A Simulation-Based Safety Analysis of CACC-Enabled Highway Platooning

Maytheewat Aramrattana

Supervisors:
Tony Larsson

Co-Supervisors:
Cristofer Englund
Jonas Jansson

DOCTORAL THESIS | Halmstad University Dissertations no. 51

A Simulation-Based Safety Analysis of CACC-Enabled Highway Platooning
© Maytheewat Aramrattana
Halmstad University Dissertations no. 51
ISBN 978-91-88749-07-9 (printed)
ISBN 978-91-88749-08-6 (pdf)
Publisher: Halmstad University Press, 2018 | www.hh.se/hup
Printer: Media-Tryck, Lund

Abstract

Cooperative Intelligent Transport Systems (C-ITS) enable actors in the transport systems to interact and collaborate by exchanging information via wireless communication networks. There are several challenges to overcome before they can be implemented and deployed on public roads. Among the most important challenges are testing and evaluation in order to ensure the safety of C-ITS applications.

This thesis focuses on testing and evaluation of C-ITS applications with regard to their safety using simulation. The main focus is on one C-ITS application, namely platooning, that is enabled by the Cooperative Adaptive Cruise Control (CACC) function. Therefore, this thesis considers two main topics: *i*) what should be modelled and simulated for testing and evaluation of C-ITS applications? and *ii*) how should CACC functions be evaluated in order to ensure safety?

When C-ITS applications are deployed, we can expect traffic situations which consist of vehicles with different capabilities, in terms of automation and connectivity. We propose that involving human drivers in testing and evaluation is important in such mixed traffic situations. Considering important aspects of C-ITS including human drivers, we propose a simulation framework, which combines driving-, network-, and traffic simulators. The simulation framework has been validated by demonstrating several use cases in the scope of platooning. In particular, it is used to demonstrate and analyse the safety of platooning applications in cut-in situations, where a vehicle driven by a human driver cuts in between vehicles in platoon. To assess the situations, time-to-collision (TTC) and its extensions are used as safety indicators in the analyses.

The simulation framework permits future C-ITS research in other fields such as human factors by involving human drivers in a C-ITS context. Results from the safety analyses show that cut-in situations are not always hazardous, and two factors that are the most highly correlated to the collisions are relative speed and distance between vehicles at the moment of cutting in. Moreover, we suggest that to solely rely on CACC functions is not sufficient to handle cut-in situations. Therefore, guidelines and standards are required to address these situations properly.

Acknowledgements

First and foremost, I would like to express my gratitude to my main supervisor, Prof. Tony Larsson; co-supervisors, Dr. Jonas Jansson and Dr. Cristofer Englund; and my mentor and manager at the Swedish National Road and Transport Research Institute (VTI), Arne Nåbo, for their guidance with encouragement and patience. I learned a lot from them, and I am grateful for having had them as supervisors and mentor. They have given me this opportunity and endless support. Financially, this work has been supported by the Knowledge Foundation (KKS), Halmstad University, SAFER[1], and VTI.

I am grateful to my parents, Manoon and Orasa, who have always inspired me, and provided me with their unconditional love and moral support. I dedicate this to them and my late grandmother.

At the school of Information Technology (ITE), I would like to thank Slawomir Nowaczyk for inspiring discussions that we had, and wisdom that he shared. I would like to thank Stefan Byttner for his advice and support as the director of doctoral education at ITE. Many thanks to Wojciech Mostowski, who has given me opportunities to involve in two exciting projects—GCDC 2016, and CPS Challenge 2018. I would like to also acknowledge the support from Magnus Jonsson, who leads the Embedded and Intelligent Systems Industrial Graduate School (EISIGS); and Jörgen Carlsson and Walid Taha, who are my previous and current managers. Moreover, I would like to thank Eva Nestius, Stefan Gunnarsson, and Jessika Rosenberg who are always eager and happy to help me with all the administrative problems.

I would like to thank all my colleagues at VTI for their assistance and making me feel comfortable in a new working environment, with a special mention to Eva Åström, Bruno Augusto, and Anders Andersson.

To all my friends, family, and colleagues who share their passions and kept me motivated through all these years, thank you very much! All of you make my life in Sweden pleasant and lively. Thank you Hassan Nemati for all the great memories we shared; and for always being there and listen to my (sometimes nonsense) stories. Thank you Yuantao Fan for all the moments we shared from

[1]Vehicle and Traffic Safety Centre at Chalmers (SAFER)

the time we studied master's program. Thank you Saeed Gholami Shahbandi for sharing all the great ideas and advices since the time you supervised my master's thesis. Thank you Sebastian Kunze, Pablo Del Moral, Ece Calikus, and Kevin Hernández Díaz for all the wonderful times we had. Thank you Jennifer David, Yingfu Zeng, Xu Fei, Süleyman Savas, Essayas Gebrewahid, Iulian Carpatorea, Mahsa Varshosaz, Sepideh Pashami, Anita Pinheiro Sant'Anna, Adam Duracz, Deycy Janeth Sanchez Preciado, Wagner Ourique de Morais, Alina Franck, Lotta Pham, Julia Söderström Malmborg, Siddhartha Khandelwal for all the great time we shared, and their support beyond work.

Last but not least, I would like to express my greatest gratitude to Mijin Kim for her trust, encouragement, and love throughout all the years that we have been together.

List of Publications

This thesis summarises the following publications:

Paper I Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Cristofer Englund. Dimensions of Cooperative Driving, ITS and Automation. In *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pages 144–149. IEEE, 2015

Paper II Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Arne Nåbo. A simulation framework for cooperative intelligent transport systems testing and evaluation. *Transportation Research Part F: Traffic Psychology and Behaviour*, 2017.

Paper III M. Aramrattana, T. Larsson, C. Englund, J. Jansson, and A. Nabo. Simulation of cut-in by manually driven vehicles in platooning scenarios. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 315–320, Oct 2017.

Paper IV Maytheewat Aramrattana, Cristofer Englund, Jonas Jansson, Tony Larsson, and Arne Nåbo. Safety analysis of cooperative adaptive cruise control in vehicle cut-in situations. In *Proceedings of 2017 4th International Symposium on Future Active Safety Technology towards Zero-Traffic-Accidents (FAST-zero)*. Society of Automotive Engineers of Japan, 2017.

Paper V M. Aramrattana, C. Englund, T. Larsson, J. Jansson, and A. Nåbo. Safety Evaluation of Highway Platooning Under a Cut-In Situation Using Simulation. *IEEE Transactions on Intelligent Transportation Systems*, 2018. (submitted in August 2018).

Other publications (if the corresponding author is not the first author, the author's name is marked in **bold**):

- Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Arne Nåbo. Extended Driving Simulator for Evaluation of Cooperative Intelligent Transport Systems. In *Proceedings of the 2016 Annual ACM Conference on SIGSIM Principles of Advanced Discrete Simulation*, SIGSIM-PADS '16, pages 255–258, New York, NY, USA, 2016. ACM

- Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Arne Nåbo. Cooperative Driving Simulation. *The Driving Simulation Conference 2016 VR*, 2016

- M. Aramrattana, J. Detournay, C. Englund, V. Frimodig, O. U. Jansson, T. Larsson, **W. Mostowski**, V. D. Rodríguez, T. Rosenstatter, and G. Shahanoor. Team Halmstad Approach to Cooperative Driving in the Grand Cooperative Driving Challenge 2016. *IEEE Transactions on Intelligent Transportation Systems*, 19(4):1248–1261, April 2018.

- Maytheewat Aramrattana, **Raj Haresh Patel**, Cristofer Englund, Jérôme Härri, Jonas Jansson, and Christian Bonnet. Evaluating Model Mismatch Impacting CACC Controllers in Mixed Traffic using a Driving Simulator. In *29th IEEE Intelligent Vehicles Symposium (IV 2018)*. IEEE, 2018.

