to an 8-lane highway. Table 4.2 indicates maximum achievable awareness ranges with various awareness probabilities for different values of time window[4]. As can be seen, the achievable awareness ranges is lowest for a high reliability requirement and a small time window T. When comparing these awareness ranges with the warning distance of 16 m and the automatic braking distance of 14 m (cf. Figure 4.10(a)) it becomes clear that a very high awareness probability, and hence a high level of reliability, is only achievable for a time window of 0.6 s and greater. In non-technical terms, this means that an LVS situation can be detected with 99.9999 % probability and allow timely reaction of the application, but the information would be 0.6 s outdated, which gives space to false positive errors. In addition, if the leading vehicle is stopped it is more probable that it is still at stop after 0.6 s. If it is desired to base application decisions on the information that is less outdated, the probability with which this could be possible has to be decreased. It should be noted that the

[4]The awareness principle is used to calculate the probability of awareness and awareness ranges achieved with lowest channel load and varying transmission parameters, calculated with the help of the empirical model of Killat et al. [KH09].

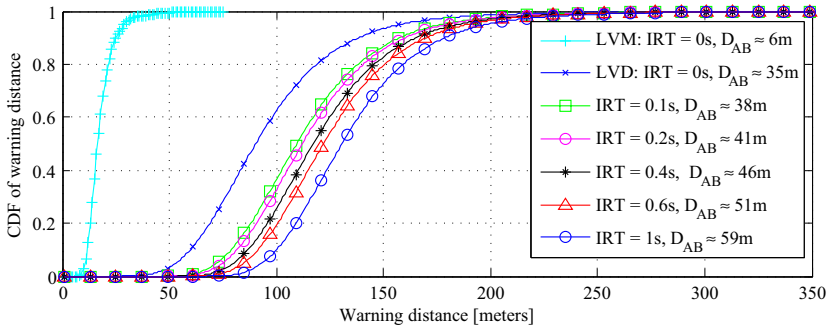
above calculations are based on the worst case assumptions: first, an FV's speed of 50 km/h is not realistic if all other vehicles share an IVD of 16 m; second, the used empirical communication model assumes strong fading at close distances, which is likely to be more optimistic in the reality, cf. Section 3.2.2.

When IRT > 0 s the nuisance alerts are possible, i.e., the warning is given although IVD is larger than the warning distance D_W. In order to calculate the size of the region where false positives can happen, we look at the following example: If "an average driver" is warned of LVS, but the LV started moving from the moment the last packet was received by the FV (LV accelerates only during IRT), the warning is ≈ 3 m too early, when v_F = 100 km/h, the IRT = 0.2 s, and the average acceleration of a starting vehicle is = 0.23 g [RPM10]. Considering that the overall warning distance is > 100 m the "early warning" might not be perceived as a nuisance alert by the driver; the 3 m error is also within the tolerance limits identified in [ISO13a].

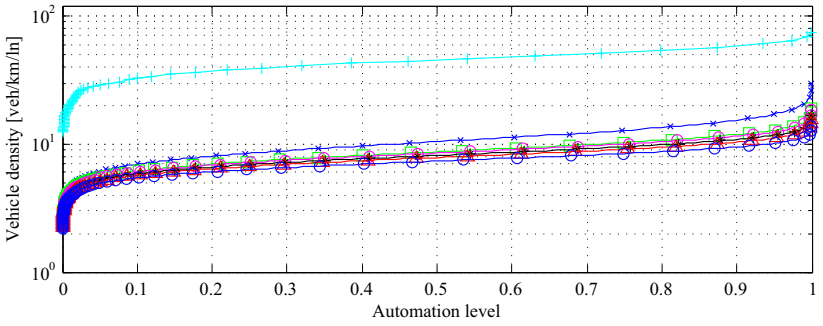
Leading Vehicle Moving and Leading Vehicle Decelerating scenarios

The evaluation results for LVM and LVD scenarios are presented together as their worst case assumption when IRT > 0 is the same. Figure 4.11 shows the CDF of the warning distance for the typical freeway scenario where vehicles are moving with similar speeds – the FV's speed is 130 km/h and the LV is moving with original speed of 100 km/h. The actual/original deceleration value of LV is a_L = 0 in the LVM scenario and a_L = −0.6 g in the LVD scenario, shown when IRT = 0. If IRT > 0, the RECA application, for LVM and LVD, assumes the worst case deceleration of the LV, i.e., a_L = −0.8 g. As can be seen in Figure 4.11(a) the IRT has a strong impact on the warning and automatic braking distance, and the gap to the curve for IRT = 0 is most significant in the LVM case, just as seen in Figure 4.8. If average driver behavior characteristics are considered, the difference between IRT = 0 s and IRT = 0.2 s, for the LVM scenario, is the same as shown in Figure 4.8(a) and the difference between IRT = 0 s and IRT = 0.2 s, for the LVD scenario, is the same as shown in Figure 4.8(b). As already been stated, the larger IRT values result in the increase of the warning distance in order not to "miss" any dangerous situation change and in the increase of the probability for a nuisance alert. The latter is especially prominent if the LV did not decelerate and FV is close to the warning distance. Higher automation levels allow to decrease the difference between the actual warning distance and what is assumed by the application. If the packet IRT is = 0.2 s and an automation level of zero is targeted, then the warning distance ≈ 236 m, which translates to a vehicle density of 4.14 veh/km/lane. By increasing the automation level to 1, i.e., such that all drivers are too slow to react in time, the warning distance reduces to 49 m, which translates to a vehicle density of 18.34 veh/km/lane, see Figure 4.11(b). The derived vehicle densities result in a highway LoS between LoS A and LoS D if packet IRT > 0 s. The vehicle density for IRT = 0 s, although representing unstable operation at breakdown according to [UB10] shows the idealistic highway capacity, when technology is not a limiting factor.

The corresponding automatic braking distances for various IRTs are given in the



(a) The CDF of warning distances for 250000 drivers



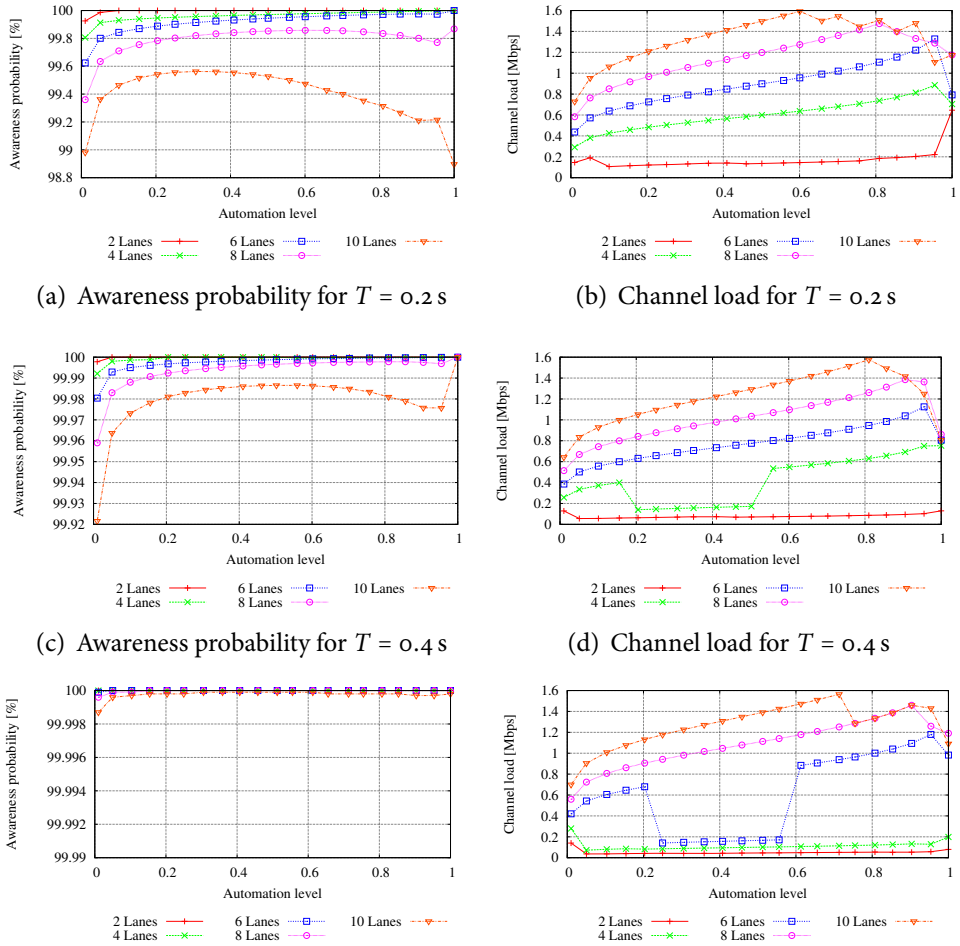
(b) Vehicle density for various automation levels

Figure 4.11: Leading Vehicle Moving and Leading Vehicle Decelerating scenarios for v_F = 130 km/h and v_L = 100 km/h

legend of Figure 4.11. For realistic cases of IRT > 0 s the automatic braking is started at IVD that is larger than the IVD subjected to penalties in Germany.

Whether vehicular communication is able to provide a reliable service in this setup is answered by Figure 4.12. We use varying time windows T = 0.2 s, T = 0.4 s, and T = 0.6 s, during which we observe whether IEEE 802.11p communication can reliably support application's requirements. In a similar fashion as in Figure 4.4, by assuming the worst case, we can also determine the minimum tolerance region that has to be accepted for given communication conditions and traffic situations. For previously considered scenario with v_F = 130 km/h and v_L = 100 km/h and the assumption of average driver characteristics, the time window T = 0.2 s corresponds to a tolerance region of ≈ 4.5 m; time window T = 0.4 s corresponds to a tolerance region of ≈ 9.4 m; and time window T = 0.6 s corresponds to a tolerance region of ≈ 14.7 m.

Figure 4.12(a) and Figure 4.12(b) plot the maximum achievable awareness probability and the resulting channel load (in Mbps) for the case when the time window T = 0.2 s, with respect to the automation level and the number of lanes considered. Each shown awareness probability value stems from the optimal combination, the one resulting with smallest channel load. As transmission range and rate are varied in discrete steps, in order to find a combination that yields the lowest channel load, the resulting



(a) Awareness probability for T = 0.2 s

(b) Channel load for T = 0.2 s

(c) Awareness probability for T = 0.4 s

(d) Channel load for T = 0.4 s

(e) Awareness probability for T = 0.6 s

(f) Channel load for T = 0.6 s

Figure 4.12: Maximum achievable awareness probability and the channel load for Leading Vehicle Moving and Leading Vehicle Decelerating scenarios

curves are not smooth. As can be seen, the awareness probability increases with an increasing automation level as long as 6-lanes or less are considered. The awareness probability reaches close to 100 % in the 2-lanes setup if the automation level is above 0.1. In the 8-lane and 10-lane scenario, the vehicular communication system is not able to provide a highly reliable service, and the increase of the automation level helps only up to a certain point. A look at Figure 4.12(b) further reveals that the offered load never exceeds 1.6 Mbps, which is less than 27 % of the available 6 Mbps. Please note that the load curves vary with respect to the automation level and that no clear tendency is visible. This is a result of the fact that different transmission rate and range combinations yield the optimal awareness probability. When considering a time window T of 0.4 s or 0.6 s, the vehicular communication system is less challenged, as can be seen in Figure 4.12(c), 4.12(d), 4.12(e), and 4.12(f). Consequently, higher awareness probability can be achieved. This is to be expected as an increase of the

time window T increases the time interval during which at least one packet has to be received successfully. At the same time, the load on the wireless channel is kept at a similar level as before.

4.2.3 Discussion

The presented evaluation estimates the performance of Rear-End Collision Avoidance application under the three most frequent pre-crash scenarios. The awareness principle together with the empirical communication model is used to determine the maximum number of communicating vehicles that can be reliably supported by IEEE 802.11p communication. In such a way, communication reliability is defined as the probability to satisfy the application's requirements. By adjusting the automation level the RECA application can control not only the share of the driver's population that has to give up the vehicle control but also the maximum vehicle density. We quantify the degree of communication reliability, a geographical area where nuisance alerts can happen, and the resulting vehicle density.

There are multiple ways to minimize the area where nuisance alerts can happen. For example, it is safe to assume that the leading vehicle will transmit the Decentralized Environmental Notification Message in case of emergency braking [HPY+14]. Thus, the application's logic can assume less intense braking as a worst case traffic situation change assumption. Adaptive transmission rates, especially when the inter-vehicle distance is close to the warning distance D_W or automatic braking distance D_{AB}, could also help reduce the nuisance alert probability. A hybrid approach of IEEE 802.11p communication together with sensor-based technologies, e.g., a RADAR, is also a promising solution to reduce nuisance alerts as the weakness of one technology can be supported by the strength of the other [HPY+14]. In addition, when the application experiences large IRTs, partial automatic braking with a smaller deceleration value can be performed. The closer vehicles get to each other, the higher the probability of successful communication. Hence, it might happen that the automatic braking maneuver might not need to be finalized. These techniques may help reduce probability of nuisance alerts and maintain the warning distance accuracy within ±15 % foreseen in the ISO 15623 standard [ISO13a].

The resulting vehicle density is comparable to that seen on today's roads, which indicates that a good level of traffic efficiency can be achieved with the fail-safe RECA application. The presented evaluation, although it represents only initial evaluation results, provides a good insight on the performance and impact of the RECA application.

4.3 Conclusion

In this chapter a Rear-End Collision Avoidance application has been designed and evaluated. The application design is based on the relaxed application requirements of zero false positives and zero false negatives, and the integrated fail-safety features. The RECA design determines the automation level that controls the amount of drivers that should give up vehicle control to avoid rear-end collisions and consequently

controls the minimum inter-vehicle distance. We used this inter-vehicle distance to determine resulting traffic efficiency, in particular, the maximum vehicle density. Following the general evaluation methodology presented in Section 3.5, we evaluated the designed RECA application performance. We showed that when higher reliability of IEEE 802.11p communication is required, i.e., the probability to satisfy applications requirements, the tolerance region where nuisance alerts can happen must be increased. The automation level has a large impact on traffic efficiency but rather limited impact on resulting communication reliability. The resulting vehicle densities are comparable to densities seen on today's highways. This indicates rather good level of traffic efficiency that can be achieved for a fail-safe RECA application.

The designed application logic, although shown on a two-vehicle case example, is valid for three- and more vehicle cases. This is because the worst case assumption makes it irrelevant whether the leading vehicle changes its speed based on another vehicle or on its own. In addition, presented application requires reception of at least one beacon message from the leading vehicle to become active. In the presence of severe communication failures, when no packets are being received, reliance on sensor-based technologies is required.

We performed evaluation of the RECA application in an analytical way with the help of the empirical communication model of Killat et al. [KH09]. This method provided us with sufficient level of realism, because the empirical communication model itself is based on a simulation study and covers a wide range of parameters. Naturally, more evaluations should be carried out before the first communication-based prototypes will be seen on roads. Future work may investigate how traffic safety and efficiency, or the nuisance alert probability, can be improved if a fusion of information provided by sensor and communication technologies is implemented. Further fail-safety features, to prevent rare failures, can be investigated by the future work, e.g., a fail-safety feature to prevent random hardware component failure.

5
Virtual Traffic Light Application

In the current chapter we discuss the design and evaluation of the Virtual Traffic Lights application. Section 5.1 addresses the design of the VTL application together with the application requirements analysis and a formal verification. In order to perform a requirements analysis for the communication-based VTL protocol, we build upon requirements of a conventional traffic light. In addition, we integrate appropriate fail-safe measures into the design to counteract unreliability introduced by communication and discuss a possible fail-safe feature to counteract the effect of unpredictability of human drivers. The resulting application design is formally verified to ensure safety of the application with a model checking approach and a tool called SPIN. This verification procedure allows us to improve the protocol design based on failed verification attempts. In the second part of this chapter we present the evaluation of the designed VTL protocol under realistic conditions via simulation. We follow the general methodology provided in Section 3.5 in order to determine the scalability of IEEE 802.11p communication to reliably support the designed VTL protocol. The VTL impact on the traffic is compared to the impact achieved by a simple all-way stop and conventional traffic lights. We conclude the chapter by elaborating on implications of our results, limitations, and future work directions.

Parts of the work presented in this chapter have been previously published in [NAH13], [NAT+12], [Neu13], and under http://dsn.tm.kit.edu/english/vtl.php

5.1 Application design

This section addresses the design of a fail-safe communication-based VTL application. Just as the RECA application, the VTL application faces the same possible failure sources, namely unreliable communication and unpredictable driver behavior. The mechanisms to counteract the effects of these failures have to be integrated in the

design of VTL application in order to achieve fail-safety. Prior to the actual design of the protocol, application requirements are analyzed. The requirements of conventional traffic lights are taken as a basis for requirements of a VTL application. In addition, formal verification is performed to ensure safety of the designed VTL protocol in the presence of distributed coordination and unreliability of communication. In such a way, requirements analysis, fail-safe mechanisms, and verification contribute to the end design of the VTL application.