- Maytheewat Aramrattana, Anders Andersson, Håkan Burden, Frida Reichenberg, and Niklas Mellegård. Testing Cooperative Intelligent Transport Systems in Driving Simulators. *The Driving Simulation Conference 2018 VR*, 2018.

- Patrizio Pelliccione, Avenir Kobetski, Tony Larsson, Maytheewat Aramrattana, Tobias Aderum, S Magnus Ågren, Göran Jonsson, Rogardt Heldal, Carl Bergenhem, and Anders Thorsén. Architecting cars as constituents of a system of systems. In *Proceedings of the International Colloquium on Software-intensive Systems-of-Systems at 10th European Conference on Software Architecture*, pages 5:1–5:7. ACM, 2016.

List of Abbreviations

ACC Adaptive Cruise Control

BASt Federal Highway Research Institute (Germany)

CACC Cooperative Adaptive Cruise Control

CC Cruise Control

C-ITS Cooperative Intelligent Transport Systems

DRAC Deceleration-rate-to-avoid-a-crash

HiL Hardware-in-the-loop

ISO International Organization for Standardization

ITS Intelligent Transport Systems

MiL Model-in-the-loop

NHTSA National Highway Traffic Safety Administration (USA)

PET Post-encroachment-time

Plexe The Platooning Extension for Veins

SAE Society of Automotive Engineering

SiL Software-in-the-loop

SSM Surrogate Safety Measures

SUMO Simulation of Urban Mobility

TCP Traffic Control Protocol

TET Time Exposed Time-to-collision

TIT Time Integrated Time-to-collision

TraCI Traffic Control Interface

TTC Time-to-collision

VANET Vehicular Ad hoc Network

VDA German Association of the Automotive Industry

Veins Vehicle in Network Simulation

VTI The Swedish National Road and Transport Research Institute

V2I Vehicle-to-infrastructure

V2V Vehicle-to-vehicle

V2X Vehicle-to-everything (usually refers to V2I and V2V together)

Contents

List of Figures

List of Tables

Chapter 1
Introduction

Cooperative intelligent transport systems (C-ITS) are one of the most recent concepts in the field of transport systems. As an improvement to intelligent transport systems (ITS), which incorporate information and communication technology with transport systems, C-ITS interconnects ITS stakeholders with wireless communication. Wireless communication between vehicles in these systems is often referred to as vehicle-to-vehicle (V2V) communication, while communication between vehicles and road infrastructures is known as vehicle-to-infrastructure (V2I) communication. Together, these two types of communication are sometimes referred to as V2X communication, which stands for vehicle-to-everything. Besides connectivity, actors in C-ITS use wireless communication and vehicle automation to cooperate and interact with each other in order to achieve the goals of improving transport systems by means of more efficient use of road space, improved traffic flow, safer traffic, and reduced fuel consumption, for example. Although vehicles and road infrastructures are usually the main points of focus in research, actors in such transport systems may also include pedestrians, cyclists, traffic management centres, etc.

This doctoral thesis summarises the work done in this context, with goals towards safety evaluation and testing methods for C-ITS applications using simulation. In particular, the studies summarised in this thesis focus on platooning applications where vehicles are controlled by cooperative adaptive cruise control (CACC) functions. In relation to this, this thesis looks mainly at evaluation of the safety of the platooning applications in the event of cutting-in by a vehicle that is driven manually and without wireless communication capability. Motivations for selecting the specific application, approaches, and scenarios are indicated in Section 1.1. Research questions addressed in this thesis are described in Section 1.2. Lastly, an overview of this thesis is presented in Section 1.3.

Note that this thesis includes the work presented in the author's *licentiate*[1] thesis [13].

1.1 Motivations

Using wireless communication enables C-ITS actors to access information that is difficult to obtain with on-board sensors, or beyond the range of their own sensors; acceleration of two vehicles driving ahead, for example, or information indicating that the vehicle in front is about to turn left. C-ITS applications can potentially improve transport systems in many aspects: by improving traffic flow, making more efficient use of road space, increasing systems actors' awareness, etc. On the other hand, there are many challenges to be overcome before the C-ITS applications can be implemented and deployed on public roads. One of the challenges involves ensuring the safety and integrity of C-ITS applications in real traffic situations, which involve different road users that sometimes behave in unpredictable ways. Hence, testing and evaluation are important as a way of ensuring that C-ITS actors are able to operate together with their cooperative partners as well as conventional vehicles. Testing and evaluation are two of the challenges presented when it comes to developing and deploying C-ITS applications, because these systems are becoming increasingly complex and testing every possible situations is no longer feasible. Therefore, new approaches and methods are required for efficient and effective testing of new applications.

Many aspects and use cases in respect of these applications need to be tested and evaluated in order to support deployment of C-ITS applications—such as platooning—on public roads. Research literature and projects in this context have reported on studies related to challenges, potential benefits of the applications, and implementation approaches, as will be summarised in Chapter 2. Apart from those, safety is another topic that needs to be addressed prior to deployment of C-ITS applications. There are few examples of research literature contributing to safety evaluation of C-ITS applications, especially in the field of platooning applications as summarised in this recent survey [14]. The survey concludes that there are only few studies, which deal with the safety of platooning: they complement each other but none of them provides a complete picture of the field. Moreover, there are gaps in the knowledge, particularly more practical work based on established safety concepts are needed. The survey also mentions that many different variants of platooning exist and the best one is still to be decided. Nevertheless, safety should be one of the deciding factors. Thus the motivation for conducting research in testing and safety evaluation is well-supported.

[1]In Sweden, "A licentiate degree covering at least 120 credits, with a thesis worth at least 60 credits, may be awarded as a separate qualification or as a step on the way to a doctoral degree." Source: `https://www.studera.nu/startpage/doctoral-studies/degrees/licentiate-degree/`. In this case, the author's licentiate degree is in the field of engineering within the Computer Science and Engineering subject area.

Platooning applications, that are enabled by a CACC function (or functions), were selected as test subjects due to the high potential for their deployment in the near future. Since cruise control (CC) and adaptive cruise control (ACC) are available in most new vehicles today, it is possible to expect to see CACC functions—which demonstrates further improvements in many ways compared to CC and ACC—on the road before long, potentially also facilitating the deployment of platooning applications. Several research results relating to platooning have been reported in the literature, and a selection of these will be summarised in Section 2.3.

As suggested in [15], CACC functions normally intend to automate only longitudinal control, and assume "level 1" automation in the levels of driving automation proposed by the Society of Automotive Engineering (SAE) [16]. The *level 1* automation implies that the driver is still responsible for monitoring the environment and has to act as a fall-back. Moreover, we can expect to see combinations of manually driven (level 0) and automated (levels 1-5) vehicles in traffic for a long time to come until everything is automated—if, indeed, this is even possible. This general assumption and situation suggest that it is necessary to involve human drivers in testing and evaluation. Therefore, this raises an important question: *how effectively are connected and automated functions able to handle interactions and coexist with human drivers, pedestrians, cyclists, etc. in traffic systems?*

Simulation is useful in many research fields, especially in fields where working with real products or situations would be costly, time-consuming, or dangerous. This is also the case when a technology is new and prototypes may not be available. These advantages of simulation fit our needs perfectly, for the following reasons. Firstly, as mentioned above, human drivers need to be involved in testing and evaluation. To evaluate safety, the systems need to be tested under risky and hazardous conditions as well as normal situations. As human drivers are involved, creating such situations with real vehicles is even more costly and dangerous. Secondly, CACC functions are not available commercially and are still being developed. Also, many variants of CACC exist, as mentioned above. It would be difficult to obtain a fleet of vehicles equipped with CACC. Moreover, standards and requirements in respect of C-ITS are not well-defined yet, and may be changed in the future. Using simulation offers an alternative that is more flexible than creating a hardware prototype. Therefore, simulation is a suitable tool for testing and evaluation within the scope of this thesis.