5.1.1 VTL application outline

In 2010, Ferreira et al. proposed a Virtual Traffic Lights approach to manage intersections without traffic lights via communication [FFC+10]. The following description of the VTL protocol is provided: when vehicles approach an intersection, their VTL application checks whether there is already a virtual traffic light running that must be obeyed, or whether there is a need to create one. The need to create a new virtual traffic light instance is based on the received beacons from vehicles whose paths cross on intersection's shared center area. If it is necessary to create a new virtual traffic light instance, then all approaching vehicles must agree on electing one of them to become *a VTL Leader*, which has a role of a temporary infrastructure and broadcasts traffic light signal for the whole intersection. The *VTL Leader election* requires all approaching vehicles to form *clusters* containing all vehicles on the same road segment. Each cluster determines its closest vehicle to the intersection, called *a Cluster Leader*. All Cluster Leaders from approaching road segments agree on electing the VTL Leader among each other. The VTL Leader should have two characteristics: 1) should broadcast a red light to its own road, and thus be stopped at the intersection, and 2) be the closest vehicle to the intersection center on its own road segment. As long as there is a VTL Leader, all other vehicles are passive and simply obey the broadcast signal. During its lifetime, the VTL Leader only broadcasts the current signal without changing of the current phase. When the current phase is over, the VTL Leader and all vehicles on its road get the green light and can cross the intersection. If there are other vehicles stopped at the red light, the VTL Leader can *handover* its leadership, if there are no other stopped vehicles, the VTL Leader election can be repeated whenever needed later in time. An example scenario for the VTL application is also depicted in Figure 5.1.

The lack of further design details as well as aspects outlined in Section 2.3.2 prohibit direct usage of the protocol presented by Ferreira et al. in [FFC+10] for the research questions stated in Chapter 1. Nevertheless, we design a VTL application from scratch based on the approach of Ferreira et al., including preserved terminology. In this thesis, the protocol presented by Ferreira et al. is referred to as "Ferreira VTL".

The Ferreira VTL has the same assumptions as made in this thesis, namely, assumptions on 100 % IEEE 802.11p equipment ratio, possession of digital maps, and lane-level accuracy of GNSS signal. Additional assumptions of [FFC+10]—"security, reliability, and latency of the wireless communication protocol are assumed to be adequate for the requirements of the VTL protocol"—are only partially made in this thesis (see below).

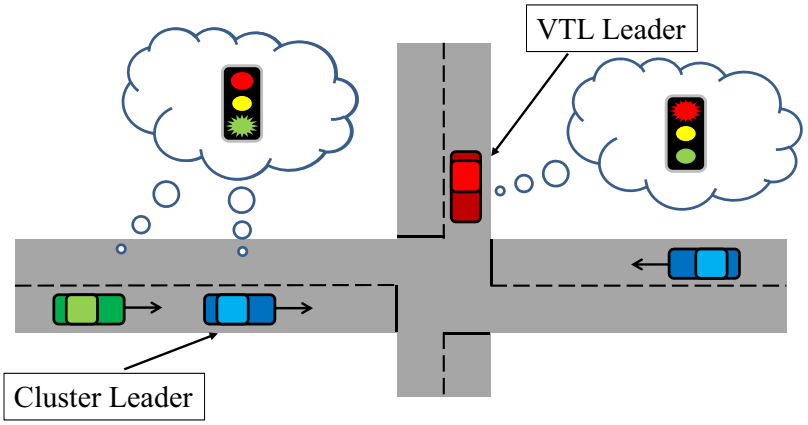


Figure 5.1: An example scenario for a Virtual Traffic Lights application

5.1.2 Application requirements analysis

Requirements of a VTL application are strongly based on the requirements of conventional traffic lights. Conventional traffic light alternate intersection crossing order by switching its lights between green, red, and yellow. Green light allows the traffic to cross, whereas red light prohibits intersection crossing. Yellow light signalizes a light switch between red and green, and allows crossing with caution if vehicles are unable to safely stop. Traffic volume, pedestrians, and public transportation can dynamically effect the duration of red and green phases. As discussed in [RPM10], two safety constraints are posed on conventional traffic lights, which if provided, ensure overall safety of conventional traffic lights as long as drivers are compliant and not distracted. These constraints are as following:

- Vehicles with a red light displayed must be able to safely stop in front of the intersection;
- Conflicting vehicles must not enter the intersection at the same time. Two vehicles are conflicting if their paths may try to occupy the same physical space at the same time.

The first constraint is fulfilled by using a yellow light phase; the duration of the yellow light phase allows for safe braking in front of an intersection or a safe crossing. Thus, a *safety distance* D_{safe} is specified, at which the information on a traffic light has to be available as it marks the latest point when the driver needs to decide whether he should decelerate and stop or he can safely cross the intersection. The safety distance is dependent on vehicle's speed v and driver's comfortable deceleration a_{comf}. The traffic light systems, conventional or virtual, have to provide the current traffic light phase to the driver at latest at this safety distance.

The safety distance is based on kinematic laws and can be calculated as following:

$$D_{safe} = \frac{v^2}{-2a_{comf}} \tag{5.1}$$

The time that vehicle needs from crossing the safety distance D_{safe} until stopping at the beginning of the intersection can be used for determining the minimum length of the yellow phase duration. If the traffic light phase is switched to yellow before an approaching driver can cross the safety distance D_{safe}, the driver can safely decelerate and stop at the intersection. If the light is switched to yellow after the driver has crossed the safety distance, he can still safely cross the intersection. Traffic authorities suggest a yellow phase duration of 3 s for the roads with the speed limit of 50 km/h (4 s and 5 s for 60 km/h and 70 km/h, respectively) [Thu99].

The second constraint is fulfilled if a green light is only assigned to the roads with no conflicting vehicles, i.e., only vehicles that do not try to simultaneously occupy the same physical space are allowed to proceed to intersection crossing.

Thus, the two main requirements of *virtual traffic lights* can be formulated as follows:

1. Drivers get notified on the current traffic light phase at latest at safety distance D_{safe}
2. Conflicting vehicles must not get the right of way, i.e., green light, at the same time. In other words, traffic lights information has to be *consistent*.

Requirements on receiving information

Based on application's requirements, requirements on received information can be drawn, which later can be used for estimating required communication parameters. Due to unreliability of vehicular communication, the requirements on received information are not strict, since cannot be guaranteed, and serve only as an orientation for choosing the transmission parameters. The fail-safety features, integrated into the design (explained in Section 5.1.3), ensure application still remains safe even if information reception is not as required.

The safety distance D_{safe} marks the closest point to intersection where drivers should be able to decide whether they can cross the intersection or have to decelerate; thus, the safety distance indicates the minimum distance to intersection at which information (from vehicles on other approaching road segments, as discussed in Section 5.1.3) should be received. For example, if vehicles are approaching the intersection with constant speed of $v = 50$ km/h, and drivers' comfortable deceleration rate is $a_{comf} = -2\,\mathrm{m/s^2}$, the reception of required information should take place no later than at $D_{safe} \approx 49$ m. Similarly as in the design of the rear-end collision application, the comfortable deceleration varies for different drivers, cf. Section 2.1.2.

There is no strict requirements for the frequency with which VTL information should be updated, as the VTL information is not expected to change drastically. If information on VTL signal phase is received prior to reaching the safety distance, this information can be trusted throughout the whole duration of the signal phase, see details on the design.

5.1.3 Design of a fail-safe VTL application

A fail-safe design of the VTL application also requires integration of fail-safe features, which are addressed separately, below.

Addressing unpredictable driver behavior

The challenge of choosing appropriate comfortable deceleration value a_{comf} for each driver is the same as in the design of a Rear-End Collision Avoidance application, namely, drivers possess different comfortable deceleration parameters and some drivers might not decelerate at all, see Section 4.1.4. Here too, the VTL application can calculate an automatic braking distance at which application starts deceleration if the driver fails to do so. The automatic braking distance can be calculated just as in Equation 5.1, with a_{comf} changed to, e.g., the maximum physically possible deceleration of a vehicle. As the procedure stays the same, for the design and evaluation of the VTL application it is assumed that the driver's comfortable deceleration value is known, namely $a_{comf} = -2\,\mathrm{m/s^2}$, the driver is obedient and non-distracted.

Addressing unreliable vehicular communication

Two types of communication conditions are possible for VTL application – Line-Of-Sight (LOS) communication and Non-Line-Of-Sight (NLOS) communication conditions. The cluster formation and Cluster Leader election happen primarily under LOS communication conditions. Whereas *VTL Leader election* and *handover* mostly happen under NLOS communication conditions. In the current design we focus on the more challenging communication condition, namely, the NLOS. The LOS communication conditions are assumed to be perfect: a vehicle is assumed to be able to "sense" whether another vehicle is driving in front or whether it is the first vehicle on a particular road segment to reach the intersection (i.e., if it is a Cluster Leader). This can be achieved with, e.g., fusion of information from communication and in-vehicle sensors.

Below the fail-safety features, that are included in the design of the VTL application to counteract the unreliability of vehicular communication for the VTL Leader election and the handover processes under NLOS conditions, are presented separately.

Fail-safety features for VTL Leader election:

- Upon approaching of an intersection the Cluster Leaders agree on electing one of them to be a VTL leader. Unreliable communication can cause packets from one or more Cluster Leaders to be lost and several Cluster Leaders may decide to become VTL Leaders. Thus, the fail-safe feature requires a Cluster Leader to possess information on all other Cluster Leaders for the VTL Leader election process. In particular, Cluster Leaders that are participating in the election process attach an *ElectionReady* flag to their beacons; a Cluster Leader must possess beacons with the *ElectionReady* flag from all other Cluster Leaders in order to become a VTL Leader. As all vehicles are assumed to possess digital maps, each Cluster Leader is aware of the expected number of total Cluster Leaders for each particular intersection. If information from all approaching Cluster Leaders is not received, a safe VTL Leader election process can not take place. Hence, vehicles are required to perform *a fallback approach*, by braking in front of the intersection and crossing it in a First Come, First Served (FCFS) manner. Naturally, a road segment can be empty, and thus no packets can be

received. In this case, a fallback approach results in the unnecessary loss of traffic efficiency but maintains traffic safety.

- In addition, information from Cluster Leaders has to be timely limited to avoid multiple and inconsistent VTL instances. In particular, a Cluster Leader must possess beacons with the *ElectionReady* flag from all other Cluster Leaders, and must let its own *ElectionReady* information expire prior to becoming a VTL Leader. If all participating vehicles possess the required information, a vehicle, e.g., with highest ID can become a VTL Leader. Similarly, waiting for information to expire increases the time for the election and impacts traffic efficiency, but is required for safety reasons, as shown during the failed verification process, see Section 5.1.4. In particular, if a Cluster Leader becomes a VTL Leader (based on its, currently, highest ID) its delayed beacons can participate in election process started later by a new vehicle, who might have higher ID and who becomes a VTL Leader itself.

- Once a VTL Leader is elected, it starts broadcasting current traffic light phase, including phase duration. VTL Leaders are not allowed to change already published phase durations, as packet reception cannot be guaranteed.

Fail-safety features for handover: Once the stopping phase of the current VTL Leader expires, it can leave the intersection by handing over its leadership to another vehicle. It is possible to have an overlapping time period when two VTL Leaders exist, or to have a time gap between the two leaderships. As latter approach would cause more vehicles to use fallback approach, we choose an overlapping leadership for the handover process. However, a fail-safe feature is required to make sure no conflicting information is broadcast by two VTL Leaders during their shared leadership. A VTL Leader does not need to perform a handover, and can simply leave the intersection; its virtual traffic information will then simply expire after a certain period of time.

- An old leader decides on a new VTL Leader and sets a handover flag in its beacons, which has an expiry time. Upon reception of such beacons, the new VTL Leader lets its *ElectionReady* flag expire and becomes a VTL Leader. The new VTL Leader is not allowed to change the VTL information until old VTL information expires, as communication cannot guarantee that all vehicles receive new information. The overall handover process has to be limited to the time when the old VTL Leader leaves the intersection, otherwise multiple and inconsistent VTL Leaders can be created—as also learned from failed verification, see Section 5.1.4.

Resulting application logic

The resulting application is the outcome of requirements analysis, initial modeling, and failed verification attempts (the verification allows to trace back to conditions that lead to errors, see Section 5.1.4).

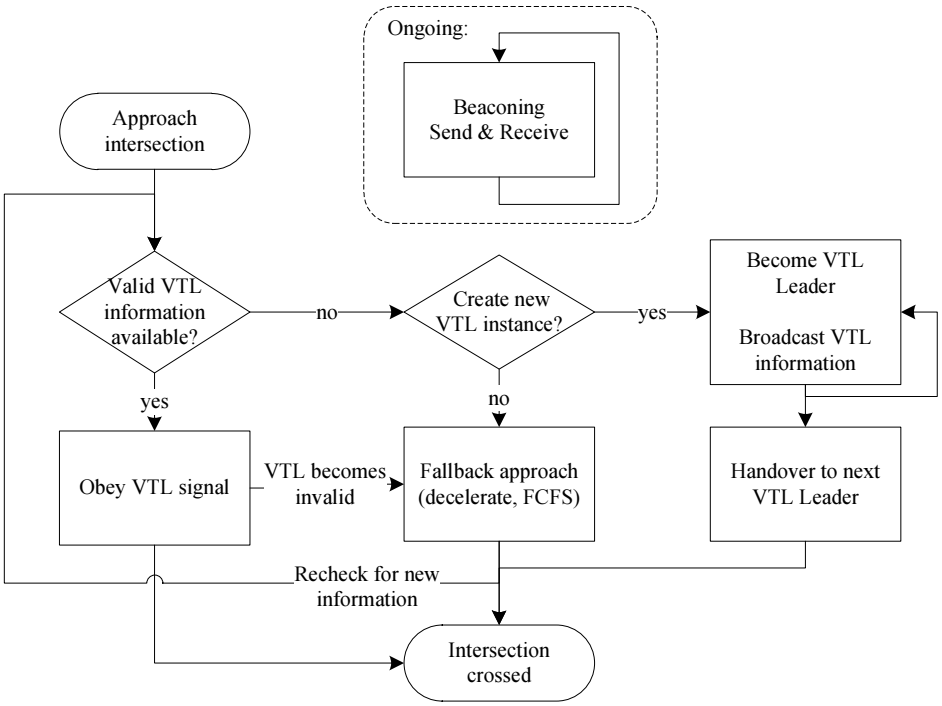


Figure 5.2: Flowchart of a fail-safe Virtual Traffic Lights application

The Virtual Traffic Lights application follows these abstract steps:

- Upon approaching an intersection, vehicles try to determine whether there is already a vehicle, called a *VTL Leader*, that controls the intersection, and broadcasts traffic light information;
- If there is a VTL Leader, all vehicles receiving its messages obey the traffic signal;
- If there is no VTL Leader, vehicles determine, among possibly several approaching vehicles, called *Cluster Leaders*, which one takes the VTL Leader role;
- If no VTL Leader is present and it is not possible to determine a new VTL Leader because of insufficient information, vehicles brake at safety distance D_{safe} and cross the intersection in an FCFS manner, i.e., they perform *a fallback approach*;
- A VTL Leader can, before leaving the intersection, perform a handover so that another vehicle becomes the VTL Leader and overtakes control of the intersection.

Figure 5.2 depicts the VTL protocol as a state change diagram. The two most critical aspects, as they should take place under NLOS communication conditions, are the leader election and the handover.

The two main requirements of the VTL protocol, defined in Section 5.1.2, together with fail-safe features serve as a basis for the resulting protocol. The first requirement—drivers get notified on the current traffic light phase at latest at safety distance D_{safe}—cannot be always guaranteed. If at the safety distance D_{safe} no information on the traffic light is available, the VTL Leader election did not, yet, take place. VTL Leader election requires information from all participating Cluster Leaders, if such information is not available (due to unreliability of communication or absence of other Cluster Leaders), the fail-safety feature requires that vehicles perform a fallback intersection approach to maintain safety. In such a way, the fallback intersection approach maintains fail-safety in case the first requirements cannot be satisfied. This is also depicted in Figure 5.2: if no valid VTL information is available and it is not possible to create a new VTL instance, a vehicle decelerates and intersection is crossed in the FCFS manner.

The second requirement—traffic lights information has to be consistent—is easy to fulfill with a single instance that is responsible for the intersection management. For a VTL protocol it is therefore necessary to ensure that at most one vehicle is a VTL Leader at an intersection at each moment, except during the handover process. Hence, a vehicle may only become a VTL Leader if it can guarantee that no other vehicle becomes the VTL Leader as well. The decision *Create New VTL Instance?* shown in Figure 5.2 is made as follows: The number of vehicles that can become a VTL Leader is limited to those vehicles that are Cluster Leaders. In order to prevent multiple VTL instances, a vehicle can only become a VTL Leader if it possesses valid information from all other road segment's Cluster Leaders and if it has the highest unique ID among them. By satisfying the second requirement this procedure also ensures fail-safety in case of unreliability of communication, as beacons from Cluster Leaders can be lost. In case information from all approaching road segments is not available, a vehicle has to perform the fallback approach. Consistent traffic light information during the handover process is also supported by a fail-safety feature, described earlier. Namely, a new VTL Leader is not allowed to change the traffic light signal within an interval defined by the old VTL Leader.

Consequently, the following differences to the VTL protocols described in [FFC+10] and [NAT+12] exist:

- The requirement to receive beacons from all possible Cluster Leaders for the VTL Leader election process;
- Limited validity of VTL information, e.g., beacons from Cluster Leaders that participate in the VTL Leader election process and VTL handover beacons;
- A Cluster Leader with the highest ID becomes a VTL Leader, rather than a Cluster Leader which is the closest to the intersection;
- VTL Leaders are not allowed to change the duration of already published traffic light phase;
- Vehicles are required to perform a fallback intersection approach in case one or more of the required conditions cannot be timely met.

5.1.4 Verification

We choose to perform a formal verification with *model checking* in order to make sure that the resulting VTL design is in accordance with its requirements. As has been mentioned in Section 2.1.3, verification of driver assistance systems has to consider three main aspects – movement, application's protocol, and communication. The safety must be verified on all three levels, but the differences in the used operands (e.g., continuous values and discrete events) make the verification of vehicular applications challenging. After a short introduction into the model checking we describe how we were able to perform verification on all three levels, followed by the verification results.

Background: model checking, PROMELA, and SPIN

Model checking is a model-based verification technique where a system behavior is modeled in a mathematically precise and unambiguous manner. Model checking, given a model of a system and a formal property, systematically checks whether this property holds throughout the model. This verification technique explores all possible system states in a brute-force manner, revealing even the subtle errors that might remain undiscovered using emulation, testing, or simulation. Naturally, verification with model checking is only as good as the system's model is [BK08].

Several tools exist for model checking; among one of them is SPIN (**S**imple **P**romela **IN**terpreter) with its modeling language PROMELA (**PRO**tocol **ME**ta **LA**nguage), introduced in the late 80's by Holzmann et al. [Hol91]. We choose to use the SPIN tool because of its support of communication, powerful process modeling, and mature technology.

The modeling language PROMELA allows defining a system model in a C-like manner. PROMELA utilizes concurrent processes (defined by `proctype`), variables of different data types (e.g., `bit`, `bool`, `byte`, `short`, etc.), communication channels (defined by `chan`), and various statements (e.g., `if`-statement). The `assert`-statement is used within PROMELA to check whether a certain property or invariance is valid in a certain state or throughout the model. *A state* of a model (in PROMELA) consists of a set of values of all its variables (including location counter) and is stored in *a state vector*. The PROMELA model begins with an initial state (marked by `init`), which typically initiates the creation of several concurrent processes. Each process has a list of variables, channels, and *a sequence of statements* that describe the behavior of the modeled system. In addition, PROMELA allows to model sequence of statements that should be executed indivisibly and non-interleaved with other processes with the help of `dstep` or `atomic` sequences. The sequence `dstep` is used for deterministic fragments and `atomic` sequence allows non-determinism. For example, probabilistic packet reception can be modeled in the following way:

```
atomic {
        if
        :: sent == true -> skip
        :: sent == true -> receive()
        fi;
}
```

When both options of this `if`-statement can be executed, i.e., `sent == true`, one of it will be randomly executed, either `skip` (do nothing) or `receive()`, i.e., reception of a packet.

SPIN interprets PROMELA models and can work both as a simulator and a verification engine. SPIN simulation can be used to perform a sanity check or validation, whereas SPIN verification uses a Depth-First Search algorithm to generate and explore the complete *state space* that can *possibly*[1] occur starting with the initial state. Several approaches are provided in SPIN to check for invariance, e.g., SPIN can check for invariance in every possible state, or whenever variables involved in the `assert`-statement may change, with the latter option being substantially more efficient. If only one assertion evaluates to false, the SPIN verification program will stop with an error message. Traces are provided to trace back to conditions that lead to a failed verification. It is possible not only to verify assertions but also to search for invalid end states, deadlocks, livelocks, unreachable code, etc.

Although temporal aspects can be modeled with PROMELA, SPIN's simulation and verification do not proceed on the time-scale but rather traverse through possible model states; contrary to the discrete-event simulations typically used for modeling of communication networks.

SPIN provides two variations for verification: exhaustive search and bitstate hashing. An exhaustive search verification begins with an initial state, evaluates every possible next system state, and checks whether the defined invariant holds true. To avoid multiple evaluations of the same state, every evaluated state must be stored in a state vector. Although this is feasible for small models, the verification of larger models can require a lot of memory for the state storage. For example, a model that requires 100 Byte to store one system state and has a total number of 10^9 reachable system states requires over 93 GByte of memory. Although compression mechanisms can reduce the memory consumption, larger models easily reach the memory limits of today's hardware.

For models that are too large to be exhaustively verified, a bitstate search can be used. Instead of storing each model state completely, a hash function is used to map each system state to a single number between zero and $2^w - 1$. The constant w is chosen so that 2^w Bit would fit in the computer's memory. During verification, the corresponding bit of the calculated hash value of each evaluated state will be set in a bitmap. To check for already evaluated states, the verification algorithm hashes the current state and checks whether the corresponding bit in the bitmap is already set. Therefore, the storage of each state requires only one Bit of memory. The big drawback is that it is possible for *hash collisions* to occur (i.e., two different states are mapped to the same hash value). If this happens during verification, the algorithm will treat a new state as a known state and therefore will not evaluate it. Hence, the search is not necessarily exhaustive but a partial, randomized search. The coverage (the number of reached system states divided by the number of reachable system states) of the bitstate search strongly depends on the size of the bitmap, as the likeliness of hash collisions decreases with larger bitmap sizes. Hence, choosing a large value of w is recommended for best coverage. The bitmap's occupancy ratio (the number

[1] not all states are reachable from the initial state.

of visited states divided by the total number of bits, is also called a hash factor, h) serves as an indicator for the search coverage. A hash factor close to 1 indicates a full bitmap, which causes too many hash collisions, whereas a large hash factor ($h > 100$) is an indicator of a good coverage [Hol91].

More details on model checking in general as well as on PROMELA and SPIN in particular, are available in [Hol91], [Holo3], [BA08], and [BK08].

Verification method

For verification of the VTL protocol we modeled a crossing of a single intersection. At the beginning the desired number of vehicles are created as separate processes. Each vehicle or process has a certain number of variables that describe its model state. For example, the variable `road` is randomly selected for each vehicle process to assign a vehicle to one of the roads. In addition, a communication channel is assigned to each vehicle for sending and receiving of beacons. The VTL application behavior sketched in Figure 5.2 is modeled with PROMELA with the help of a sequence of statements that utilize assigned variables and channels. The constructs `dstep` and `atomic` are utilized to mark sequences of statements that should be executed indivisibly either deterministically or non-deterministically. For example, probabilistic packet reception is modeled as an `atomic` statement.

Recall the two requirements of the VTL protocol:

1. Drivers get notified on the current traffic light phase at latest at safety distance D_{safe}
2. Conflicting vehicles must not get the right of way, i.e., green light, at the same time. In other words, traffic lights information has to be *consistent*.

If the first requirement cannot be met, i.e., no traffic information is available at the safety distance D_{safe} the integrated fail-safety feature requires vehicle to decelerate and stop in front of the intersection, followed by an FCFS intersection crossing. If the second requirement is not met, then the protocol is **not safe.** *Hence, it must be verified that traffic light inconsistency never occurs.*

As it is fair to assume that vehicles are aware of their own safety distance, it must be only verified that the VTL information available to the vehicles is consistent; if no information is available vehicles perform a fallback approach. The ability of a vehicle to detect crossing of their own safety distance D_{safe}, which triggers either display of VTL information or a fallback approach, eliminates the need to explicitly model the safety distance and further verification of continuous vehicle kinematics.

The consistency of traffic light signals is ensured if vehicles approaching an intersection either possess the same traffic light signal set or do not possess any traffic light signal at all, as vehicles perform a fallback approach in this case. *A traffic light signal set* consists of the traffic light phase for each approaching lane of the intersection. This is defined formally: Let V be the set of all vehicles at one intersection and $v \in V$ be a vehicle. A traffic light signal set $s = (l_1, ..., l_n)$ is a tuple that indicates the

current light phase for each lane and direction ($l_{1,\ldots,n} \in \{\text{red}, \text{green}\}$; n is the number of approaching lanes); and S denotes the set of all possible traffic light signal sets.[2] For each vehicle v, $s_v \in S \cup \{\perp\}$ denotes the traffic light signal set possessed by v. If a vehicle does not possess any valid traffic light signal set, then $s_v = \perp$.

The consensus invariant that must hold is formulated as following:

$$\exists \sigma \in S \; \forall v \in V : s_v = \sigma \vee s_v = \perp$$

The defined invariant is modeled as an `assert`-statement which is called explicitly whenever a new VTL beacon is received or whenever a VTL Leader assigns a green light to a road. The goal of verification is to evaluate all possible system states and to ensure that the defined invariant is never violated. For example, after a VTL Leader assigns a green light to a road (using the variable `tgreen`), the `assert`-statement checks for every vehicle (variable `i` is used to iterate through all vehicles), that the information on which road has currently green (stored in the variable `greenRoad[i]`) is the same as what has been announced by the VTL Leader. Vehicles performing a fallback approach are allowed not to possess such information, in this case `greenRoad[i]` is set to 255. The checked assertion looks as following:

```
assert(greenRoad[i] == tgreen || greenRoad[i] == 255)
```

Consequently, aspects relevant for verification of a communication-based VTL application that are related to application protocol, communication, and movement of vehicles, are considered in PROMELA.

In Table 5.1 we summarize modeled variables used to describe a single *state* including their data types. Although a state of each individual vehicle consists of most of the presented variables, some variables are shared among all vehicles to accelerate the evaluation, e.g., `greenRoad[]`, `isLeader[]`. The safety distance D_{safe} is modeled only implicitly, hence there is no separate variable representing this distance, see above. Based on the information provided in the table it is possible to estimate the upper limit of number of states that can be evaluated and the lower limit of the space required to store each state. It should be noted that not all state variations are occupied and that the actual number of states that are checked by SPIN is limited by the modeling. As an illustration, assume *a toy example*: we want to model just *one* vehicle, *two* roads, and *three* beacon messages. Following variables are used: `road` and `msg` to represent the road ID and the message sequence number. Both have the data type `byte`. A vehicle is modeled as a process that can be "located" at one of the two possible roads, i.e., its variable `road` can have a value 0 or 1 and can "possess" only one of the three possible messages, i.e., its variable `msg` can have a value of 0, 1 or 2. In such a way, the number of checked states is $(2 * 3)^1 = 6$, which requires $6 * 2 = 12$ bytes (each state requires 2 bytes to store two variables of 1 Byte each, i.e., a state vector size is 2 bytes). If we would model *three* vehicles, *two* roads, and *three* beacon messages, we need to create one process per vehicle. This increases the number of states that has to be checked to $(2 * 3)^3 = 216$ which requires $216 * 2 = 432$ Bytes. This makes it clear,

[2] S includes signal sets that must not occur (e.g., all lanes green). The protocol assumes that VTL Leaders do not broadcast such signal sets with the help of map data.

Variable	Data type	Size (Bits)	Comment
id	byte	8	vehicle ID
road	byte	8	road ID
myChan	chan	41	communication channel holds one message of type {mtype, byte, byte, bool, byte, byte}
byhandover	bool	1	used by VTL Leader to handover
isCL	bool	1	indicates if a vehicle is a Cluster Leader
msg	mtype	8	defines symbolic names for different messages {BEACON, VTLINFO, HANDOVER} stored as unsigned char
senderId	byte	8	indicates sender ID
senderRoad	byte	8	indicates road ID of the sender
senderIsCL	bool	1	indicates if sender is a Cluster Leader
senderGreenRoadId	byte	8	indicates road ID of the old VTL Leader (during handover)
senderHandoverTo	byte	8	indicates an ID of a new VTL Leader (during handover)
lgreen	byte	8	indicates a next road to get green
handoverto	byte	8	indicates a handover vehicle
maxId	byte	8	used for VTL Leader election
tgreen	byte	8	indicates a road ID that has green
i	byte	8	used to iterate through vehicles or roads
isLeader[]	bool	1	indicates if a vehicle is a VTL Leader
greenRoad[]	byte	8	used by each vehicle to store a road ID that has green light
clusterLeader[]	byte	8	used to store a Cluster Leader ID for each road
informationMap[]	byte	8	used to store other Cluster Leader IDs received for each road

Table 5.1: Modeled state variables and their data types

that the number of variables that describe a state but especially the overall number of concurrent processes, have a strong impact on the total number of checked states and the required memory. Note that 432 bytes required to store all states in this example, is the lower limit, indicating a space needed only for the variables; for each state a

location counter is also stored. For comparison, the *maximum* number of all states that can be occupied with the `byte` data type for two variables is $2^8 * 2^8 = 2^{16}$ which requires $2 * 2^{16}$ Bytes to store only the state variables excluding the location counter.

A toy example with only 2 variables results in total of 216 states requiring 432 Bytes. It becomes clear that all variables used to model a VTL protocol's state, summarized in Table 5.1, first of all, lead to a large number of possible states, second of all, require a lot of storage. The number of states that has to be checked, and as a consequence the required space, is increasing substantially when the number of vehicles is increased, namely, *the number of possible states*^{n}, where n reflects the number of processes or vehicles. This makes exhaustive verification for a large number of vehicles rather unrealistic.

Prior to verification, SPIN simulation allows to validate the PROMELA model for major VTL functionality.

Verification results

Verification has been performed for *one* intersection and varying numbers of vehicles and approaching roads, summarized below:

Scenario A: two vehicles and two one-way roads

Scenario B: three vehicles and two one-way roads

Scenario C: three vehicles and three one-way roads

Scenario D: four vehicles and two one-way roads

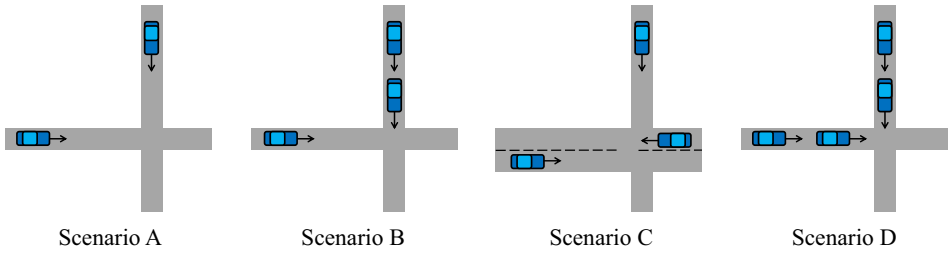


Figure 5.3: A possible scenario layout for the Virtual Traffic Lights verification

Although, the exact geometry of the verification scenario is not important for SPIN since vehicles do not "move" in time and space, we exemplary illustrate how a scenario could look like in Figure 5.3. Note that due to the modeled non-determinism, vehicles can be placed differently to the illustration, e.g., all on the same road.

All verification runs were performed on a 2.67 GHz Intel Xeon E7–8837 machine equipped with 512 GByte of main memory. The version of used SPIN tool is 6.2.4. The SPIN evaluates all possible system states, checks invalid end states, deadlocks, and assertion violation. Table 5.2 lists the number of evaluated system states as well as the

Scenario	Output			
	States	**Errors**	**Memory**	**Comment**
A	267, 441	0	34 MB	Exhaustive
B	$5.42 * 10^9$	0	216 GB	Exhaustive
C	$1.91 * 10^{10}$	0	256 GB	Bitstate, $h = 115$
D	$> 10^{11}$	0	256 GB	Bitstate, $h < 22$

Table 5.2: VTL verification results

required memory for each scenario. Most importantly, no violation of the consistency constraint or deadlocks could be found by SPIN in any scenario.

When the number of vehicles is increased, the verification complexity increases drastically: whereas the scenario with two vehicles only required 34 MByte of memory and 1.37 s of computation time, adding one vehicle to the scenario caused the verification to require 216 GByte of memory and 7.2 h of computation time. Because of the hardware memory limit of 512 GByte, only a bitstate verification could be performed for larger scenarios. The scenario with three vehicles and three roads could be verified with sufficient coverage as indicated by a hash factor of 115. However, scenarios with four vehicles or more could not be verified with sufficient coverage because of memory restrictions.

5.1.5 Discussion

Although the idea of devising a communication-based decentralized intersection management protocol has been researched for some time now, a fail-safe and verified protocol is not openly available. In this section we describe not only the detailed VTL protocol design together with its requirement analysis and fail-safety aspects but its formal verification using the model checking. The protocol design and verification, to some extend, are performed in parallel, as the verification process sheds light on, at first, non-intuitive problem states. Model checking allowed to consider not only the application's protocol and communication aspects but also to integrate "movement" of vehicles. As vehicles are aware of their safety distance D_{safe}, we model either obeying of the VTL signal, if valid VTL information is available or a fallback approach, if no information is available. The consistency of available VTL information is verified by SPIN. Although the verification could only be successfully performed for small scenarios, the results indicate that the presented protocol is indeed safe and consistency of available information is never violated.

It must not be ignored that model checking only verifies a system *model* and not the actual system itself; any obtained result is thus as good as the system model with no guarantee of completeness [BK08]. Further testing and verification is required to verify hardware and software aspects of actual implementation. For example, *an equivalence check* is often used to formally verify that two models are equivalent. In addition, either different levels of system abstraction can be modeled or only main functionalities of the system, which can make formal verification for large and complex systems feasible.

Nevertheless, the designed and formally verified model provides a solid foundation for further evaluation. The VTL implementation in a network simulator, used for evaluation presented in Section 5.2, is based on the verified model, although not verified in a formal way on its own. The formal verification with, e.g., equivalence check is left for future work.

5.2 Evaluation of a VTL application

The VTL protocol, which design and verification is presented in Section 5.1, is evaluated in order to understand how VTL application performs in realistic environment. In particular, strict fail-safe features integrated in the design may result in frequent fallback approaches, which can eliminate the benefit of relying on virtual traffic lights and be equivalent to already existing FCFS approach.

5.2.1 Applied methodology

The complexity of the VTL protocol requires the use of simulation approach for the performance evaluation. Various models to depict realistic environments are required, in particular the networking model, the mobility model, the radio propagation model, and the driver behavior model. The verified VTL protocol is implemented in the NS-3 network simulator[3], and the vehicles' movement is modeled by the Intelligent Driver Model (IDM) [THH00]. Both models proved themselves valid for evaluation of IEEE 802.11p vehicular communication networks and have been extensively used in the research community.