However, very few simulation tools consider all aspects of C-ITS and fit our purposes. Such simulation tools should be able to include at least one human driver in the simulation and adequately support modelling of an important set of C-ITS environments and applications; CACC and platooning in particular. Thus a simulation framework—combining driving, network, and traffic simulators—is developed in this work. Development of the simulation framework is presented in **Paper II**, and more details will be presented and discussed in Chapter 4.

The majority of studies regarding testing and evaluation of platooning applications consider all vehicles in the studies to be identical, and we refer to these as *homogeneous* scenarios. To complement the research field and extend the scenarios further, this thesis is also interested in *heterogeneous* scenarios, where traffic is made up of a mixture of different types of vehicles in terms of ability to communicate and vehicle automation levels. Moreover, another major challenge involves ensuring that the applications are able to operate during disruptions and failures such as other road users cutting in, communication and sensor failures, etc. These are the challenges that platoons will face in real traffic situations, and they need to be considered in testing and evaluation processes before new C-ITS functions can be deployed.

In particular, there is interest in how automated vehicles in a platoon will react to conventional vehicles when they disturb the platoon by cutting in. In the opinion of the author, cut-in situations occur commonly in today's traffic, and platoons will also face such challenges unless they are assigned to dedicated lanes. Cut-in refers to situations, in which a vehicle changes lane to slot into a gap between two other vehicles. This could be a frequent occurrence during the early C-ITS deployment phases, where connected and automated vehicles will be sharing the roads with conventional vehicles. Furthermore, cut-in manoeuvres by other road users, especially at on-ramps, was mentioned as one of the main reasons to disengage platooning functions in the European Truck Platooning Challenge [17]. Apart from safety concerns, cutting in is also mentioned as irritating as well as time-consuming while reconnecting after the platoon is separated by the cut-in vehicle, according to a truck driver in the challenge. Despite the importance of this factor in the opinion of the author, only few studies have considered cut-in situations: examples are presented in [12, 18, 19, 20].

Therefore, this thesis aims to investigate the safety evaluation of highway[2] platooning in scenarios that involve cutting in by a conventional vehicle controlled manually by a human driver (as opposed to automated). This is done using a simulation framework developed as part of this study, which also aims to facilitate the research studies presented.

1.2 Research questions and thesis statements

The main research question to which this work contributes is ***how can the safety of C-ITS applications be evaluated efficiently using simulation?*** In relation to this, the following two research topics are considered:

- How to model, test, and evaluate new C-ITS applications, in an organised and efficient manner.

[2]In this thesis, this is limited to controlled-access highway known as motorway, freeway, or expressway.

- How to perform testing and evaluation using simulation of C-ITS situations.

With respect to the first research topic above, simulation has been selected as a tool for modelling, testing, and evaluating new C-ITS applications. The more specific research question below arose as a consequence.

RQ-1 What should be modelled and simulated for testing and evaluation of C-ITS applications?

Regarding how to perform testing and evaluation of C-ITS by means of simulation, the focus of this thesis is on evaluation of the safety of CACC controllers and platooning applications as the function and system under test. The question, then, is *under which situations should CACC controllers be evaluated in order to ensure safety?* As motivated above, cut-in situations are identified as an important scenario to investigate. Therefore, with respect to cut-in situations, the following research questions are also addressed in this thesis:

RQ-2 How should CACC controllers be evaluated in order to ensure safety?

RQ-3 How can we assess the safety of highway platooning applications?

RQ-4 What defines safe CACC controller behaviours?

RQ-5 Can a CACC controller handle all aspects of platooning alone?

RQ-6 How can CACC controllers used in platooning deal with disturbances caused by conventional vehicles driven manually?

The following thesis statements are formulated in relation to the six research questions listed above:

T-1 Safety of C-ITS applications can be evaluated by means of simulation.

T-2 Testing and evaluation of C-ITS using simulation should include a way of involving human drivers.

T-3 In the context of highway platooning, cut-in situations are hazardous and CACC controllers alone will be unable to handle such situations properly.

1.3 Overview of the thesis

As mentioned earlier, the goal of this doctoral thesis is to contribute to testing and safety evaluation methods for C-ITS applications using simulation. Of C-ITS applications, CACC and platooning applications have been selected as use cases for evaluation in this thesis.

Firstly, **Paper I** contributes to understanding the concept and complexity of C-ITS and identifies challenges in the research field. Among the challenges, this thesis focuses on addressing the lack of testing and evaluation methods and tools for C-ITS applications in mixed traffic environments.

Thus a simulation framework—which is a combination of driving, network, and traffic simulators—was proposed as a research tool in **Paper II**. The simulation framework has been developed for testing and evaluation of C-ITS applications, in particular CACC and platooning. It also allows a human driver to be involved in the simulation scenarios via the driving simulator.

To demonstrate the importance of including a human driver, the platooning applications were evaluated in scenarios where a platoon is disturbed by a cut-in manoeuvre by a vehicle driven manually by a human driver and has no wireless communication with the platooning vehicles. Such a vehicle is referred to as a *manually driven vehicle* or a *conventional vehicle* in this thesis. Moreover, situations of this kind are referred to as a common hazard for platooning vehicles [14]. Thus this thesis contributes to a better understanding of such situations and challenges that they will bring. These are presented in **Paper III**, **Paper IV**, and **Paper V**.

Furthermore, in **Paper V**, safety evaluation of CACC controllers is demonstrated using existing safety indicators, namely time-to-collision (TTC), Time Exposed Time-to-collision (TET), and Time Integrated Time-to-collision (TIT). The results support the fact that these safety indicators are suitable for safety evaluation of the cut-in scenario considered in the **Paper V**, due to the fact that the main factors for collisions in the cut-in scenarios considered in the **Paper V**—i.e. the difference between speed of vehicles and the distance between vehicles—are both captured in the calculation of TTC, TET, and TIT.

1.3.1 Concepts and definitions

Many concepts in the field of C-ITS are relatively new, and a variety of definitions are used in the research literature. Thus this section describes the definitions and concepts adopted in this thesis.

Platooning refers to an application where two or more vehicles follow each other with a short inter-vehicular distance between them. This group of vehicles is called a *platoon*, and the term *platooning vehicles* refers to vehicles in a platoon. The first vehicle in a platoon is referred to as the *platoon leader*. In the context of this thesis, all platooning vehicles—including the leader—are connected and automated, although the leader can be controlled manually in some cases in the literature, and only the followers are automated. This thesis assumes that all vehicles are equipped with a wireless communication module for V2V communication. Moreover, in this thesis only longitudinal control along the road is automated for platooning vehicles, although control of lateral dimension may also be automated as described in [21].

As an improvement to ACC controllers, CACC controllers allow for improved longitudinal control based on the use of wireless communication, and if sophisticated enough can also support automated longitudinal control in platooning applications. There are two commonly used approaches as regards maintaining inter-vehicular distance: *i*) Constant Distance Gap (CDG) and *ii*) Constant Time Gap (CTG). The constant distance gap approach maintains a fixed inter-vehicular distance regardless of speed of the vehicles. The constant time gap, however, maintains a fixed *time gap* between the front bumper of the ego vehicle and the rear bumper of the preceding vehicle. Thus this distance is dependent on the speed of the vehicles. For example, a 1-second time gap at a speed of 25 m/s corresponds to a 25-metre inter-vehicular gap, and hence this gap will become 30 metres when the vehicle is travelling at 30 m/s.

Definitions and operation concepts are summarised in [15, 22], which defines clear points of separation between CACC and platooning. According to their definitions, the CACC merely provides longitudinal control of the vehicle; pairwise information is used by the controller. On the other hand, vehicles in platooning applications are more tightly linked using the constant distance approach and the information from the lead vehicle is required by all followers. Despite these differences, this thesis considers them to be the same category of C-ITS applications, defined above as platooning. Moreover, because this thesis investigates one controller of each kind (one typical CACC and one designed for platooning), the terms will be used interchangeably. Therefore, for the aforementioned reasons only a subset of platooning applications is considered in this thesis.

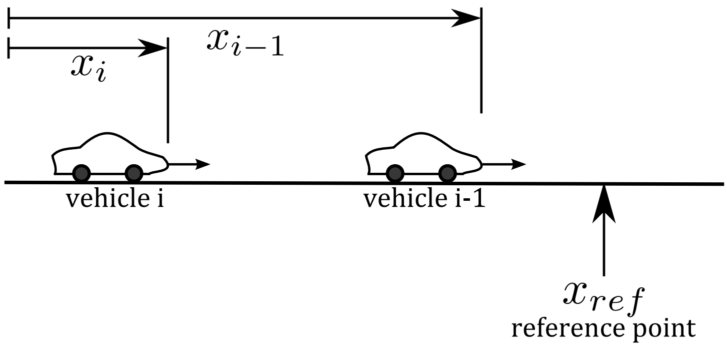


Figure 1.1: Definitions of notations for defining the time headway.

The *time gap* concept is not to be confused with *time headway*, which has a similar definition. However, time headway is defined as the time between the front bumpers of two consecutive vehicles passing a fixed reference point; whereas the time gap concept measures the time between the front bumper of the ego vehicle and the rear bumper of the preceding vehicle. According to Fig. 1.1, if we assume that $t_{ref}(i)$ is the time at which the front bumper of the vehicle i reached the reference point, i.e. $x_i = x_{ref}$, and $t_{ref}(i-1)$ is the time at which the front bumper of the vehicle $i-1$ reached the reference point, *time* *headway* between vehicles i and $i-1$, $t_h(i,i-1)$ is defined as described in the following equation 1.1:

$$t_h(i,i-1) = t_{ref}(i-1) - t_{ref}(i) \tag{1.1}$$

where

$$t_{ref}(i) = t(x_i = x_{ref})$$
$$t_{ref}(i-1) = t(x_{i-1} = x_{ref})$$

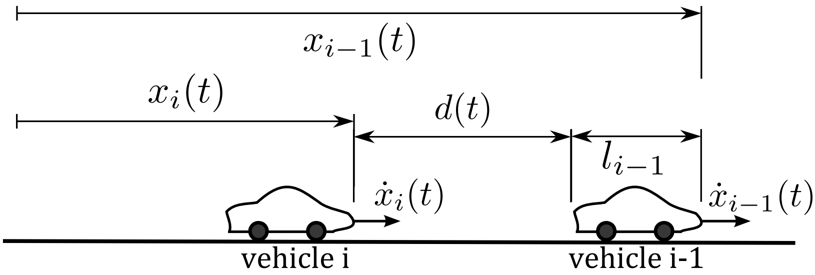


Figure 1.2: Definitions of notations for calculation of TTC and time gap.

The time gap, on the other hand, is defined using notations in Fig. 1.2. Thus the time gap between vehicle i and vehicle $i-1$, $t_g(i,i-1)$, is defined by the following equation 1.2:

$$t_g(i,i-1)(t) = \frac{d(t)}{\dot{x}_i(t)} = \frac{x_{i-1}(t) - x_i(t) - l_{i-1}}{\dot{x}_i(t)} \tag{1.2}$$

where x_i is the position of vehicle i in metres, $\dot{x}_i$ is the speed of vehicle i in m/s, and l_{i-1} is the length of vehicle $i-1$ in metres, as summarised in Fig. 1.2.

The time-to-collision (TTC) and its extensions, as proposed in [23] are used as safety indicators for safety evaluation in **Paper V**. TTC is a traffic safety indicator proposed by *John C. Hayward* in [24] as a time-measured-to-collision in order to define near-miss incidents. It is normally used in car-following situations and defined as the time remaining until a collision unless one of the vehicles changes speed and/or heading. If we assume that vehicle i follows vehicle $i-1$, the calculation of TTC at time t is defined by the following equation.

$$\mathrm{TTC}(t) = \frac{x_{i-1}(t) - x_i(t) - l_{i-1}}{\dot{x}_i(t) - \dot{x}_{i-1}(t)}, \quad \forall \dot{x}_i(t) > \dot{x}_{i-1}(t) \tag{1.3}$$

1.3.2 Outline

The rest of this thesis is organised as follows. Chapter 2 describes background and related work in the context of C-ITS, platooning, testing and evaluation of platooning, and concludes with their relationship to this work. Chapter 3 summarises results from the appended papers. A description of the components in C-ITS simulation is provided in Chapter 4, with the contributions of this thesis in the field. An overview and contributions in terms of safety analysis and evaluation of CACC controllers are elaborated upon in Chapter 5. Finally, Chapter 6 concludes this thesis, and discusses presented and future work.

Chapter 2
Background and related work

This chapter introduces the three main research areas in the context of Cooperative Intelligent Transport Systems (C-ITS) considered in this thesis.

C-ITS utilise wireless communication between vehicles (V2V) and between vehicles and road infrastructures (V2I). This wireless communication allows data exchange and cooperation between actors in such transport systems, which can be vehicles, road infrastructures, pedestrians, cyclists, etc. Consequently, C-ITS actors—especially autonomous vehicles—now have an alternative to interact with other actors and communicate their intentions. This is sometimes referred to as *cooperative driving*. Note that a vehicle's internal user interface, and the interaction between an autonomous vehicle and its driver, e.g. in [25], is beyond the scope of this thesis. This thesis also focuses on the safety aspects of platooning applications, when they are disturbed by a vehicle driven manually which has no wireless communication capability, and thus can cause hazardous situations.

2.1 Testing, evaluation, and safety analysis

Testing and evaluation are important and required at different development phases of products and services; also in the filed of C-ITS. Modelling and simulation techniques may be used in early development phases to test and evaluate ideas before making a prototype, or implementing it in a real product. Eventually, one can add more details to the simulation by using model-in-the-loop (MiL), software-in-the-loop (SiL), or hardware-in-the-loop (HiL) simulation approaches. The applications could then be tested using test beds, and finally field tests before deployment in consumer markets.

As motivated in **Paper I**, that C-ITS are complex systems with many dependencies and interactions between actors in the systems; it is difficult to test and evaluate such complex systems, or even create a model of an entire system. In this regard, we see a need for tools and methodologies for testing and evaluation of C-ITS, especially in terms of safety. Hence the focus of this thesis is on testing and evaluation of C-ITS applications using simulation, because it is more cost-effective and safer than creating a test bed or do a field test. Especially for C-ITS, where at least two actors need to be involved, it could be costly to have access to many vehicles or infrastructure platforms that comply with C-ITS standards. Furthermore, our focus is on safety-related issues where limits of the system might need to be stretched; hazardous and risky situations that may lead to collisions need to be investigated. Involving real actors in such situations may lead to catastrophic consequences.

Moreover, in early years of new systems deployment, the newly developed systems—whether they are automated driving functions or C-ITS applications—would have to co-exist with conventional vehicles, which are not automated and do not have any wireless communication capability. Such situations where different types of vehicles exist and can potentially act as disturbances to each other are rarely considered. Thus, it is important to involve human drivers in the simulation to enable studies that consider these mixed traffic situations.

To the author's knowledge, there are currently no standards or guidelines for functional safety and testing in the area of C-ITS. While there are existing standards and guidelines in automotive industry, e.g. ISO 26262 standard [26], or *A Vision for Safety 2.0* [27] which is a guideline for safe deployment of automated vehicles in USA. The paper from *Nilsson et al.* [28] suggests that ISO 26262 cannot be applied directly to vehicles in cooperative systems, thus they have proposed a way to extend the standard and apply it to cooperative systems with platooning as an example.