Radio propagation in intersections is marked with NLOS conditions; complex and sophisticated models are required to depict radio propagation in such environments. The VirtualSource11p [MKH11] radio propagation model, which is based on the real-world measurements in the intersection environment, is chosen for our evaluation. The VirtualSource11p provides path loss calculation in the NLOS environment, which is the result of empirical measurements fitting. In addition, the VirtualSource11p models fading in NLOS according to normal distribution with $\sigma = 4.1$ dB, as it matched the measured values the best. The authors of [MKH11] indicate the *inter-building distance* (IBD) as one of the main parameters that affect radio propagation at intersections, especially for communication between intersecting roads. Larger IBDs improve radio propagation conditions, whereas smaller IBDs impair them. In [MSKH11] the authors indicate the most common inter-building distance clusters for the city of Munich. For example, for 50 % of all 4-leg intersections the distance from intersection center to the nearest building is ≈ 17.8 m, which corresponds to inter-building distance of ≈ 25.2 m. For 75 % of all 3-leg intersections the distance from intersection to the nearest building is similar as for the 4-leg case and is ≈ 18.7 m. The VirtualSource11p accounts the

[3] NS-3 version 3.15, PHY parameters are set according to the default values of YansWifiPhyHelper, except EnergyDetectionThreshold which is set to −92 dBm. In addition, an OFDM WiFi model with 6 Mbps data rate and bandwidth of 10 Mhz is used. The code is provided under http://dsn.tm.kit.edu/english/vtl.php

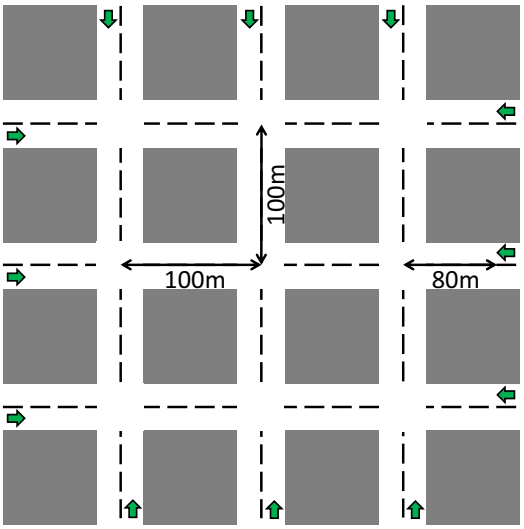


Figure 5.4: The 4x4 Manhattan grid scenario; green arrows represent vehicle sources

inter-building distance for the path loss calculations and the inter-building distance of 22 m has been chosen for our evaluation, unless stated otherwise.

As for the driver behavior model – an obedient, not distracted driver, is modeled, see Section 5.1.3.

The analyzed road layout represents a $4x4$ Manhattan grid, where each road has one lane per direction, cf. Figure 5.4. The distance between intersections is equal to 100 m. At each end of the road, 80 m away from the next intersection a vehicle *source* is located, totaling 12 sources. Generated vehicles, which are 5 m long, simply drive straight without overtaking or turning, until they cross the whole grid and disappear at the other end. Vehicles are driving at their desired speed—50 km/h—from the moment they are generated from a source. The deceleration and acceleration is performed according to the IDM and has the following parameters: desired deceleration is equal to $-2\,\mathrm{m/s^2}$, desired acceleration is equal to $1\,\mathrm{m/s^2}$, and maximum deceleration is limited to $-8\,\mathrm{m/s^2}$.

Each simulation runs for 300 s with additional warm-up time of 180 s during which no statistics is being gathered. Different seeds are used for each run, although common random numbers are used to evaluate the impact of one parameter. For each evaluation a total of 10 runs is performed, and statistics from all vehicles are evaluated together.

The following metrics represent a special interest for the current evaluation:

Definition 5.1. ***Throughput*** *is measured as a ratio between the inflow of vehicles entering the road network (vehicles per minute, per source) and the outflow of vehicles leaving the road network (vehicles per minute, per sink).*

Definition 5.2. ***Average travel time*** *represents the mean time a vehicle needs to travel though the road network, measured in seconds.*

The general performance evaluation methodology presented in Section 3.5 is also followed here:

1. **Determining minimum transmission parameters:** The requirements of the VTL application defined in Section 5.1.2 provide orientation for necessary transmission parameters. In particular, a safety distance D_{safe} is defined at which vehicles should be aware of the current traffic light phase in order to either safely cross an intersection or safely come to a stop in front of the intersection. Hence, the safety distance D_{safe} represents a minimum distance at which communication should take place. In addition, the time a vehicle needs to reach its safety distance can determine a time interval during which the traffic light information should be received. Contrary to the RECA application, information on the traffic light does not have to be updated in the immediate proximity to the decision point at safety distance, as traffic light information does not change unexpectedly. In addition, the phase duration is assumed to be included with information on the traffic light signal. Moreover, multiple vehicles can approach an intersection, leading to a very dynamic and complex transmission parameter adjustment. Hence, the default transmission power of 20 dBm[4] and transmission rate (Tx. rate) of 10 Hz is chosen for evaluation. The adjustment of transmission rate is also discussed.

2. **Determining the number of communicating vehicles:** The considered scenario can experience various vehicle densities, which can be dependent on geographical location or the time of the day. Hence, an evaluation with varying number of communicating vehicles has been performed. The time interval between vehicles entering the simulation grid from each source is exponentially distributed with a mean value varying between 1 and 10 vehicles per minute, per source. In such a manner, we use different values for *vehicle inflow*, which is a number of vehicles entering the simulation grid per minute, per source.

3. **Determining the reliability of IEEE 802.11p communication to support an application:** The fail-safe VTL integrates a fallback mechanism in case the application's requirement (to possess traffic light information before reaching the safety distance) is not fulfilled. This requirements might not be fulfilled due to the absence of other vehicles or due to unreliability of communication and loss of sent packets. The fallback mechanism assumes the worst, i.e., unreliability of communication, and forces a vehicle to decelerate and stop before entering the intersection followed by an FCFS intersection crossing. With such an approach the unreliability of IEEE 802.11p communication does not impact safety of the vehicle or traffic in general. However, the fallback intersection

[4]The transmission power of 20 dBm corresponds to a communication range of ≈ 60 m from the intersection center under the NLOS condition calculated with VirtualSource11p radio propagation model [MKH11], when a carrier sense/decoding threshold is $-90dBm$, the inter-building distance is 22 m, and the sender is 60 m away from the intersection center. Assuming a vehicle is approaching an intersection with 50 km/h the safety distance is equal to ≈ 49 m which is within the communication range resulted from the chosen transmission power.

approach deteriorates traffic efficiency, often unnecessary, as intersection might be indeed empty. We propose *an optimized VTL protocol* that allows vehicles to decelerate only until a certain *reliability* of successful communication is reached and afterward proceed with the desired speed. Communication reliability for a VTL application is defined as following:

Definition 5.3. ***Communication reliability*** *is a probability to detect the presence of other vehicles by means of communication. The confidence level that present vehicles are detected is referred to as* ***a safety level****.*

4. **Determining the scalability of IEEE 802.11p communication to support application and the impact on the road traffic**. The last two steps from the general evaluation methodology, described in Section 3.5, are performed here in parallel. The scalability of IEEE 802.11p communication to support non-optimized and optimized VTL is simulated for the increasing vehicle density, i.e., the inflow of vehicles is increased from 1 veh/min/source to 10 veh/min/source, with 1 step increments. The resulting throughput and travel time are compared to the throughput and travel time achieved with three conventional intersection management approaches, namely, **FCFS**, **CTL**, and **CTLsync**. The **FCFS** approach depicts *an all-way stop*, which realizes the first come, first served principle; the **CTL** approach represents conventional traffic light, when phase switching is performed in a static, pre-timed (fixed phase duration of 30 s), and *asynchronous* manner, i.e., random switch at each intersection; the **CTLsync** represents the same conventional traffic light approach, when phase switching is done *synchronously*, i.e., green or red wave for all horizontal or all vertical roads.

5.2.2 Results

Figure 5.5 shows the resulting throughput; it is evaluated whether the number of vehicles that leave the grid is the same as the number of vehicles entering the grid. If the inflow is higher than the outflow, the road network is becoming congested. As shown in Figure 5.5, all intersection management approaches perform similarly for vehicular densities below five vehicles per minute, per source. As vehicle inflow grows, FCFS starts to throttle, whereas VTL still performs close to the optimum until vehicle inflow is approximately seven vehicles per minute, per source. With the increased inflow, VTL as well as conventional traffic light approaches, begin to deteriorate because the road network is now saturated, although VTL is slightly worse. These results indicate that VTL can outperform FCFS approaches and perform close to CTL with respect to throughput.

Figure 5.6 depicts the *average travel time* that vehicles require to cross the grid from the moment they enter at the source until they reach the sink. For small vehicle inflow, i.e., 1–2 vehicles per minute, per source, VTL's performance is similar to that of FCFS and asynchronous CTL. Sparse vehicle density is causing vehicles to switch to the

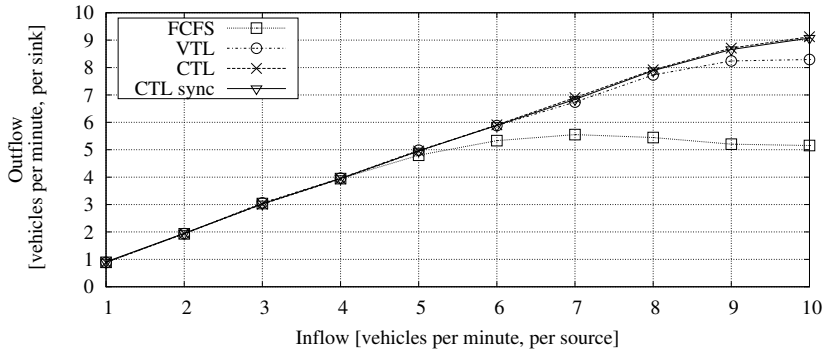


Figure 5.5: Throughput under various vehicle inflow. The IBD is 22 m, the Tx. rate is 10 Hz, and desired speed is v = 50 km/h

fallback option, as no VTL instance can be established in most of the cases. The benefit of asynchronous CTL is also not noticeable at low vehicle densities. Synchronized CTL approach performs by far the best. For comparison, the absolute best travel time for one vehicle that does not have to stop at any intersections is 25.7 s (360 m divided by 14 m/s). With the increase in traffic density, FCFS performs noticeably worse than VTL, because VTL vehicles are now able to establish VTL instances and manage intersections without going into fallback. VTL performs similarly to asynchronous CTL; synchronized CTL, however, still performs best.

Throughout all performed simulations vehicles performed over 150.000 intersection approaches (due to red light or FCFS approach), where the required deceleration did not exceed -2 m/s², with the majority of the values being less than -0.5 m/s². This indicates that VTL application does not cause late, unsafe decelerations.

The results show that the verified VTL protocol performs better than FCFS with respect to throughput and travel time. However, for low vehicle densities VTL is worse than the synchronized CTL approach because of the lack of vehicles on all other road segments. Not detecting vehicles on other road segments could be the result of lost messages or the intersection is really empty. It is possible to optimize the VTL protocol's efficiency by detecting that an intersection is in fact empty and there is no need to brake. This is discussed below.

Tradeoff safety level vs. travel time

Performance gains, mainly for low vehicle densities, are possible if more VTL instances could be created and vehicles would not have to brake at every single intersection performing the fallback approach. A vehicle that crosses an intersection without coming to a stop, must ensure that there is no conflicting vehicle approaching the intersection from other road segments. It is therefore essential for vehicles to be able to "detect" whether a conflicting road is in fact empty or whether all beacons originating from vehicles on a conflicting road got lost, as a result of bad radio channel conditions.

With the help of the VirtualSource11p path-loss and fading model [MKH11] it is possible to determine the probability of packet reception (PPR) of a single packet

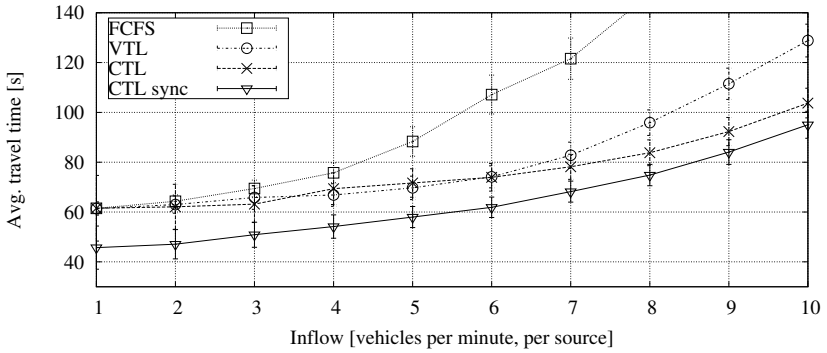


Figure 5.6: Average travel time vs. vehicle inflow with 95 % confidence interval. The IBD is 22 m, the Tx. rate is 10 Hz, and desired speed is v = 50 km/h

based on an average reception power and a receiver reception threshold. This allows to calculate the safety level which is calculated by approaching vehicles as the cumulative reception probability for at least one beacon, assuming that a present sender vehicle is approaching the intersection. In particular, a packet is received if the received power is larger than the received threshold. The average probability of a single packet reception can be derived from 1 – *CDF* (cumulative distribution function) of the received power distribution, which is based on the path-loss model calculated by VirtualSource11p and fading modeled by a normal distribution. The PPR for a single packet allows to calculate the probability that a vehicle does not receive any packets (non-reception probability) from conflicting road segments during intersection approach. For this, for each transmitted packet and depending on the sender-receiver position, non-reception probabilities are multiplied with each other, resulting in cumulative non-reception probability. In addition, observation distance and time are limited. The safety level, see **Definition 5.3**, is equal to *one minus cumulative non-reception probability*. The safety level quantifies the vehicle's estimation on "the emptiness" of another road segment. The higher the safety level is, the higher is the probability that the other road segment is, in fact, empty. For calculation of the safety levels we assume that vehicles travel at their maximum speed (50 km/h) and approach the intersection simultaneously (the worst case).

Two possibilities to modify the fail-safe VTL protocol in order to improve its efficiency while relaxing the safety constraint, i.e., allowing safety level < 100 % are considered: *Adaptive transmission rate* and *Adaptive braking distance*.

Adaptive transmission rate: To maximize traffic efficiency by avoiding unnecessary braking maneuvers, it is desirable that approaching vehicles reach a sufficient safety level at safety distance D_{safe}. This could be achieved by adapting the sender's transmission rate depending on the desired safety level and the radio channel conditions that are strongly influenced by the intersection layout.

Figure 5.7 shows the required Tx. rate for various safety levels and three different, but common, intersection layouts. Naturally, the stricter the safety level requirement, the higher is the required Tx. rate. However, even the smallest presented safety level of 0.9 requires a Tx. rate above 20 Hz for an IBD of 20 m. The highest presented safety

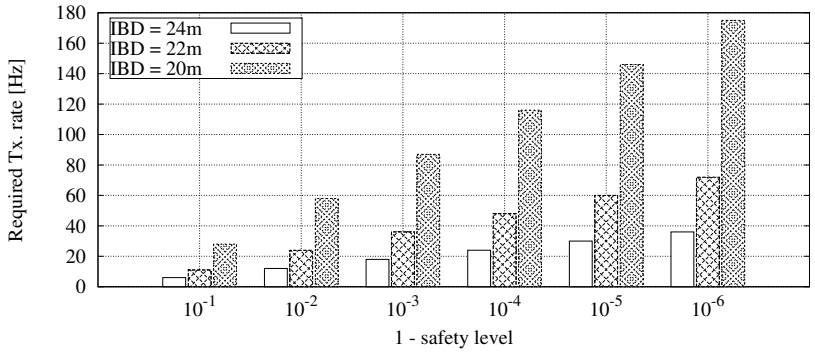


Figure 5.7: Required transmission rate vs. safety level at the safety distance $D_{safe} = 49m$ for various IBDs, desired speed is $v = 50$ km/h, and comfortable deceleration is $a_{comf} = -2$ m/s²

level requires a Tx. rate well above typically assumed communication capabilities, even for wide intersections. Another issue is that a slight change in the IBD has a non-linear effect on the required transmission rate. This could cause large errors in required Tx. rate estimation in case of slight errors in IBD information. Although this approach would maximize traffic efficiency, it is not feasible.

Adaptive braking distance: If sufficient safety level cannot be achieved at a distance of D_{safe} (e.g., 49 m at 50 km/h), vehicles must decelerate at D_{safe} in order to maintain a safe state but could continue to drive at their desired speed once a sufficient safety level is reached. The efficiency of this mechanism strongly depends on the braking distance, i.e., the distance between D_{safe} and the point where a sufficient safety level is reached, defined as D_{brake}.

Assume that all vehicles can communicate at a fixed Tx. rate of 10 Hz and that all vehicles are aware of that fact. Figure 5.8 shows the distance to intersection at which a vehicle reaches a desired safety level. The length of the arrow labeled D_{brake} represents the braking distance (approx. 8 m), which is necessary to reach the desired safety level of $1 - 10^{-5}$ for an IBD of 24 m. Higher safety levels cause vehicles to reach a safety level later and, therefore lead to longer braking distances that reduce efficiency. However, in contrast to the Adaptive transmission rate method, it is possible to achieve high safety levels at a reasonable efficiency loss. In addition, a variation of the IBD now has a limited effect on the distance that vehicles need to brake.