Lastly, *surrogate safety measures* (SSM) are often used as indicators for safety in the context of traffic safety analysis. Motivated by the fact that accidents are not frequent and sometimes not even recorded correctly, SSM are measures of road safety that do not rely on accident records. Instead, SSM consider conflicts in traffic to measure traffic safety. Examples of SSM are time-to-collision (TTC), deceleration-rate-to-avoid-a-crash (DRAC), post-encroachment-time (PET), etc. This thesis focuses on using TTC and its extensions, i.e. *Time Exposed Time-to-collision* (TET) and *Time Integrated Time-to-collision* (TIT) [23]. These safety indicators were used in a platooning safety evaluation in [29], where the safety of a dedicated lane for platoon is investigated. TTC is used in **Paper IV** to analyse the safety of two CACC controllers in a scenario where cut-in occurs from an adjacent lane. Moreover, TTC, TIT, and TET are used in **Paper V** to analyse safety of the two CACC controllers in a highway cut-in scenario.

2.2 Modelling and simulation

As mentioned above, simulation provides a safe and cost-efficient alternative for studying systems, where working with actual systems would be costly, time-consuming, or dangerous. In transportation systems research, driving and traffic simulators are commonly used. Driving simulators are often used in human factors research, e.g. studies to understand human driver behaviours in specific situations, or studies to understand interaction between human drivers and newly developed systems. With the main focus on human drivers, more efforts are put on achieving realistic driving experiences. Thus the model of the ego vehicle[1] is usually more detailed than the surrounding vehicles in driving simulators. Moreover, details are important for the ego vehicle because it is usually the vehicle under test, or it has the system-under-test implemented inside.

Traffic simulators are used to study transportation systems on a larger scale compared to driving simulators, hence multiple vehicles in a larger area are usually considered. For instance, the goals of traffic simulation studies could be to estimate traffic congestions, calculate travel time, assess the impacts of new road infrastructures or new types of vehicles, etc. While studies in traffic simulators involve bigger road networks and multiple vehicles, the vehicles are normally modelled in less detail than a vehicle in driving simulators. There are two levels of abstraction that are common in traffic simulations. Firstly, macroscopic level, this approach model vehicles as flows similar to fluid streams. Secondly, microscopic level, this approach consider each vehicle individually, and *car-following models* are typically used to define car-following behaviours of the vehicles in simulation.

Network simulator are used to model wireless communication network in C-ITS—which is sometimes referred to as VANET (Vehicular Ad hoc Network). Communication nodes in network simulators are typically static, meaning the nodes are fixed in one location, but this is not the case for the communication nodes in C-ITS that may be vehicles. Therefore, movements of the vehicle nodes are often obtained from traffic simulators, either in real-time or using recorded trajectories such as in [30]. Consequently, combinations between network and traffic simulators are commonly used for modelling and simulation of C-ITS in [31, 32, 33, 34, 35], for instance.

Alternatively, other stand-alone simulation tools have also been considered for C-ITS research such as in [36]. Nevertheless, driving simulators are less involved in C-ITS simulations, and have been used by only a few researchers, e.g. [37, 38, 39, 40]. This implies that modelling and simulation of C-ITS rarely involves human drivers. Therefore, we proposed a simulation framework for testing and evaluation of C-ITS using combination of driving, network, and traffic simulators in **Paper II**. The simulation framework combines VTI's driving simulator with a combination of open-source network and traffic simulators, namely Plexe (Platooning Extension for Veins) [34].

[1]The term *ego vehicle* is often referred to the vehicle that is controlled by the person in driving simulator.

2.3 Cooperative Adaptive Cruise Control and platooning

Cooperative Adaptive Cruise Control (CACC) is proposed as an improvement to ACC, which is an advanced driver assistance system developed from CC. All three functions are categorised in *level 1* of the driving automation levels, which means the driver is responsible for monitoring the environment, and the functions only automate longitudinal or lateral control of the ego vehicle. CC only maintains constant speed set by the driver. ACC also maintains constant set speed as long as there is no vehicle closer than a distance defined by the driver; if there is a vehicle within the defined range, it uses the vehicle's forward-looking radar to estimate and maintain constant distance from the preceding vehicle. CACC uses the same concept with ACC, but CACC has access to information from wireless communication network. This information is usually received from a vehicle (or multiple vehicles) ahead of the ego vehicle; and sometimes it can be from road infrastructure devices. This additional information allow CACC-equipped vehicles to drive closer to each other with improved string stability, because the wireless communication eliminates delays that propagate through the vehicles' sensor systems. For instance, if five ACC-equipped vehicles are following each other and the first vehicle starts to brake, the second vehicle needs to sense the braking behaviour, process, and react to that by braking; only after the second vehicle performs the braking, the third vehicle can react to the braking. Consequently, there is a huge delay until the last vehicle can react to the first vehicle braking. On the other hand, this delay can significantly be reduced if all vehicles can exchange information using wireless communication, assuming the wireless network is in normal condition. Furthermore, the last vehicle can even start braking before the first one, if they are coordinating properly, e.g. future actions are communicated. Thus reducing the risk for rear-end collisions and improving longitudinal control stability in the group of vehicles. Detailed discussions about CACC concepts can be found in two publications by *Shladover et al.* [15, 22].

Platooning refers to a group of vehicles (*platoon*) following each other with short inter-vehicular distance. If the inter-vehicular distance is short enough, air-drag between vehicles is reduced, hence reducing fuel consumption for vehicles in the platoon; especially for heavy-duty vehicles as suggested in [41, 42, 43, 44]. One way of enabling platooning with the short inter-vehicular distance is to equip platooning vehicles (vehicles in the platoon) with CACC functions, which is the kind of platooning considered in this thesis. Therefore, only a subset of platooning applications is considered in this thesis, because platooning also include applications where the control of vehicles involved are automated in both longitudinal and lateral control [21]. This thesis considers only the case where all platooning vehicles including the leader are automated longitudinally, although a platoon leader can be driven manually by a professional driver and only the followers are automated.

Table 2.1: Examples of projects that demonstrated platooning on roads, and the characteristics of their demonstrations.

Name	Year	Type	Gap (meters)	Speed (km/h)
PATH(1986-present)	1994	Cars	4	n/a
	1997	Cars	6.5	96.5(60 mph)
	2017	Trucks	15	88.5(55 mph)
CHAUFFEUR2 (2000-2003)	2003	Trucks	6-12	n/a
KONVOI (2005-2009)	2009	Trucks	10	≈60-80
Energy ITS (2008-2012)	2013	Trucks	4.7,10,30	80
SARTRE (2009-2012)	2012	Mixed	6	90
GCDC (2011, 2016)	2011	Mixed	n/a	n/a
	2016	Mixed	25-30	60-80
ETPC[1](2016)	2016	Trucks		
Country/Region[2]				
Sweden			11-12.5 (0.5s)	80-90
Denmark			0.5 seconds	n/a
Southern Germany			11 (0.5s)	80
Northern Germany			22 (1s)	80
Belgium			25-30 (1-1.2s)	90
The Netherlands			15-29(0.7-1.3s)	80

Note that some projects have their gap definition only in terms of *time gap*.
mph = miles per hour; s = seconds.
[1] European Truck Platooning Challenge.
[2] The project drove through several country with different regulations

Throughout the past decades, platooning concepts have been demonstrated by many projects. Examples of these projects are listed in Table 2.1. Starting from 1994 in the US, California PATH project [45] demonstrated a four-vehicle platoon with automatic longitudinal control. This was later extended to an eight-vehicle platoon in 1997, also to demonstrate *automated highway systems* (AHS) concept [46]. Truck platooning was the latest demonstration by this project in 2017 on the I-66 highway in the US. In Europe, CHAUFFER II project (2000-2003) [47] demonstrated truck platooning, where longitudinal and lateral control of the following trucks were automated using combination of data from V2V communication and images. This was based on the concept [48] presented in the previous CHAUFFER I project (1996-1998). Later, KONVOI project (2005-2009) realized a truck platoon on a German highway in 2009 [49].

SARTRE project (2009-2012) [50] also concluded with a demonstration of a "road train" of four vehicles on a proving ground in Sweden, with a truck as the leading vehicle and the followers are passenger cars. In Japan, Energy ITS project (2008-2012) conducted two truck platooning experiments on a test track and an express way, mainly to investigate energy saving of platooning applications among other things. Grand Cooperative Driving Challenge (GCDC) was organized in 2011 [51] and 2016 [52, 53], both editions include platooning scenarios, where vehicles in the platoons are combinations of different trucks and passenger cars. In European Truck Platooning Challenge (2016) [17], semi-automated truck platoons drove on public roads to Rotterdam from Sweden, Denmark, Germany, Belgium, and the Netherlands. The results [17] reported on the processes and experiences: regarding obtaining the permits from authorities, the implementation, feedbacks from the truck drivers, and challenges that comes with each step. The participants include DAF Trucks, Daimler Trucks, Iveco, MAN Truck & Bus, Scania, and Volvo Group.

Among CACC and platooning research, testing and evaluation for efficiency is common in term of impacts on traffic flow, this is studied and reported, e.g. in [54, 55, 56]; there are also studies with respect to fuel consumption as mentioned above. Another big topic for CACC is string stability, which means the ability to maintain a stable vehicle string without amplifying errors backwards in the string. In other words, if a vehicle deviates from its desired states (with respect to the desired speed or desired inter-vehicular distance), CACC controllers that can maintain the string stability will ensure that this deviation is not amplified on the vehicles behind; and the deviation should be dampened instead. This is usually done from perspectives of control theories. Most studies assume ideal cases where all actors are identical in term of physical appearances and functionalities; and very few have considered external disturbances or failures. Furthermore, studies about safety in CACC and platooning are fewer than expected, as suggested in this recent survey [14].

2.4 Summary

This thesis contributes towards the following topics, which we identified as important but still lacking in the research literature related to testing and evaluation of C-ITS applications.

1. Involving human drivers in testing and evaluation processes.
2. Testing and evaluation of platooning applications with a focus on safety-related issues.
3. Consideration of mixed traffic scenarios, where mixtures of vehicles with different automation and communication capabilities share roads.
4. Influences from external disturbances and sensor failures on safety.

Chapter 3
Results

This chapter summarises five papers appended at the end of this thesis. A brief summary followed by contributions and findings of each paper are given below.

Paper I presents a view on cooperative driving, in comparison to automated driving and ITS. More specifically, it presents an attempt to understand the complexity of C-ITS. Some challenges in development and testing of C-ITS were identified. With respect to testing and evaluation, the main contributions of this thesis are twofold, as described below.

Firstly, a simulation framework for testing and evaluation of C-ITS applications is proposed in **Paper II**. CACC and platooning applications were chosen as the first set of applications to evaluate. Hence, use cases related to CACC and platooning were presented to ensure that most aspects of the applications are modelled in the simulation framework. This simulation framework is a combination of driving, network, and traffic simulators. Using the driving simulator in the simulation framework offers a perspective from driver's seat of a connected and automated vehicle. It also enables a human driver to control one of the vehicles in the simulation, hence the driver can interact with other connected and automated vehicles.

Secondly, this thesis contributes to the testing and evaluation of CACC controllers in two different highway cut-in situations, where a manually driven vehicle cut-in between platooning vehicles from *i*) an adjacent lane; and *ii*) an on-ramp. As mentioned before, the manually driven vehicle in this thesis is assumed to be a non-connected vehicle, meaning that the vehicle has no communication capability, and thus cannot communicate with other connected vehicles such as platooning vehicles. The cut-in from an adjacent lane scenario has been created in **Paper III**. Later, **Paper IV** uses TTC to analyse safety of the adjacent lane cut-in situation with a few participants in the study. Furthermore, **Paper V** considers the cut-in from an on-ramp at a merging point of a highway. Results from a driving simulator study with 39 participants were presented, and safety evaluation of CACC controllers in the cut-in situation using TTC, TIT, and TET is presented. In summary, the evaluations pointed out that the cut-in situations could be hazardous, with a few factors found to be correlated with collisions.

3.1 Summary of Paper I

This paper presented: *i*) a definition of C-ITS; *ii*) analysis of C-ITS from the perspectives of driver behaviours and platform architecture; and *iii*) challenges related to development and deployment of C-ITS.

At the time of writing **Paper I**, classification of automated driving functions in to the "levels of driving automation" was proposed by several organisations such as SAE, BASt, NHTSA, and VDA. As it is today, the one proposed by SAE [16] is the most accepted and widely used, even by NHTSA. Each level of driving automation are related to the complexity of the systems as each level defines limits and responsibilities of the automated tasks. For instance, a level 2 automated function controls acceleration and steering under some circumstances, but the driver still has full responsibility to monitor driving tasks and environment at all time. Thus developers know which requirements needed to be fulfilled when the complexity and responsibility are defined in relation to the levels of driving automation. This also facilitate testing and evaluation as creating test cases within a defined scope could be easier.

On the other hand, C-ITS has not been as clearly defined as automated driving. Actors in C-ITS need to rely on the quality of communication network and received information in order to have successful cooperation, interactions, or negotiations. These dependencies make C-ITS more complex and difficult to clearly define responsibilities. As an attempt to classify C-ITS applications in term of their complexity, three dimensions of C-ITS are proposed: *i*) the number of actors in the system; *ii*) the driving tasks; and *iii*) the scope of goals. They are illustrated in Fig. 3.1, where the complexity of C-ITS grow from the origin outwards.

In this paper, actors in the systems are assumed to be vehicles and infrastructures such as road signs, or traffic management centres. Regarding the number of actors in the system, this starts from two as we need at least two actors in C-ITS. Adding more actors to the system will increase the complexity as the C-ITS applications need to handle more interactions, and consider cases when it fails to communicate or when the other actors do not cooperate.

Furthermore, according to the driver behaviour models in [57], driving can be seen as problem solving tasks at three different levels: operational, tactical, and strategical. Tasks such as steering and control of acceleration or brake pedals are seen as operational. Operational tasks are usually the least complex. Tactical tasks involves short-term decision making, e.g. whether to change lane, cross intersection, etc. Strategical involves planning of the entire journey such as route choices, goal of the trip (e.g. whether to save fuel or reach the destination as fast as possible, etc.), for instance. Strategical tasks are typically the most complex among the three tasks as they require a lot of information. Therefore,

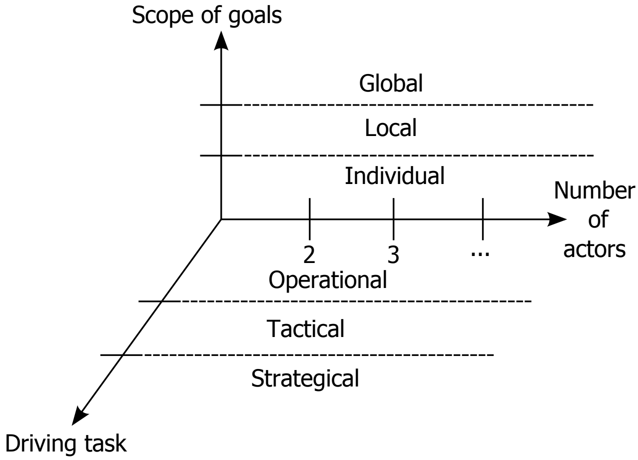


Figure 3.1: Dimensions of cooperative ITS

the complexity of the C-ITS applications will be different depending on which driving tasks they are taking care of. This can also depend on purposes of the applications, for example, functions with safety as the main goal might be more complex than the ones aiming at driver's comfort.

We suggest that the scope of goals can also effect complexity of C-ITS, and thus we proposed three levels of scope: individual, local, and global. The scope of goals in this context means the scope of actors that would benefit from a C-ITS application reaching its goals (a C-ITS application may have more than one goal). In other words, it can be called *the scope of benefits*. For instance, vehicles communicate with each other to make way for an emergency vehicle is a good example of individual benefits, where actors in the system cooperate to provide benefits to an individual actor; the emergency vehicle in this case. Local scope refer to a small area such as a platoon, an intersection, a merging point on a highway, etc. Finally, the global scope will give benefits to all actors in a city area, or much larger area such as a region or a country.