This mechanism is implemented to compare its efficiency with the verified and non-optimized VTL protocol and other intersection management methods. Two extreme safety levels, 0.9 and 0.99999999999, are depicted for comparison in Figure 5.9. The efficiency-optimized VTL protocol outperforms synchronized CTL for low vehicle densities and outperforms the non-optimized VTL protocol for all investigated vehicle densities. The effect of the two shown safety levels on efficiency turns out to be minimal, in the range of < 3 s or < 9 % of additional travel time in the evaluated scenario. This is non-intuitive, as one could expect a higher efficiency penalty for such a high safety level gain, considering the reduced risk of falsely not detecting a conflicting vehicle by a factor of 10 billion. The efficiency penalty for a safety

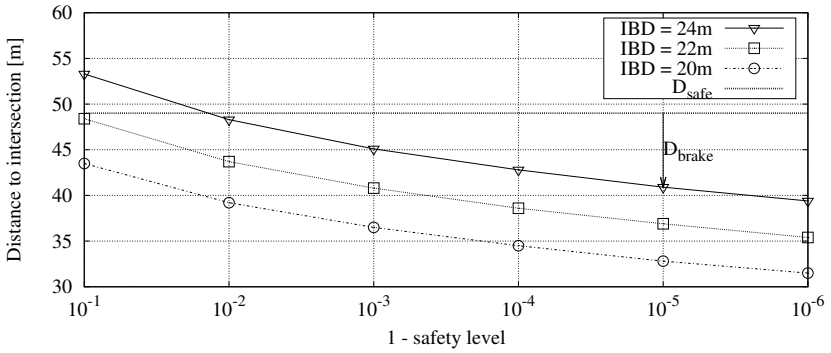


Figure 5.8: Distance to intersection at which a safety level value can be reached for a fixed Tx. rate of 10 Hz after braking at safety distance D_{safe}. The distance D_{brake} is the distance between safety distance D_{safe} and the point where a sufficient safety level is reached

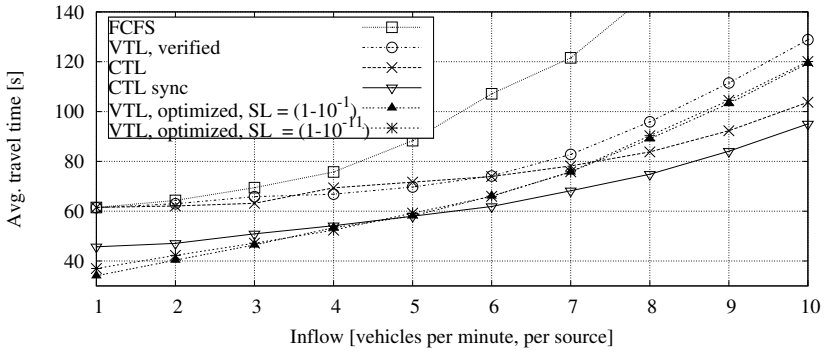


Figure 5.9: Average travel time vs. vehicle inflow. The IBD is 22 m, Tx. rate is 10 Hz, and vehicles speed is v = 50 km/h. VTL includes three configurations: verified and performance optimized with two safety levels (0.9and 0.99999999999). Confidence intervals are omitted for better visibility and are in the same order as in Figure 5.6

level increase by a factor of 1000 is around 2 % for safety levels between $1 - 10^{-1}$ and $1 - 10^{-11}$ throughout different vehicle densities. The highest efficiency penalty of 4 % is observed for the lowest inflow of 1 vehicle per minute, per source and a safety level increase from $1 - 10^{-1}$ to $1 - 10^{-4}$ (factor 1000).

Although the proposed mechanism improves VTL's efficiency, several limitations should be mentioned:

- The radio channel model only accounts for path loss and fading and not for packet collisions;
- Because of channel congestion, vehicles might have to reduce their Tx. rate.

In the following these limitations are briefly addressed. Figure 5.10 shows the effect of independent packet collision probabilities ranging from 0 to 0.8 on the distance at which a given safety level can be reached. Higher packet collision rates worsen

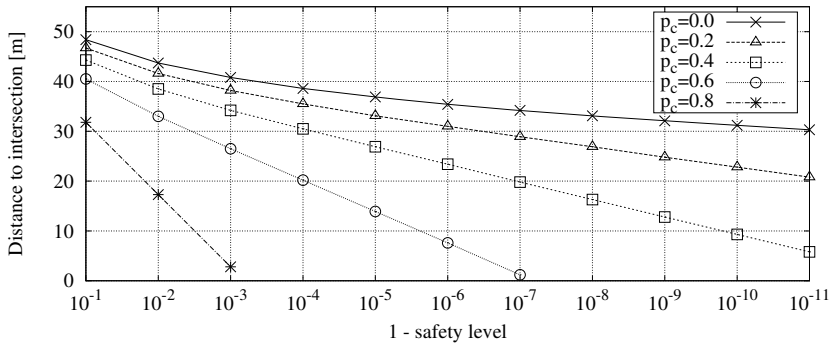


Figure 5.10: Distance to intersection at which a safety level value can be reached for various packet collision probabilities p_c for a fixed Tx. rate of 10 Hz and a constant speed 50 km/h

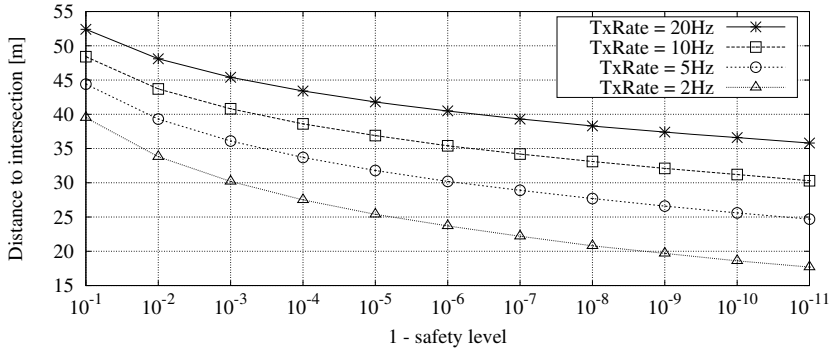


Figure 5.11: Distance to intersection at which a safety level value can be reached for various Tx. rates and a constant speed of 50 km/h and an IBD of 22 m

the performance of this optimization mechanism. However, up to a packet collision probability of 0.4, all safety levels can be achieved before reaching the intersection. The occurrence of even higher packet collision rates is unlikely during an approach of presumably empty intersection.

Figure 5.11 shows the effect of various transmission frequencies on the braking distance. By defining a minimum Tx. rate that must be obeyed by the congestion control algorithm, it would still be possible to calculate a lower bound on the safety level[5]. Channel sensing approaches could also be integrated into VTL in order to estimate the real Tx. rate of other vehicles. Although there is more room for efficiency optimization, the evaluated mechanism does improve VTL's performance while maintaining high and configurable safety levels.

[5]The minimum Tx. rate could be defined per intersection and stored in map data available for all vehicles.

5.2.3 Discussion

In this section we analyzed the traffic efficiency that can be achieved with a fail-safe VTL protocol and quantified the reliability of IEEE 802.11p communication to support VTL. Simulation results show how efficient the verified fail-safe VTL protocol is when compared with other intersection management solutions such as conventional traffic light and FCFS. Applying the efficiency optimization technique, that detects empty roads based on the non-reception of beacons, resulted in a significant efficiency gain (up to 44 %). However, the efficiency increase led to the fact that safety can only be guaranteed to a certain level. This level is referred to as *safety level*, which is used as a quantification of the capability to reliably detect an empty intersection based on communication indications.

The tradeoff between safety level and travel time has been found to be approx. 2 % of the efficiency penalty for a safety level increase by a factor of 1000. This allows the use of extremely high safety levels as specified by automotive industry standards, e.g., ISO 26262 [ISO11], with only marginal loss in efficiency. Note that safety level of 10^{-x} reflects neither the probability of failure in VTL operation nor the probability of vehicle collision but it reflects the probability of not detecting another vehicle when it is present. Proximity to an intersection center with prevailing LOS communication conditions, presence of human drivers controlling the vehicle, the possibility of using in-vehicle sensors, as well as the existence of other intersection collision avoidance applications, can support optimized VTL protocol to lead through "guaranteed" collision-less intersection crossing.

5.3 Conclusion

In the current chapter we demonstrated the design and evaluation of a fail-safe VTL application. Resulting VTL design has been formally verified utilizing the model checking approach. Although formal verification is feasible only for a small number of vehicles, it indicates the general safety of the designed protocol. Simulation-based evaluation showed that the VTL application can be reliably supported by IEEE 802.11p communication and result in the reasonable traffic efficiency achieved by conventional intersection management approaches.

As it has already been stated, the VTL application is a very ambitious application that requires a 100 % equipment ratio. Other traffic players, such as bicyclists and pedestrians, as well as non-equipped vehicles could be integrated through personal electronics, e.g., smartphones [NVT13].

Future work can focus on further efficiency optimization techniques, e.g., incorporation of channel congestion control and dynamic transmission rate adaptation. In addition, future work may extend the protocol to allow several crossing vehicles on the intersection area simultaneously, as long as their safety can be guaranteed. Traffic light phase scheduling optimization has a potential to improve efficiency but has to consider fairness among all vehicles, e.g., by considering overall stopped time. In the analytical studies the packet reception probability is assumed to be independent, but the effect of burst errors on application should also be studied.

6
Conclusion and Outlook

Nowadays, as never before, innovative technologies revolutionize every aspect of our lives, – the way we work, organize our free-time, and communicate. Likewise, inter-vehicle communication is going to effect the way people travel. Just as we can no longer imagine our lives without smartphones and e-mails, in a not-too-distant-future, it will be unimaginable to drive a car that does not wirelessly communicate with the rest of the world. The potential of inter-vehicle communication to improve safety and efficiency of driving via driver assistance applications has motivated the work behind this thesis. Various driver assistance applications are envisioned to provide information to the driver and even take over vehicle control to improve traffic safety and efficiency. Due to active research for the past decades, many milestones have been reached: the IEEE 802.11p standard that defines major communication parameters has been approved, successful FOTs are carried out around the world, and the first IEEE 802.11p-capable chips are being manufactured. Although a lot of knowledge has been acquired, it is still not quite clear whether IEEE 802.11p communication is apt for the challenging dynamic road conditions, and especially for safety-critical applications. Up until recently, most of the research efforts have been addressed towards understanding the technical feasibility of inter-vehicle communication; and only now is the research focus shifting towards investigating whether inter-vehicle communication can successfully support driver assistance applications, and as a consequence, road traffic safety and efficiency.

In the same vein, we stated a general research question in the beginning of this thesis: **Can rear-end collision avoidance and intersection management applications be supported by inter-vehicle communication and result in safe and efficient traffic?** Two concrete applications have been developed in this thesis to answer the stated question: **Rear-End Collision Avoidance (RECA)**, a safety application aiming to avoid rear-end collisions and **Virtual Traffic Lights (VTL)**, an efficiency application aiming

to improve traffic efficiency on intersections that are not governed by conventional traffic lights. Although both applications pursue different goals, both applications are clearly **safety-critical**, which poses a higher load on inter-vehicle communication.

The answer to the general research question is complicated by the two main challenges that lie in the domain of application design and application evaluation. Consequently, we distinguished two detailed questions. The first question is addressing the challenge of application design, namely, *how to design a safety-critical application that is fail-safe against unreliability of vehicular communication and unpredictable driver behavior.* Unreliability of vehicular communication and unpredictable driver behavior are the two most obvious failure sources that can deteriorate the safe operation of applications, and thus chosen to be explicitly addressed in this work. In order to answer the first detailed question, existing related (sensor-based) applications, particularly the functional and safety requirements of applications have been studied. The best practices have been selected and adopted for the design of communication-based RECA and VTL applications. In addition, since driver assistance applications are envisioned to be supported by vehicular communication, required transmission parameters should be identified. The default communication requirements, e.g., update frequency of 10 Hz, are not always justified by applications requirements. Moreover, congestion control mechanisms adapt transmission parameters to keep the channel load under control. If this is done without accounting for application requirements, applications might not be able to fulfill their purposes. We performed a study of application requirements on reception of information which provides an indication on how optimal transmission parameters can be set. In addition, conditions can be identified when information provided by communication can be combined with information obtained via in-vehicle sensors when higher accuracy for application is desired. For the RECA application, strict requirements for **zero false positive and zero false negative errors** have been aimed. Obviously, both zero false positive and zero false negative can never be achieved, but we showed that it is possible to limit a small geographical region, called *tolerance region*, where only false positives (excluding false negatives) can happen, which makes it feasible for IEEE 802.11p communication to support an application. The VTL application does not need to face such strict requirements as traffic light information does not change unexpectedly. Nevertheless, the performed analysis on when information should be received allows a meaningful transmission parameter setting. Contrary to the related work, the designed applications integrate explicit **fail-safety features** to counteract the effect of the two most probable failure sources: **unreliability of communication** and **unpredictability of human drivers**. Fail-safety features exclude false negative errors by making use of a worst case assumption. In particular, if no new communication packets are being received, the fail-safe RECA application assumes the worst case situation change that is possible from the moment the last packet has been received. The fail-safe VTL application, in case no packets are being received, assumes information packet loss and expects the presence of other vehicles. Similarly, a fail-safe application can not solely rely on the driver to take appropriate actions but has to be ready to take over vehicle control in case the driver does not react. Although the worst case assumption might be detrimental

for traffic efficiency, as vehicles might reduce their speeds, it allows vehicles and their drivers to remain safe. The VTL application design as based on challenging distributed coordination was chosen for the **formal verification** procedure. The VTL application requirement to possess information before a certain distance is reached allowed to perform verification, not only for application and communication aspects but also for the "discretized" movement of vehicles. Typically, formal verification is only performed on some of these domains or simplified abstractions are being made. We performed formal verification with the model checking, and although verification was performed for only a small number of vehicles, the verification indicates general safety of the designed protocol. As a result, in the first part of the thesis, RECA and VTL applications have been designed in a fail-safe manner, and, as in the case of VTL application, formally verified to be safe.

The second detailed research question addresses the challenge of application evaluation and is as follows: *how to determine the scalability of IEEE 802.11p communication to reliably support applications and to evaluate the impact of a fail-safe application on the road traffic.* The answer to this question has two facets. First, the determination of IEEE 802.11p communication scalability to reliably support applications requires a definition of "reliability". Within the scope of this thesis, two variations of "reliability" are used for RECA and VTL applications separately. We assume capability of IEEE 802.11p communication to satisfy the RECA application requirements to represent communication reliability. In case of VTL protocol which requires communication between various vehicles for intersection crossing coordination, the probability to detect the presence of other vehicles over communication is assumed to represent reliability. We presented **a method to link application and networking layers**, and thus quantify the capability of communication to satisfy the RECA application requirements. Notably, an analytical method, called **awareness principle**, is introduced to translate the RECA application requirements to the required transmission parameters. For the majority of realistic scenarios, IEEE 802.11p communication supports RECA and VTL applications with high reliability. If the required reliability level cannot be achieved, e.g., due to congested channel caused by a large number of communicating vehicles, the designed fail-safe applications can sacrifice traffic efficiency to achieve a desired level of reliability. The impact an application can have on traffic is evaluated by comparing the resulting traffic efficiency to conditions when no communication-based applications are in place. We showed that the vehicle density achieved with RECA application is comparable to the current highway level of services and correspond to various levels up to the most congested one. Travel time achieved with the VTL application is closely comparable to what can be achieved with conventional traffic lights today. The general evaluation methodology is presented that provides stepwise guidelines to quantify the impact an application might have on road traffic, together with quantifying how IEEE 802.11p communication scales to support the application.

Our approach enables to evaluate reliability and scalability of vehicular communication to support a driver assistance application and to quantify the resulting impact on road traffic, when network and radio conditions are known. In reality though, network and radio conditions can only be estimated or extra efforts need to be in-

vested in order to determine them precisely. As a part of this thesis, **a sensitivity analysis** has been performed that identifies situations when accurate information on network and radio conditions is important for the correct estimation of network performance. In addition, errors in estimation of network performance caused by inaccurate information are quantified. This information can be used by application designers if errors cannot be tolerated, e.g., by integrating fail-safe features or investing extra efforts to determine accurate information.

As has been demonstrated, the general question of whether rear-end collision avoidance and intersection management applications can be supported by inter-vehicle communication and result in safe and efficient traffic can be answered positively. Recalling the concept of dependability, where a single error does not lead to a system failure (e.g., vehicle collision), it is clear that a non-perfect communication reliability does not translate to a vehicle collision.

Before first communication-based driver assistance applications can be commercially available, a lot of work is still to be done. Future research work can address application design and application evaluation aspects alike. For example, several options for application design optimization are conceivable; RECA application can benefit from information fusion of communication and sensors, and VTL protocol can be optimized by allowing several crossing vehicles on the intersection area simultaneously or by including communication between several intersections for dynamic light phase switching. Future work can address the effect of burst errors in communication, inaccurate localization information, as well as the presence of unequipped vehicles. More work can be done for formal verification; e.g., an equivalence check can be used to verify that any further implementation of a verified model is equivalent to the verified model itself. In addition, different level of system abstraction or detail degree can be modeled depending on the verification purposes. Additional fail-safety features, that exclude even a very rare error, can also be integrated, provided that the gained safety is justifiable for the lost efficiency.

The results presented in this thesis provide initial indication regarding the impact that communication-based Rear-End Collision Avoidance and Virtual Traffic Lights applications might have on road traffic and how well IEEE 802.11p communication can support them. Presented methodology can also be used for the design and evaluation of further driver assistance applications. One can also employ different or more realistic models, but the methodology presented in this thesis remains valid. In general, this thesis supports the optimistic forecast for the proliferation of communication-aided vehicles that support not only traffic efficiency but also safety-critical applications.

Bibliography

[A. 13] A. Skabardonis. Advanced traffic signal control algorithms. Technical Report UCB-ITS-PRR-2013-XX-A, State of California. Department of Transportation. California PATH, September 2013.

[AAE+14] Daimler AG, Volkswagen AG, Institute EURECOM, Renault SAS, Hitachi Europe Ltd., and Marben Products. Intelligent Transport Systems (ITS); Vehicular communications; Basic set of applications; Part 2: Specification of cooperative awareness basic service. Technical Report ETSI EN 302 637–2 v1.3.2 (2014–11), The European Telecommunications Standards Institute (ETSI), November 2014.

[AATP+13] M. Ahmane, A. Abbas-Turki, F. Perronnet, J. Wu, A. El Moudni, J. Buisson, and R. Zeo. Modeling and controlling an isolated urban intersection based on cooperative vehicles. *Transportation Research Part C: Emerging Technologies*, 28(0):44–62, 2013.

[ADM+12] R.W.G. Anderson, S.D. Doecke, J.R.R. Mackenzie, G. Ponte, D. Paine, and M. Paine. Potential benefits of forward collision avoidance technology. Technical Report CASR0106, Centre For Automotive Safety Research, The University of Adelaide, Australia, April 2012.

[AFV+12] T.-C. Au, C.-L. Fok, S. Vishwanath, C. Julien, and P. Stone. Evasion planning for autonomous vehicles at intersections. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1541–1546, October 2012.

[AGH11] N. An, T. Gaugel, and H. Hartenstein. VANET: Is 95% probability of packet reception safe? In *Proceedings of the 11th International Conference on ITS Telecommunications (ITST)*, pages 113–119, August 2011.

[AKG06] O. Ararat, E. Kural, and B.A. Guvenc. Development of a collision warning system for adaptive cruise control vehicles using a comparison analysis of recent algorithms. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 194–199, June 2006.

[ALR00] A. Avižienis, J.-C. Laprie, and B. Randell. Fundamental concepts of dependability. In *Proceedings of the 3rd IEEE Information Survivability Workshop (ISW)*, pages 7–12, October 2000.

[AMB+12] M. Asplund, A. Manzoor, M. Bouroche, S. Clarke, and V. Cahill. *A formal approach to autonomous vehicle coordination*, volume 7436 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012.

[AMH14] N. An, J. Mittag, and H. Hartenstein. Designing fail-safe and traffic efficient 802.11p-based rear-end collision avoidance. In *Proceedings of the IEEE Vehicular Networking Conference (VNC)*, pages 9–16, Paderborn, Germany, December 2014.

[AMH15] N. An, J. Mittag, and H. Hartenstein. Designing fail-safe and traffic efficient 802.11p-based rear-end collision avoidance. *Elsevier Journal on Ad Hoc Networks*, 2015.

[AMJ+13] N. An, M. Maile, D. Jiang, J. Mittag, and H. Hartenstein. Balancing the requirements for a zero false positive/negative forward collision warnings. In *Proceedings of the 10th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS)*, pages 191–195, Banff, Canada, March 2013.

[AMSETM11] N. An, J. Mittag, F. Schmidt-Eisenlohr, and M. Torrent-Moreno. Accurate knowledge of radio channel and network conditions – When does it matter? In *Proceedings of the 8th International Conference on Wireless On-Demand Network Systems and Services (WONS)*, pages 109–116, January 2011.

[AR04] G. Abe and J. Richardson. The human factors of collision warning systems: System performance, alarm timing, and driver trust. In *Proceedings of the 48th Annual Meeting of Human Factors and Ergonomics Society*, pages 2232–2236, September 2004.

[ARM09] N. An, J. Riihijärvi, and P. Mähönen. Studying the delay performance of opportunistic communication in VANETs with realistic mobility models. In *Proceedings of the 69th IEEE Semiannual Vehicular Technology Conference (VTC-Spring)*, pages 1–5, April 2009.

[AST10] ASTM International. Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems – 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications. American Society for Testing and Materials, 2010.

[Ati10] M. Atif. *Development of a framework for in-vehicle rear-end collision warning system considering driver characteristics*. PhD thesis, Ryerson University, Toronto, Ontario, Canada, 2010.

[AYHI96] H. Araki, K. Yamada, Y. Hiroshima, and T. Ito. Development of rear-end collision avoidance system. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 224–229, September 1996.

[BA08] M. Ben-Ari. *Principles of the SPIN model checker*. Springer-Verlag London Limited, 1st edition, 2008.

[Bat07] Battelle. Final report. Evaluation of the Volvo intelligent vehicle initiative field operational test. Version 1.3. Technical report, U.S. Department of Transportation, Federal Highway Administration, January 2007.

[BBF00] M. Bertozzi, A. Broggi, and A. Fascioli. Vision-based intelligent vehicles: State of the art and perspectives. *Robotics and Autonomous Systems*, 32(1):1–16, 2000.

[BC98] P. Barber and N. Clarke. Advanced collision warning systems. In *Proceedings of the IEE Colloquium on Industrial Automation and Control: Applications in the Automotive Industry*, pages 2/1–2/9, April 1998.

[BCM+98] A.L. Burgett, A. Carter, R.J. Miller, W.G. Najm, and D.L. Smith. A collision warning algorithm for rear-end collisions. In *Proceedings of the 16th International Conference on the Enhanced Safety of Vehicles (ESV)*, Windsor, Canada, June 1998.

[BF14] P. Bourque and R.E. Fairley. *Guide to the software engineering body of knowledge*. IEEE Computer Society, 3rd edition, 2014.

[BHL+13] K. Basak, S.N. Hetu, Z. Li, L.C. Azevedo, H. Loganathan, T. Toledo, R. Xu, Y. Xu, L.-S. Peh, and M. Ben-Akiva. Modeling reaction time within a traffic simulation model. In *Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 302–309, October 2013.

[BK06] F. Bai and H. Krishnan. Reliability analysis of DSRC wireless communication for vehicle safety applications. In *Proceedings of the IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 355–362, September 2006.

[BK08] C. Baier and J.-P. Katoen. *Principles of model checking*. The MIT Press, 2008.

[BKPP02] S.J. Brunson, E.M. Kyle, N.C. Phamdo, and G.R. Preziotti. Alert algorithm development program. NHTSA rear-end collision alert algorithm. Final report. Technical Report DOT HS 809 526, U.S. Department of Transportation, National Highway Traffic Safety Administration, September 2002.

[BKR13] G. Bansal, J.B. Kenney, and C.E. Rohrs. LIMERIC: A linear adaptive message rate algorithm for DSRC congestion control. *IEEE Transactions on Vehicular Technology*, 62(9):4182–4197, November 2013.

[BLM01] T.L. Brown, J.D. Lee, and D.V. McGehee. Human performance models and rear-end collision avoidance algorithms. *Human Factors*, 43(3):462–482, 2001.

[BM00] F. Browand and M. Michaelian. Platoon travel saves fuel – How much? *Intellimotion*, 9(2):1–5, 2000.

[BMCR07] J. Brown, M. McCallum, J. Campbell, and C. Richard. Integrated vehicle-based safety system objective test scenario warning strategies: Kinematic analyses and DVI outputs. Technical report, University of Michigan Transportation Research Institute (UMTRI), May 2007.

[BMY05] R.L. Bertini, C.M. Monsere, and T. Yin. Benefits of intelligent transportation systems technologies in urban areas: A literature review. Final report. Technical report, Portland State University, Department of Civil and Environmental Engineering, School of Urban Studies and Planning, Center for Transportation Studies (CTS), April 2005.

[BR11] F. Bella and R. Russo. A collision warning system for rear-end collision: A driving simulator study. *Procedia - Social and Behavioral Sciences*, 20(0):676–686, 2011.

[Bro07] G.M. Brooker. Mutual interference of millimeter-wave radar systems. *IEEE Transactions on Electromagnetic Compatibility*, 49(1):170–181, February 2007.

[BSK10] F. Bai, D. Stancil, and H. Krishnan. Toward understanding characteristics of Dedicated Short Range Communications (DSRC) from a perspective of vehicular network engineers. In *Proceedings of the 16th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 329–340, New York, NY, USA, September 2010.

[BTD06] S. Biswas, R. Tatchikou, and F. Dion. Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety. *IEEE Communications Magazine*, 44(1):74–82, January 2006.

[CC09] T.-H. Chang and C.-J. Chou. Rear-end collision warning system on account of a rear-end monitoring camera. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 913–917, June 2009.

[Cen08] Joint Transport Research Centre. Towards Zero: Ambitious road safety targets and the Safe System approach. Summary document. Technical report, OECD–ITF, October 2008.

[CFKV15] D. Crolla, D.E. Foster, T. Kobayashi, and N. Vaughan. *Encyclopedia of automotive engineering*. John Wiley & Sons, Inc. Publication, Hoboken, New Jersey, 2015.

[CG78] R.J. Caudill and W.L. Garrard. Vehicle-follower longitudinal control for automated transit vehicles. *Journal of Dynamic Systems, Measurement, and Control*, 99(4):241–248, 1978.

[CGM12] A. Cabrera, S. Gowal, and A. Martinoli. A new collision warning system for lead vehicles in rear-end collisions. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 674–679, June 2012.

[CHCP08] T. Choe, J.W. Hur, J.S. Chae, and Y.-W. Park. Real-time collision avoidance method for unmanned ground vehicle. In *Proceedings of the International Conference on Control, Automation and Systems (ICCAS)*, pages 843–846, October 2008.

[COM96] COMSIS Corporation. Preliminary human factors guidelines for crash avoidance warning devices. Technical Report DTNH22-91-C-07004, U.S. Department of Transportation, National Highway Traffic Safety Administration, January 1996.

[Com99] Federal Communications Commission. Amendment of parts 2 and 90 of the commission's rules to allocate the 5.850-5.925 GHz band to the mobile service for dedicated short-range communications of intelligent transportation systems. Technical Report 99–305, FCC, October 1999.

[Com09a] Road Safety Commission. Towards Zero – Road safety strategy to reduce road trauma in Western Australia 2008–2020, 2009.

[Com09b] The European Commission. Standardization Mandate addressed to CEN, CENELEC and ETSI in the field of information and communication technologies to support the interoperability of

co-operative systems for intelligent transport in the European community. Technical Report DG ENTR/D4 M/453 EN, October 2009.

[Com12] The European Commission. Commission regulation (EU) No 347/2012 of 16 April 2012 implementing regulation (EC) No 661/2009 of the European Parliament and of the Council with respect to type-approval requirements for certain categories of motor vehicles with regard to advanced emergency braking systems. *Official Journal of the European Union*, pages I.109/1–I.109/17, April 2012.

[DBG+10] K. Dar, M. Bakhouya, J. Gaber, M. Wack, and P. Lorenz. Wireless communication technologies for ITS applications [Topics in Automotive Networking]. *IEEE Communications Magazine*, 48(5):156–162, May 2010.

[dBKvKN04] D. de Bruin, J. Kroon, R. van Klaveren, and M. Nelisse. Design and test of a cooperative adaptive cruise control system. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 392–396, June 2004.

[DBN+94] A. Doi, T. Butsuen, T. Niibe, T. Takagi, Y. Yamamoto, and H. Seni. Development of a rear-end collision avoidance system with automatic brake control. *JSAE Review*, 15(4):335–340, 1994.

[DDC14] Z. Dimitrova, S. Dermendzhieva, and K. Chakarova. *Thematic research summary – Security and safety*. European Commission, 2014.

[Dep14] Department of Transportation and National Highway Traffic Safety Administration. Federal Motor Vehicle Safety Standards: Vehicle-to-Vehicle (V2V) Communications. *Federal Register*, 79(161), August 2014.

[dLF10] A. de La Fortelle. Analysis of reservation algorithms for cooperative planning at intersections. In *Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 445–449, September 2010.

[dMB08] L. de Moura and N. Bjørner. Z3: An efficient SMT solver. In C.R. Ramakrishnan and J. Rehof, editors, *Tools and algorithms for the construction and analysis of systems*, volume 4963 of *Lecture Notes in Computer Science*, pages 337–340. Springer Berlin Heidelberg, 2008.

[DMSS04] E. Dagan, O. Mano, G.P. Stein, and A. Shashua. Forward collision warning with a single camera. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 37–42, June 2004.

[DRI] The project DRIVE C2X. `http://www.drive-c2x.eu/project`. Accessed: 2015-10-10.

[DS04] K. Dresner and P. Stone. Multiagent traffic management: A reservation-based intersection control mechanism. In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 530–537, July 2004.

[DS05] K. Dresner and P. Stone. Multiagent traffic management: An improved intersection control cechanism. In *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 471–477, New York, USA, 2005.

[DS07] K. Dresner and P. Stone. Sharing the road: Autonomous vehicles meet human drivers. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1263–68, January 2007.

[DS08] K. Dresner and P. Stone. Mitigating catastrophic failure at intersections of autonomous vehicles. In *Proceedings of the AAMAS Workshop on Agents in Traffic and Transportation*, pages 78–85, Estoril, Portugal, May 2008.

[DZ12] L. Delgrossi and T. Zhang. *Vehicle safety communications – Protocols, security, and privacy*. Wiley series on information and communication technology. Wiley, 2012.

[EGH+06] T. ElBatt, S.K. Goel, G. Holland, H. Krishnan, and J. Parikh. Cooperative collision warning using dedicated short range wireless communications. In *Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks (VANET)*, pages 1–9, New York, NY, USA, September 2006.

[E.H10] E.H. Choi. Crash factors in intersection-related crashes: An on-scene perspective. Technical Report DOT HS 811 366, U.S. Department of Transportation, National Highway Traffic Safety Administration, September 2010.

[F. 11a] F. Ahmed-Zaid et al. Vehicle Safety Communications – Applications (VSC-A) Final report. Technical Report DOT HS 811 492A, U.S. Department of Transportation, National Highway Traffic Safety Administration, September 2011.

[F. 11b] F. Ahmed-Zaid et al. Vehicle Safety Communications – Applications (VSC-A) Final report: Appendix volume 1 System design and objective Test . Technical Report DOT HS 811 492B, U.S. Department of Transportation, National Highway Traffic Safety Administration, September 2011.

[FFC+10] M. Ferreira, R. Fernandes, H. Conceiçao, W. Viriyasitavat, and O.K. Tonguz. Self-organized traffic control. In *Proceedings of the 7th ACM International Workshop on VehiculAr InterNETworking (VANET)*, pages 85–90, New York, NY, USA, September 2010. ACM.

[FHSK10] Y.P. Fallah, C.-L. Huang, R. Sengupta, and H. Krishnan. Congestion control based on channel occupancy in vehicular broadcast networks. In *Proceedings of the 72nd IEEE Semiannual Vehicular Technology Conference Fall (VTC-Fall)*, pages 1–5, September 2010.

[FHSK11] Y.P. Fallah, C.-L. Huang, R. Sengupta, and H. Krishnan. Analysis of information dissemination in vehicular ad-hoc networks with application to cooperative vehicle safety systems. *IEEE Transactions on Vehicular Technology*, 60(1):233–247, January 2011.

[GBM10] M. Goppelt, H.-L. Blöcher, and W. Menzel. Automotive radar – investigation of mutual interference mechanisms. *Advances in Radio Science*, 8:55–60, 2010.

[GD97] H.P. Groll and J. Detlefsen. History of automotive anticollision radars and final experimental results of a mm-Wave car radar developed by the Technical University of Munich. *IEEE Aerospace and Electronic Systems Magazine*, 12(8):15–19, August 1997.

[GdSMH01] A.R. Girard, J.B. de Sousa, J.A. Misener, and J.K. Hedrick. A control architecture for integrated cooperative cruise control and collision warning systems. In *Proceedings of the 40th IEEE Conference on Decision and Control*, volume 2, pages 1491–1496 vol.2, December 2001.

[Gen05] General Motors Corporation. Automotive Collision Avoidance Systems Field Operational Test (ACAS FOT). Final program report. Technical Report DOT HS 809 886, U.S. Department of Transportation, National Highway Traffic Safety Administration, March 2005.

[GGD+07] V. Gradinescu, C. Gorgorin, R. Diaconescu, V. Cristea, and L. Iftode. Adaptive traffic lights using car-to-car communication. In *Proceedings of the 65th IEEE Semiannual Vehicular Technology Conference (VTC-Spring)*, pages 21–25, April 2007.

[GHMK14] M. Gholibeigi, G. Heijenk, D. Moltchanov, and Y. Koucheryavy. Analysis of a receiver-based reliable broadcast approach for vehicular networks. In *Proceedings of the IEEE Vehicular Networking Conference (VNC)*, pages 89–96, December 2014.

[GNK05] R. Groenevelt, P. Nain, and G. Koole. The message delay in mobile ad hoc networks. *Performance Evaluation*, 62(1–4):210–228, October 2005.

[GPSV06] O. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen. Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations. *Vehicle System Dynamics*, 44(7):569–590, 2006.

[Gre00] M. Green. How long does it take to stop? methodological analysis of driver perception-brake times. *Transportation Human Factors*, 2(3):195–216, 2000.

[GS07] J. Gozalvez and M. Sepulcre. Opportunistic-driven adaptive radio resource management technique for efficient wireless vehicular communications. In *Proceedings of the 66th IEEE Semiannual Vehicular Technology Conference (VTC-Fall)*, pages 2116–2120, September 2007.

[GS15] N. Guenther and H. Salow. Collision avoidance and operator guidence - innovating mine vehicle safety. In *Proceedings of the Engineers Australia (MEMMS)*, 2015.

[GSKB07] X. Guan, R. Sengupta, H. Krishnan, and F. Bai. A feedback-based power control algorithm design for VANET. In *Proceedings of the Mobile Networking for Vehicular Environments*, pages 67–72, May 2007.

[H. 13] H. Schittenhelm. Advanced brake assist: Real world effectiveness of current implementations and next generation enlargements by Mercedes-Benz. In *Proceedings of the 23rd International Conference on the Enhanced Safety of Vehicles (ESV)*, May 2013.

[Hay71] J.C. Hayward. *Near misses as a measure of safety at urban intersections*. PhD thesis, Pennsylvania State University, 1971.

[HCCV13] M.R. Hafner, D. Cunningham, L. Caminiti, and D. Del Vecchio. Cooperative collision avoidance at intersections: Algorithms and experiments. *Intelligent Transportation Systems, IEEE Transactions on*, 14(3):1162–1175, September 2013.