The benefits of defining these dimensions is that they can be used to define the complexity of a C-ITS application, i.e. with respect to which task(s) they are supposed to take care of, how many actors are involved, and how big is the scope of their benefits. Thus the requirements and constraints of a C-ITS application can be defined once the complexity is known. Consequently, testing and evaluation criteria can be clearly defined based on these dimensions.

Challenges towards deployment of C-ITS are also identified in the paper. Firstly, the challenges of providing sufficient communication coverage with reliability. Secondly, the interoperability issues need to be considered as most of the research are done with an assumption that vehicles are identical or capable of operating at the same level of driving automation. However, there will be a mix of different vehicles in real driving situations, in which they need to cooperate. Thirdly, the challenges regarding safety of the C-ITS applications. International Organization for Standardization (ISO) released ISO 26262 standard [26], which defines a functional safety standard for automotive electrical and/or electronics systems. In the standard, a procedure for hazard analysis and risk assessment, which result in Automotive Safety Integrity Level (ASIL) is proposed. However, it does not cover systems that involves V2X communication and there is no other such standard defined for C-ITS. Lastly, C-ITS introduce more complex scenarios and new possibilities, hence going through all of them is almost impossible. Therefore, a new methodology might be required to ensure sufficient testing and evaluation.

3.2 Summary of Paper II

A simulation framework—combining driving, traffic, and network simulators—was proposed in **Paper II** in order to create a unified simulation tool for C-ITS, which can involve a human driver and consider the traffic system and wireless communication. The **Paper II** is based on the previous work [13], where efforts have been put on creating the simulation framework and ensure that most aspects of C-ITS are adequately modelled. The previous version of the simulation framework can involve a human in the simulation loop only as an "operator", but not as a driver, i.e. the person can experience C-ITS applications from the driver's perspective but does not have an ability to control the vehicle. Thus this paper also presents the development of the framework in order to involve a human driver in the simulation via the driving simulator. The paper concludes that including the human driver in a simulation environment is important for testing and evaluation of C-ITS, especially during the transition period towards fully connected and automated transportation systems.

As mentioned above, the simulation framework consists of *i*) driving simulator; *ii*) network simulator; and *iii*) traffic simulator. It is specifically based on three existing simulation software:

1. VTI's driving simulation software.
2. Vehicle in Network Simulation (Veins) [32] (Plexe [34] version).
3. Simulation of Urban MObility (SUMO) [58] (Plexe [34] version).

VTI's driving simulation software is developed in-house at VTI. The software can be used for driving simulation in a desktop computer, or in one of the moving-base driving simulators owned by VTI.

As motivated in Chapter 1, CACC and platooning applications were selected as the first applications to be evaluated using the simulation framework. Therefore, Plexe (the Platooning Extension for Veins) [34] was chosen as the simulation tool for wireless communication network and surrounding traffic.

Plexe is based on a widely used network simulator and microscopic traffic simulator—Veins (Vehicle in Network Simulations) [32] and SUMO (Simulation of Urban MObility) [58]. To avoid confusion between the original version and the extension, *plexe-veins* and *plexe-sumo* will hereafter refer to the Plexe version of Veins and SUMO, respectively. Please refer to [34] for more details regarding Plexe.

Plexe provides a few implementations of CACC controllers according to their proposed designs. Among the existing CACC controllers in Plexe, two following CACC controllers were used:

1. the CACC controller proposed by *Rajamani* [59, Chapter 7]; this will hereafter be referred to as the *Rajamani* controller.

2. the CACC controller presented by *Ploeg et al.* in [60]; this will hereafter be referred to as the *Ploeg* controller.

These two controllers have different operating concepts, i.e. the *Rajamani* controller follows the constant distance gap concept, while the *Ploeg* controller uses the constant time gap approach (see Section 1.3.1 for more details about the concepts). Moreover, their control laws rely on different type and source of information as summarised in Table 3.1 below.

Table 3.1: Information used by the two CACC controllers and their sources.

Control parameters	**Controller**	
	Rajamani	*Ploeg*
Speed of platoon leader	V2V	-
Acceleration of platoon leader	V2V	-
Speed of preceding vehicle	V2V	Radar
Acceleration of preceding vehicle	V2V	V2V
Distance to preceding vehicle	Radar	Radar

All simulation software modules are written in the C++ programming language, and they are connected to each other using Transmission Control Protocol (TCP) connections as depicted in Fig. 3.2 below. The TCP connections are used for exchanging data between the simulators as well as the synchronisation between them. $\mathrm{TCP}_{\mathtt{TraCI}}$ is the name of the connection where *plexe-sumo* and *plexe-veins* used to exchange information following a standard protocol, called *traffic control interface (TraCI)* [61]: a protocol for on-line interaction with SUMO during simulation. This connection is used for several purposes such as the synchronisation between SUMO and Veins, sending information that Veins subscribed to, controlling of vehicle's behaviour in SUMO. The $\mathrm{TCP}_{\mathtt{sync}}$ connection is used for the synchronisation between the driving simulator and *plexe-veins*; and also for forwarding the subscribed data from *plexe-sumo* to the driving simulator. The TCP_{app} connection is used for data exchange between *plexe-veins* and the driving simulator. For more details, please refer to the appended **Paper II**.

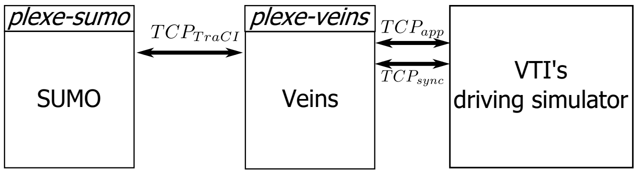


Figure 3.2: Connections between the three simulators in the simulation framework.

In the study cases of **Paper III**, **Paper IV**, and **Paper V** where a human driver is driving the ego vehicle, the responsibility for controlling vehicles in the simulation is distributed, as depicted in Fig. 3.3. Despite being controlled by different simulators, all vehicles appear in both simulation environments, i.e. the manually driven vehicle has its representation in *plexe-sumo*, and the platooning vehicles have their representations in the driving simulator.

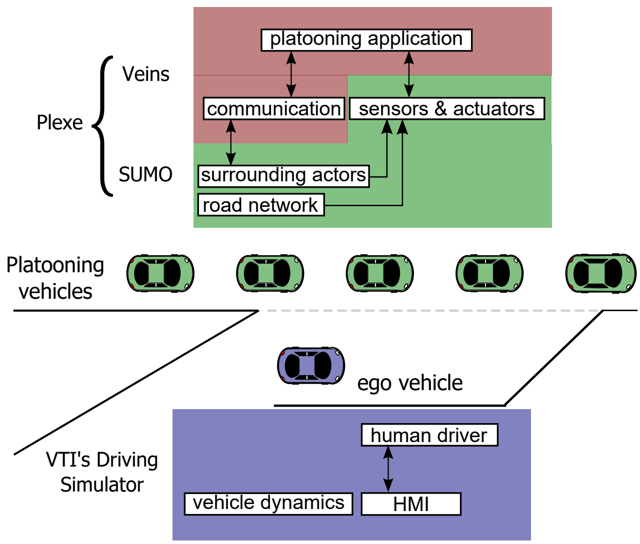


Figure 3.3: Controlling of vehicles in the situations when a human driver is controlling the ego vehicle: the platooning vehicles (marked in green) are controlled by *plexe-sumo*, and the ego vehicle (marked in blue) is controlled by the human driver in the driving simulator

Currently, the simulation framework uses Plexe version 2.0 (Plexe-2.0), which is based on Veins version 4.4 that supports OMNeT++ 5.0. The SUMO version used in Plexe-2.0 is based on SUMO 0.26.0. Moreover, the following change is made in *plexe-sumo*: the CACC activation range has been changed from 20 to 160 metres in order to ensure that the vehicle always uses the CACC controller. If the preceding vehicle is further than this range, ACC will be used instead. The VTI's driving simulation software is based on the 2016 version. The software with these versions and changes was used in the study in the **Paper V**. The latest version of *plexe-sumo* and *plexe-veins* used in this work can be found on the following GitHub repositories:

- `https://github.com/whatgit/plexe-sumo/tree/plexe-2.0-dev` (for *plexe-sumo*).
- `https://github.com/whatgit/plexe-veins/tree/plexe-2.0-dev-ds` (for *plexe-veins*).