[HFSK10] C.-L. Huang, Y.P. Fallah, R. Sengupta, and H. Krishnan. Adaptive intervehicle communication control for cooperative safety systems. *IEEE Network*, 24(1):6–13, January 2010.

[HH10] J. Haas and Y.-C. Hu. Communication requirements for crash avoidance. In *Proceedings of the 7th ACM International Workshop on VehiculAr InterNETworking (VANET)*, pages 1–10, New York, NY, USA, September 2010.

[His95] M. Hischke. Collision warning radar interference. In *Proceedings of the Intelligent Vehicles Symposium*, pages 13–18, September 1995.

[HL10] H. Hartenstein and K. Laberteaux. *VANET – Vehicular applications and inter-networking technologies*. Intelligent Transport Systems. John Wiley & Sons, December 2010.

[HLA+12] J. Hérard, M. Lesemann, A. Aparicio, H. Eriksson, and J. Jacobson. A comparative study – rear-end collision avoidance. *Procedia – Social and Behavioral Sciences*, 48(0):173–183, 2012.

[HMNS91] J.K. Hedrick, D. McMahon, V. Narendran, and D. Swaroop. Longitudinal vehicle controller design for IVHS systems. In *Proceedings of the American Control Conference*, pages 3107–3112, June 1991.

[Hol91] G.J. Holzmann. *Design and validation of computer protocols*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1991.

[Hol03] G.J. Holzmann. *SPIN model checker: The primer and reference manual*. Addison-Wesley Professional, first edition, 2003.

[HPY+14] J. Harding, G. Powell, R. Yoon, J. Fikentscher, C. Doyle, D. Sade, M. Lukuc, J. Simons, and J. Wang. Vehicle-to-vehicle communications: Readiness of V2V technology for application. Technical Report DOT HS 812 014, U.S. Department of Transportation, National Highway Traffic Safety Administration, August 2014.

[HRW08] A. Hoare, D.G. Regan, and D.P. Wilson. Sampling and sensitivity analyses tools (sasat) for computational modelling. *Theoretical Biology and Medical Modelling*, 5(4), 2008.

[HTB08] M. Husain, T. Tiernan, and D. Bezzina. Integrated vehicle-based safety systems light vehicle verification test plan. Technical report, University of Michigan Transportation Research Institute (UMTRI), March 2008.

[Hug02] W. Hugemann. Driver reaction times in road traffic. In *Proceedings of the European Association for Accident Research and Analysis, Annual Convention*, September 2002.

[HXRK09] K. Hong, D. Xing, V. Rai, and J. Kenney. Characterization of DSRC performance as a function of transmit power. In *Proceedings of the 6th ACM International Workshop on VehiculAr InterNETworking (VANET)*, pages 63–68, New York, NY, USA, 2009.

[IEE12] IEEE Computer Society. 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. Standard, 2012. IEEE Std 802.11™-2012.

[IEE14] IEEE Vehicular Technology Society. IEEE Guide for Wireless Access in Vehicular Environments (WAVE) – Architecture. Technical report, March 2014. IEEE Std 1609.0™-2014.

[IHN95] T. Ito, Y. Hiroshima, and K. Nishioka. Fuzzy reasoning method with learning function for rear-end collision avoidance system. In *Proceedings of the World Congress on Intelligent Transport Systems*, pages 1175–1180, 1995.

[Ins09] The European Telecommunications Standards Institute. Intelligent Transport Systems (ITS); Vehicular communications; Basic set of applications; Definitions. Technical Report ETSI TR 102 638 v1.1.1 (2009–06), ETSI, June 2009.

[Ins11] The European Telecommunications Standards Institute. Intelligent Transport Systems (ITS); Decentralized congestion control mechanisms for intelligent transport systems operating in the 5 GHz range; Access layer part. Technical Report ETSI TS 102 687 V1.1.1 (2011-07), ETSI, July 2011.

[Int07] International Transport Forum. Congestion: A global challenge. The extent of and outlook for congestion in inland, maritime and air transport. Reference document for Session 1. Technical Report CEMTITF(2007)6, European Conference of Ministries of Transport., May 2007.

[Int12] International Transport Forum. Transport outlook 2012: Seamless transport for greener growth. Technical report, OECD/ITF, 2012.

[ISO09] ISO standard. ISO 22179: Intelligent transport systems – Full Speed Range Adaptive cruise control (FSRA) systems – Performance requirements and test procedures. International Organization for Standardization, 2009.

[ISO10] ISO standard. ISO 15622: Intelligent transport systems – Adaptive Cruise Control systems – Performance requirements and test procedures. International Organization for Standardization, 2010.

[ISO11] ISO standard. ISO 26262: Road vehicles – functional safety. International Organization for Standardization, 2011.

[ISO13a] ISO standard. ISO 15623: Intelligent transport systems – Forward vehicle collision warning systems – Performance requirements and test procedures. International Organization for Standardization, 2013.

[ISO13b] ISO standard. ISO 22839: Intelligent transport systems – Forward vehicle collision mitigation systems – Operation, performance, and verification requirements. International Organization for Standardization, 2013.

[Jan05] J. Jansson. *Collision avoidance theory with application to automotive collision mitigation*. PhD thesis, Linköping University, Sweden, 2005.

[JBS+14] S. Joerer, B. Bloessl, M. Segata, C. Sommer, R. Lo Cigno, and F. Dressler. Fairness kills safety: A comparative study for intersection assistance applications. In *Proceedings of the 25th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1442–1447, Washington, D.C., September 2014.

[JCD08] D. Jiang, Q. Chen, and L. Delgrossi. Optimal data rate selection for vehicle safety communications. In *Proceedings of the 5th ACM International Workshop on VehiculAr Inter-NETworking (VANET)*, pages 30–38, New York, NY, USA, 2008.

[J.L07] J.L. Campbell and C.M. Richard and J.L. Brown and M. McCallum. Crash warning system interfaces: Human factors insights and lessons learned. Final report. Technical Report DOT HS 810 697, U.S. Department of Transportation, National Highway Traffic Safety Administration, January 2007.

[JLC08] A.H. Jamson, F.C.H. Lai, and O.M.J. Carsten. Potential benefits of an adaptive forward collision warning system. *Transportation Research Part C: Emerging Technologies*, 16(4):471–484, 2008.

[JR71] G. Johansson and K. Rumar. Driver's brake reaction times. *Human Factors*, 13(1):23–27, 1971.

[JTWL14] P. Ji, H.-M. Tsai, C. Wang, and F. Liu. Vehicular visible light communications with LED taillight and rolling shutter camera. In *Proceedings of the 79th IEEE Semiannual Vehicular Technology Conference (VTC-Spring)*, pages 1–6, May 2014.

[KCF+03] R.J. Kiefer, M.T. Cassar, C.A. Flannagan, D. LeBlanc, M.D. Palmer, R. Deering, and M. Shulman. Forward collision warning requirements project. Task 1. Final report. Technical Report DOT HS 809 574, U.S. Department of Transportation, National Highway Traffic Safety Administration, January 2003.

[KCF+05] R.J. Kiefer, M.T. Cassar, C.A. Flannagan, C.J. Jerome, and M. Palmer. Forward collision warning requirements project. Task 2 and 3a. Final report. Technical Report DOT HS 809 902, U.S. Department of Transportation, National Highway Traffic Safety Administration, August 2005.

[Ker04] B.S. Kerner. *The physics of traffic. Empirical freeway pattern features, engineering applications, and theory*. Springer-Verlag Berlin Heidelberg, 1st edition, 2004.

[KG11] K.D. Kusano and H.C. Gabler. Method for estimating time to collision at braking in real-world, lead vehicle stopped rear-end crashes for use in pre-crash system design. *SAE International Journal of Passenger Cars – Mechanical Systems*, 4(1):435–443, 2011.

[KG12] K.D. Kusano and H.C. Gabler. Safety benefits of forward collision warning, brake assist, and autonomous braking systems in rear-end collisions. *IEEE Transactions on Intelligent Transportation Systems*, 13(4):1546–1555, December 2012.

[KGRK11] S. Kaul, M. Gruteser, V. Rai, and J. Kenney. Minimizing age of information in vehicular networks. In *Proceedings of the 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pages 350–358, June 2011.

[KH09] M. Killat and H. Hartenstein. An empirical model for probability of packet reception in vehicular ad hoc networks. *EURASIP J. Wirel. Commun. Netw.*, 2009:4:1–4:12, January 2009.

[KLF05] R.J. Kiefer, D.J. LeBlanc, and C.A. Flannagan. Developing an inverse time-to-collision crash alert timing approach based on

drivers' last-second braking and steering judgments. *Accident Analysis & Prevention*, 37(2):295–303, 2005.

[KLP+99] R. Kiefer, D. LeBlanc, M. Palmer, J. Salinger, R. Deering, and M. Shulman. Development and validation of functional definitions and evaluation procedures for collision warning/avoidance systems. Technical Report DOT HS 808 964, U.S. Department of Transportation, National Highway Traffic Safety Administration, August 1999.

[KLTY08] L. Kun, Y. Luo, G. Tei, and S. Yang. Attention recognition of drivers based on head pose estimation. In *Proceedings of the IEEE Vehicle Power and Propulsion Conference (VPPC)*, pages 1–5, September 2008.

[KMH+93] R.R. Knipling, M. Mironer, D.L. Hendricks, L. Tijeripa, J. Everson, J.C. Allen, and C. Wilson. Assessment of IVHS countermeasures for collision avoidance: Rear-end crashes. Technical Report DOT HS 807 995, U.S. Department of Transportation, National Highway Traffic Safety Administration, May 1993.

[KOUK97] K. Kodaka, M. Otabe, Y. Urai, and H. Koike. Rear-end collision avoidance assist system. Technical report, Honda R&D Co., Ltd, 1997.

[KSdPM12] B. Kloiber, T. Strang, and F. de Ponte Muller. Slipstream cooperative adaptive cruise control – A conceptual ITS application for electric vehicles. In *Proceedings of the IEEE International Electric Vehicle Conference (IEVC)*, pages 1–5, March 2012.

[LBT+08] D. LeBlanc, D. Bezzina, T. Tiernan, K. Freeman, M. Gabel, and D. Pomerleau. System performance guidelines for a prototype Integrated Vehicle-Based Safety System (IVBSS) – Light vehicle platform. Technical report, University of Michigan Transportation Research Institute (UMTRI), March 2008.

[Lev95] N.G. Leveson. *Safeware: System safety and computers*. ACM, New York, NY, USA, 1995.

[Lev00] N.G. Leveson. System safety in computer-controlled automotive systems. Technical Report 2000-01-1048, SAE Technical Paper, 2000.

[LHH04] J.D. Lee, J.D. Hoffman, and E. Hayes. Collision warning design to mitigate driver distraction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 65–72, New York, NY, USA, 2004.

[LLW+14] G. Lu, L. Li, Y. Wang, R. Zhang, Z. Bao, and H. Chen. A rule based control algorithm of connected vehicles in uncontrolled intersection. In *Proceedings of the 17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 115–120, October 2014.

[LMBR02a] J. Lee, D. McGehee, T. Brown, and M. Reyes. Driver distraction, warning algorithm parameters, and driver response to imminent rear-end collisions in a high-fidelity driving simulator. Technical Report DOT HS 809 448, U.S. Department of Transportation, National Highway Traffic Safety Administration, Human Factors and Vehicle Safety Research Program University of Iowa Public Policy Center, March 2002.

[LMBR02b] J.D. Lee, D.V. McGehee, T.L. Brown, and M.L. Reyes. Collision warning timing, driver distraction, and driver response to imminent rear-end collisions in a high-fidelity driving simulator. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 44(2):314–334, 2002.

[LMZW08] C. Li, H. Meng, H. Zhang, and X. Wang. Evaluation and improvement of required deceleration algorithm in frontal collision warning systems. In *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1038–1042, October 2008.

[LP05] K. Lee and H. Peng. Evaluation of automotive forward collision warning and collision avoidance algorithms. *Vehicle System Dynamics*, 43(10):735–751, October 2005.

[LP11a] S.M. Loos and A. Platzer. Safe intersections: At the crossing of hybrid systems and verification. In *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1181–1186, October 2011.

[LP11b] H. Lu and C. Poellabauer. Analysis of application-specific broadcast reliability for vehicle safety communications. In *Proceedings of the 8th ACM International Workshop on Vehicular Internetworking (VANET)*, pages 67–72, New York, NY, USA, 2011.

[LP12] J. Lee and B. Park. Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment. *Trans. Intell. Transport. Sys.*, 13(1):81–90, March 2012.

[LT06] M. Lindman and E. Tivesten. A method for estimating the benefit of autonomous braking systems using traffic accident data.

Technical Report 2006-01-0473, SAE Technical Paper, January 2006.

[LvEW+11] C. Lei, E.M. van Eenennaam, W.K. Wolterink, G. Karagiannis, G. Heijenk, and J. Ploeg. Impact of packet loss on CACC string stability performance. In *Proceedings of the 11th International Conference on ITS Telecommunications (ITST)*, pages 381–386, Saint Petersburg, Russia, August 2011.

[LW06] L. Li and F.-Y. Wang. Cooperative driving at blind crossings using intervehicle communication. *IEEE Transactions on Vehicular Technology*, 55(6):1712–1724, November 2006.

[MBML11] N. Maslekar, M. Boussedjra, J. Mouzna, and H. Labiod. VANET based adaptive traffic signal control. In *Proceedings of the 73rd IEEE Semiannual Vehicular Technology Conference (VTC-Spring)*, pages 1–5, May 2011.

[Mei98] H.H. Meinel. Automotive millimeterwave radar history and present status. In *Proceedings of the 28th European Microwave Conference*, volume 1, pages 619–629, October 1998.

[MHD08] S.B. McLaughlin, J.M. Hankey, and T.A. Dingus. A method for evaluating collision avoidance systems using naturalistic driving data. *Accident Analysis & Prevention*, 40(1):8–16, 2008.

[MHDK09] S.B. McLaughlin, J.M. Hankey, T.A. Dingus, and S.G. Klauer. Development of an FCW algorithm evaluation methodology with evaluation of three alert algorithms. Final report. Technical Report DOT HS 811 145, U.S. Department of Transportation, National Highway Traffic Safety Administration, June 2009.

[Mil92] PATH Project Milestone. 4-car platoon demos a significant step for IVHS. *Intellimotion*, 2(1):1–2, 1992.

[MKH11] T. Mangel, O. Klemp, and H. Hartenstein. 5.9 GHz inter-vehicle communication at intersections: A validated non-line-of-sight path-loss and fading model. *EURASIP J. Wirel. Commun. Netw.*, 2011, November 2011.

[MMY09] S. Moon, I. Moon, and K. Yi. Design, tuning, and evaluation of a full-range adaptive cruise control system with collision avoidance. *Control Engineering Practice*, 17(4):442–455, 2009.

[MPGO12] V. Milanés, J. Pérez, J. Godoy, and E. Onieva. A fuzzy aid rear-end collision warning/avoidance system. *Expert Systems with Applications*, 39(10):9097–9107, 2012.

[MSEK+08] J. Mittag, F. Schmidt-Eisenlohr, M. Killat, J. Härri, and H. Hartenstein. Analysis and design of effective and low-overhead transmission power control for VANETs. In *Proceedings of the 5th ACM International Workshop on VehiculAr Inter-NETworking (VANET)*, pages 39–48, New York, NY, USA, 2008.

[MSKH11] T. Mangel, F. Schweizer, T. Kosch, and H. Hartenstein. Vehicular safety communication at intersections: Buildings, non-line-of-sight and representative scenarios. In *Proceedings of the 8th IEEE International Conference on Wireless On-Demand Network Systems and Services (WONS)*, pages 35–41, January 2011.

[MSS+14] V. Milanés, S.E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, and M. Nakamura. Cooperative adaptive cruise control in real traffic situations. *IEEE Transactions on Intelligent Transportation Systems*, 15(1):296–305, February 2014.

[NAH13] T. Neudecker, N. An, and H. Hartenstein. Verification and evaluation of fail-safe virtual traffic light applications. In *Proceedings of the IEEE Vehicular Networking Conference (VNC)*, pages 158–165, December 2013.

[NAT+12] T. Neudecker, N. An, O.K. Tonguz, T. Gaugel, and J. Mittag. Feasibility of virtual traffic lights in non-line-of-sight environments. In *Proceedings of the Ninth ACM International Workshop on Vehicular Inter-networking, Systems, and Applications (VANET)*, pages 103–106, New York, NY, USA, 2012.

[NDM+09] S. Nedevschi, R. Danescu, T. Marita, F. Oniga, C. Pocol, S. Bota, M.M. Meinecke, and M.A. Obojski. Stereovision-based sensor for intersection assistance. In G. Meyer, J. Valldorf, and W. Gessner, editors, *Advanced microsystems for automotive applications*, VDI-Buch, pages 129–163. Springer Berlin Heidelberg, 2009.

[Nek09] M. Nekovee. Quantifying performance requirements of vehicle-to-vehicle communication protocols for rear-end collision avoidance. In *Proceedings of the 69th IEEE Semiannual Vehicular Technology Conference (VTC-Spring)*, pages 1–5, April 2009.

[Neu13] T. Neudecker. Requirements of a fail-safe virtual traffic lights application. Diploma thesis, Karlsruhe Institute of Technology, Germany, 2013.

[NLS+11] E. Nodine, A. Lam, S. Stevens, M. Razo, and W. Najm. Integrated Vehicle-Based Safety Systems (IVBSS). Light vehicle field operational test. Independent evaluation. Technical Report DOT HS 811 516, U.S. Department of Transportation, National Highway Traffic Safety Administration, October 2011.

[NPN11] M. Nekoui and H. Pishro-Nik. Analytical design of inter-vehicular communications for collision avoidance. In *Proceedings of the IEEE Semiannual Vehicular Technology Conference (VTC-Fall)*, pages 1–5, September 2011.

[NRTT97] R. Naumann, R. Rasche, J. Tacken, and C. Tahedi. Validation and simulation of a decentralized intersection collision avoidance algorithm. In *Proceedings of the IEEE Conference on Intelligent Transportation System (ITSC)*, pages 818–823, November 1997.

[NSH+06] W.G. Najm, M.D. Stearns, H. Howarth, J. Koopmann, and J. Hitz. Evaluation of an automotive rear-end collision avoidance system. Technical Report DOT HS 810 569, U.S. Department of Transportation, National Highway Traffic Safety Administration, April 2006.

[NVT13] M. Nakamurakare, W. Viriyasitavat, and O.K. Tonguz. A prototype of virtual traffic lights on android-based smartphones. In *Proceeding of the 10th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pages 236–238, June 2013.

[O. 09] O. Grembek and K. Zhou and W.-B. Zhang. Red-light-running collision avoidance. Technical Report UCB-ITS-PRR-2009-15, California PATH Program. Institute of Transportation studies. University of California, Berkley, 2009.

[oEE11] Institute of Electrical and Electronics Engineers. IEEE Guide – Adoption of the Project Management Institute (PMI(R)) Standard. A Guide to the Project Management Body of Knowledge (PMBOK(R) Guide)–Fourth Edition. *IEEE Std 1490-2011*, pages 1–508, November 2011.

[OHS00] B.-K. Oh, W.-K. Hwang, and B.-S. Song. Simulator for forward collision warning and avoidance system. Technical Report 2000-05-0364, SAE Technical Paper, 2000.

[OMFM99] K. Osugi, K. Miyauchi, N. Furui, and H. Miyakoshi. Development of the scanning laser radar for ACC system. *JSAE Review*, 20(4):549–554, 1999.

[OMV+12] E. Onieva, V. Milanés, J. Villagrá, J. Pérez, and J. Godoy. Genetic optimization of a vehicle fuzzy decision system for intersections. *Expert Systems with Applications*, 39(18):13148 – 13157, 2012.

[Per99] C. Perrow. *Normal accidents: Living with high-risk technologies*. Princeton University Press, New Jersey, USA, 1999.

[PQ08] A. Platzer and J.-D. Quesel. KeYmaera: A hybrid theorem prover for hybrid systems (System description). In A. Armando, P. Baumgartner, and G. Dowek, editors, *Automated reasoning*, volume 5195 of *Lecture Notes in Computer Science*, pages 171–178. Springer Berlin Heidelberg, 2008.

[PSS+04] M. Peden, R. Scurfield, D. Sleet, D. Mohan, A.A. Hyder, E. Jarawan, and C. Mathers. World report on road traffic injury prevention. Technical Report WA 275, World Health Organization, 2004.

[PSvN+11] J. Ploeg, B.T.M. Scheepers, E. van Nunen, N. van de Wouw, and H. Nijmeijer. Design and experimental evaluation of cooperative adaptive cruise control. In *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 260–265, October 2011.

[QGMF14] X. Qian, J. Gregoire, F. Moutarde, and A. De La Fortelle. Priority-based coordination of autonomous and legacy vehicles at intersection. In *Proceedings of the 17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1166–1171, October 2014.

[Raj12] R. Rajamani. *Vehicle dynamics and control*. Springer, 2nd edition, 2012.

[RCC10] T. Robinson, E. Chan, and E. Coelingh. Operating platoons on public motorways: An introduction to the SARTRE platooning programme. In *Proceedings of the 17th ITS World Congress*, October 2010.

[RPM10] R.P. Roess, E.S. Prassas, and W.R. McShane. *Traffic engineering*. Prentice Hall, Upper Saddle River, NJ 07458, 4th edition, 2010.

[RSK07] S. Rezaei, R. Sengupta, and H. Krishnan. Reducing the communication required by DSRC-based vehicle safety systems. In *Proceedings of the 10th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 361–366, September 2007.

[RSS11] R.H. Rasshofer, M. Spies, and H. Spies. Influences of weather phenomena on automotive laser radar systems. *Advances in Radio Science*, 9:49–60, 2011.

[SAE15] SAE standard. SAE j2735: Dedicated Short Range Communications (DSRC) Message Set Dictionary. Society of Automotive Engineers (SAE), 2015.

[SBC11] M.L. Sin, M. Bouroche, and V. Cahill. Scheduling of dynamic participants in real-time distributed systems. In *Proceedings of the 30th IEEE Symposium on Reliable Distributed Systems (SRDS)*, pages 245–254, October 2011.

[SC13] M. Segata and R. Lo Cigno. Automatic emergency braking: Realistic analysis of car dynamics and network performance. *IEEE Transactions on Vehicular Technology*, 62(9):4150–4161, November 2013.

[SCS00] A. Saltelli, K. Chan, and E.M. Scott. *Sensitivity analysis*. John Wiley & Sons Ltd, Chichester, West Sussex, England, 2000.

[SE10] F. Schmidt-Eisenlohr. *Interference in vehicle-to-vehicle communication networks – Analysis, modeling, simulation and assessment*. PhD thesis, Karlsruhe Institute of Technology (KIT), 2010.

[SEH10] F. Schmidt-Eisenlohr and H. Hartenstein. Simulation-based capacity estimates for local broadcast transmissions. In *Proceedings of the 7th ACM International Workshop on VehiculAr InterNETworking (VANET)*, pages 21–30, New York, NY, USA, September 2010.

[SGAK14] M. Sepulcre, J. Gozalvez, O. Altintas, and H. Kremo. Adaptive beaconing for congestion and awareness control in vehicular networks. In *Proceedings of the IEEE Vehicular Networking Conference (VNC)*, pages 81–88, Dec 2014.

[SGHH10] M. Sepulcre, J. Gozalvez, J. Härri, and H. Hartenstein. Application-based congestion control policy for the communication channel in VANETs. *IEEE Communications Letters*, 14(10):951–953, October 2010.

[Shl78] S.E. Shladover. Longitudinal control of automated guideway transit vehicles within platoons. *Journal of Dynamic Systems, Measurement, and Control*, 100(4):302–310, 1978.

[SLL+10] R. Schmidt, R. Lasowski, T. Leinmueller, C. Linnhoff-Popien, and G. Schafer. An approach for selective beacon forwarding to improve cooperative awareness. In *Proceedings of the IEEE Vehicular Networking Conference (VNC)*, pages 182–188, Dec 2010.

[SLS+10] R. Schmidt, T. Leinmuller, E. Schoch, F. Kargl, and G. Schafer. Exploration of adaptive beaconing for efficient intervehicle safety communication. *IEEE Network*, 24(1):14–19, January 2010.

[SM03] B. Sultan and M. McDonald. Assessing the safety benefit of automatic collision avoidance systems. In *Proceedings of the*

18th International Technical Conference on the Enhanced Safety of Vehicles (ESV), May 2003.

[Spe01] Spectrum Planning Team and Radiofrequency Planning Group. A review of automotive radar systems – Devices and regulatory frameworks. Technical Report SP 4/01, Australian Communications Authority, April 2001.

[SRA+08] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola. *Global sensitivity analysis: The primer*. John Wiley & Sons Ltd, Chichester, West Sussex, England, 2008.

[SSH98] P. Seiler, B. Song, and J.K. Hedrick. Development of a collision avoidance system. Technical report, University of California-Berkley, February 1998.

[Sum00] H. Summala. Brake reaction times and driver behavior analysis. *Transportation Human Factors*, 2(3):217–226, 2000.

[T. 11] T. Unger. Berichte der ADAC Unfallforschung. Konstellationen bei Auffahrunfällen. Technical report, ADAC Unfallforschung im ADAC Technik Zentrum Landsberg/Lech, March 2011.

[Tao89] G.T. Taoka. Brake reaction times of unalerted drivers. *Institute of Transportation Engineers (ITE) Journal*, 59(3):19–21, March 1989.

[TBS14] M. Tlig, O. Buffet, and O. Simonin. Decentralized traffic management: A synchronization-based intersection control. In *Proceedings of the International Conference on Advanced Logistics and Transport (ICALT)*, pages 109–114, May 2014.

[TH99] C. Tingvall and N. Haworth. Vision zero – an ethical approach to safety and mobility. In *Proceedings of the 6th ITE International Conference Road Safety & Traffic Enforcement: Beyond 2000*, September 1999.

[Thea] The MathWorks Inc., Natick, Massachusetts, United States. Matlab and statistics toolbox. `http://www.mathworks.com`. Accessed: 2015-10-10.

[Theb] The National Highway Traffic Safety Administration. Federal Motor Vehicle Safety Standards. `http://www.fmvss.com/`. Accessed: 2015-10-10.

[The05] The Crash Avoidance Metrics Parnership (CAMP) Vehicle Safety Communications Consortium. Vehicle safety communications

project. Task 3. Final report. Identify intelligent vehicle safety applications enabled by DSRC. Technical Report DOT HS 809 859, U.S. Department of Transportation, National Highway Traffic Safety Administration, March 2005.

[The11] The UN Road Safety Collaboration. Global Plan for the Decade of Action for Road Safety 2011-2020, 2011.

[THH00] M. Treiber, A. Hennecke, and D. Helbing. Congested traffic states in empirical observations and microscopic simulations. *Phys. Rev. E*, 62:1805–1824, Aug 2000.

[Thu99] W. Thubauville. *Straßenverkehrsordnung: StVO*. Bundesanzeiger Verlag GmbH, Cologne, Germany, 1999.

[T.J82] T.J. Triggs and W.G. Harris. Reaction time of drivers to road stimuli. Technical Report Human Factors Report No. HFR-12, Monash University Accident Research Centre, June 1982.

[TJHD13] T. Tielert, D. Jiang, H. Hartenstein, and L. Delgrossi. Joint power/rate congestion control optimizing packet reception in vehicle safety communications. In *Proceeding of the 10th ACM International Workshop on Vehicular Inter-networking, Systems, and Applications (VANET)*, pages 51–60, New York, NY, USA, 2013.

[TJP97] C. Thorpe, T. Jochem, and D. Pomerleau. The 1997 automated highway free agent demonstration. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, pages 496–501, November 1997.

[TMMSH09] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein. Vehicle-to-vehicle communication: Fair transmit power control for safety-critical information. *IEEE Transactions on Vehicular Technology*, 58(7):3684–3703, Sept 2009.

[tra12] Traffic safety facts 2012. A compilation of motor vehicle crash data from the Fatality Analysis Reporting System and the General Estimates System. Technical Report DOT HS 812 032, U.S. Department of Transportation, National Highway Traffic Safety Administration, 2012.

[TSW13] S. Toma, E. Swanson, and N. G. Wassim. Light vehicle crash avoidance needs and countermeasure profiles for safety applications based on vehicle-to-vehicle communications. Technical Report DOT HS 811 733, U.S. Department of Transportation, National Highway Traffic Safety Administration, April 2013.

[TY10] A. Tang and A. Yip. Collision avoidance timing analysis of DSRC-based vehicles. *Accident Analysis & Prevention*, 42(1):182–195, 2010.

[UB10] National Research Council (U.S.) and Transportation Research Board. *Highway capacity manual*. Transportation Research Board, Washington, D.C., 5th edition, 2010.

[UBoDotHSM+10] National Research Council (U.S.), Transportation Research Board, Task Force on Development of the Highway Safety Manual, American Association of State Highway, Transportation Officials, Joint Task Force on the Highway Safety Manual, and National Cooperative Highway Research Program. *Highway safety manual*. American Association of State Highway and Transportation Officials, 444 North Capitol Street, NW, Suite 249, Washington, D.C. 20001, 1st edition, 2010.

[VAG+11] M. Varga, A. Ashok, M. Gruteser, N. Mandayam, W. Yuan, and K. Dana. Demo: Visual MIMO based LED – Camera communication applied to automobile safety. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, pages 383–384, June 2011.

[vAvDV06] B. van Arem, C.J.G. van Driel, and R. Visser. The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Transactions on Intelligent Transportation Systems*, 7(4):429–436, December 2006.

[vdH90] A.R.A. van der Horst. *A time-based analysis of road user behaviour in normal and critical encounters*. PhD thesis, Delft University of Technology, 1990.

[VDS08] M. VanMiddlesworth, K. Dresner, and P. Stone. Replacing the stop sign: Unmanaged intersection control for autonomous vehicles. In *Proceedings of the AAMAS Workshop on Agents in Traffic and Transportation*, pages 94–101, Estoril, Portugal, May 2008.

[vERH12] M. van Eenennaam, A. Remke, and G. Heijenk. An analytical model for beaconing in VANETs. In *Proceedings of the IEEE Vehicular Networking Conference (VNC)*, pages 9–16, November 2012.

[vEWKH09] M. van Eenennaam, W.K. Wolterink, G. Karagiannis, and G. Heijenk. Exploring the solution space of beaconing in VANETs. In *Proceedings of the IEEE Vehicular Networking Conference (VNC)*, pages 1–8, October 2009.

[Vin12] A. Vinel. 3GPP LTE versus IEEE 802.11p/WAVE: Which technology is able to support cooperative vehicular safety applications? *IEEE Wireless Communications Letters*, 1(2):125–128, April 2012.

[Vin14] J.W. Vincoli. *Basic guide to system safety*. John Wiley & Sons, Inc. Publication, Hoboken, New Jersey, 3rd edition, 2014.

[vJKS+05] T. von Jan, T. Karnahl, K. Seifert, J. Hilgenstock, and R. Zobel. Don't sleep and drive – VW's fatigue detection technology. In *Proccedings of the 19th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, pages 1–12, June 2005.

[WATM09] J. Wu, A. Abbas-Turki, and A. El Moudni. Discrete methods for urban intersection traffic controlling. In *Proceedings of the 69th IEEE Semiannual Vehicular Technology Conference (VTC-Spring)*, pages 1–5, April 2009.

[WBA98] Glenn R. Widmann, William A. Bauson, and Steven W. Alland. Development of collision avoidance systems at delphi automotive systems. In *Proceedings of the 1998 IEEE International Conference on Intelligent Vehicles Vol. 2*, pages 353–358, 1998.

[WBMD97] T.B. Wilson, W. Butler, D.V. McGehee, and T. Dingus. Forward-looking collision warning system performance guidelines. Technical Report 970456, SAE Technical Paper, 1997.

[Wen05] J. Wenger. Automotive radar – status and perspectives. In *Proceedings of the IEEE Compound Semiconductor Integrated Circuit Symposium, (CSIC)*, pages 21–24, October 2005.

[WER+03] L. Wischoff, A. Ebner, H. Rohling, M. Lott, and R. Halfmann. SOTIS – A self-organizing traffic information system. In *Proceedings of the 57th IEEE Semiannual Vehicular Technology Conference (VTC-Spring)*, volume 4, pages 2442–2446 vol.4, April 2003.

[WH06] J. Whitelegg and G. Haq. Vision Zero: Adopting a Target of Zero for Road Traffic Fatalities and Serious Injuries. Stockholm Environment Institute, 2006.

[Wis97] B. Wisdom. Skepticism & Credulity. Finding the balance between Type I and Type II errors. *Skeptic Magazine*, 5(2), 1997.

[WOH04] B. Wu, T.L. Ooi, and Z.J. He. Perceiving distance accurately by a directional process of integrating ground information. *Nature*, 428(6978):73–77, 2004.

[XMKS04] Q. Xu, T. Mak, J. Ko, and R. Sengupta. Vehicle-to-vehicle safety messaging in DSRC. In *Proceedings of the 1st ACM Workshop on Vehicular Ad hoc Networks (VANET)*, pages 19–28, New York, NY, USA, 2004.

[XQX14] X. Xiang, W. Qin, and B. Xiang. Research on a DSRC-based rear-end collision warning model. *IEEE Transactions on Intelligent Transportation Systems*, 15(3):1054–1065, June 2014.

[XSJC03] Q. Xu, R. Segupta, D. Jiang, and D. Chrysler. Design and analysis of highway safety communication protocol in 5.9 GHz dedicated short range communication spectrum. In *Proceedings of the 57th IEEE Semiannual Vehicular Technology Conference (VTC-Spring)*, volume 4, pages 2451–2455, April 2003.

[YDM09] F. Yan, M. Dridi, and A. El Moudni. Autonomous vehicle sequencing algorithm at isolated intersections. In *Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, October 2009.

[YEY+04] J. Yin, T. ElBatt, G. Yeung, B. Ryu, S. Habermas, H. Krishnan, and T. Talty. Performance evaluation of safety applications over DSRC vehicular ad hoc networks. In *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks (VANET)*, pages 1–9, New York, NY, USA, 2004.

[YLVF04] X. Yang, J. Liu, N.F. Vaidya, and Z. Feng. A vehicle-to-vehicle communication protocol for cooperative collision warning. In *Proceedings of the 1st Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MOBIQUITOUS)*, pages 114–123, August 2004.

[ZAG06] Y. Zhang, E.K. Antonsson, and K. Grote. A new threat assessment measure for collision avoidance systems. In *Proceedings of the 9th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 968–975, September 2006.

[ZR14] I.H. Zohdy and H.A. Rakha. Intersection management via vehicle connectivity: The intersection cooperative adaptive cruise control system concept. *Journal of Intelligent Transportation Systems*, pages 1–16, 2014.

[ZSO+05] Y. Zang, L. Stibor, G. Orfanos, S. Guo, and H.-J. Reumerman. An error model for inter-vehicle communications in highway scenarios at 5.9GHz. In *Proceedings of the 2nd ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN)*, pages 49–56, New York, NY, USA, 2005.