3.3 Summary of Paper III

Paper III is the beginning of the analysis of highway cut-in scenarios by a manually driven vehicle. This paper reports on creating a highway cut-in scenario using the simulation framework. This scenario is inspired by the scenario used in the study [12] by *V. Milanés* and *S. E. Shladover*. This scenario is referred to as the *cut-in from an adjacent lane* scenario in this thesis. The scenario is a two-lane highway with platooning vehicles driving on the right-lane, then a manually driven vehicle cuts in between platooning vehicles from the left-lane as depicted in Fig. 3.4. According to the Fig. 3.4, the vehicle no.4 is the manually driven vehicle and the platooning vehicles are vehicle no.1,2, and 3. Speed of the platooning vehicles are 90 km/h in this study, while the manually driven vehicle starts at zero speed and have to catch up before performing a cut-in manoeuvre. This scenario is also used in **Paper IV**.

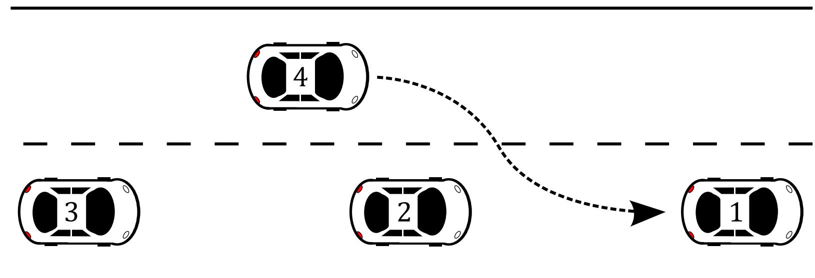


Figure 3.4: The *cut-in from an adjacent lane* scenario used in **Paper III** and **Paper IV**. The manually driven vehicle (no.4) cuts in between the platooning vehicles (no.1,2,3).

The experiments were conducted using the driving simulation software executed on a desktop computer. The desktop driving simulator is set up as depicted in Fig. 3.5. The two existing CACC controllers in *Plexe*, as mentioned above, were used with 17.5 metres desired inter-vehicular gap (0.6 seconds time gap[1]). After a few collisions were observed, more experiments were conducted at 30 metres inter-vehicular gap, where no cut-in collisions was observed. However, collisions occurred when the platoon leader accelerates after successful cut-in manoeuvres. Moreover, we observed hazardous situations caused by cut-in manoeuvres such as applying sudden brake at high deceleration rate (up to -4.5 m/s^2).

Although the CACC controllers were able to handle most situations, collisions and hazardous events were observed. Therefore, the paper concluded that CACC controllers—which were proven to be string stable—cannot be used directly in a mixed traffic scenario, and many aspects of cut-in need to be investigated further; especially safety. Large relative speed during the cut-in was identified as a cause of collisions. However, the paper concluded that *"having cut-in manoeuvres with such large relative speed on a highway is not common, however not impossible"*.

Figure 3.5: The desktop driving simulator setup.

3.4 Summary of Paper IV

This paper repeated the same experiment setup in **Paper III**: the simulated scenario was same as depicted in Fig. 3.4 and the same CACC controllers were used. Both 17.5 metres and 30 metres inter-vehicular gap were tested in this paper. Seven participants (all working at the School of ITE, Halmstad University) were asked to drive the scenario using the desktop driving simulator shown in Fig. 3.5 in order to obtain more diverse cut-in behaviours. Safety analysis of the simulated cut-in scenario using time-to-collision (TTC) was presented in

[1]The term time headway was written in the paper, but the authors actually refer to the time gap.

this paper. There was no collision in this experiment, despite observing a few occurrences of low TTC values.

The hypothesis of this study is that *cut-in manoeuvre will create hazardous situations for CACC controllers*. However, there were very few evidences in term of TTC. Hence the paper concludes that hazardous situations were not commonly observed according to the TTC values, and that both CACC controllers were able to perform well in this situation despite not designed to handle cut-in.

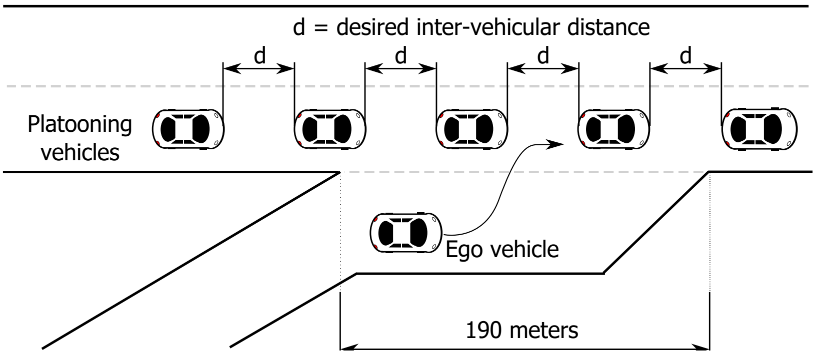
Furthermore, this paper mentioned that perhaps TTC is not the best safety measure for CACC operation. In contrast, TTC is proven to be useful later in the next paper. The reasons that lead to these contradictory conclusions will be discussed in Section 5.2 below.

3.5 Summary of Paper V

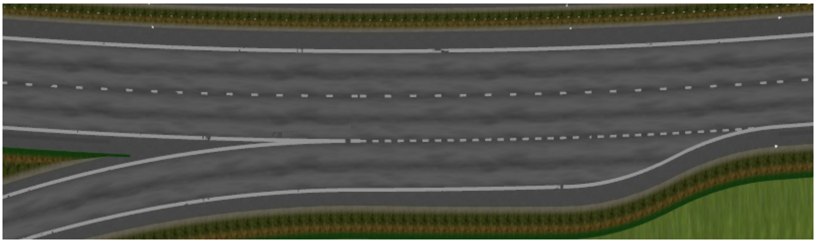
This paper investigated a cut-in situation at a merging point of a highway, known as the *cut-in from an on-ramp* scenario in this thesis. This scenario is illustrated in Fig. 3.6a. In the scenario, a manually driven vehicle (ego vehicle) starts from an on-ramp that merges into a two-lane highway where platooning vehicles are driving on the rightmost lane of the highway at 120 km/h (which is assumed to be the speed limit of the highway).

The manually driven vehicle was driven by 39 participants recruited from VTI's test drivers database. Among the participants, there were 21 female and 18 male, with an average age of 43 years old, and 33 of them had driven at least 10000 km per year. All participants completed the experiment, but we could not analyse the data from two participants due to technical errors during the experiment and data collection, thus the results from 37 participants were presented. Furthermore, instead of the desktop driving simulator used in previous work, this paper used the full-scale driving simulator, "Sim IV", located at the VTI's office in Gothenburg, Sweden. The driving simulator is shown in Fig. 3.7, and more details regarding the *Sim IV* can be found in [62].

The same CACC controllers used in previous studies were used, with four inter-vehicular gap settings: *i*) 15 metres; *ii*) 22.5 metres; *iii*) 30 metres; and *iv*) 42.5 metres. They are approximately 0.6, 0.8, 1, and 1.4 seconds time headway respectively. All in all, there were 8 different combinations between CACC controllers and the gap settings. Each participant drove all eight scenarios in a pre-defined order generated using the *balanced Latin Square* design, in order to prevent effects where the behaviours of participants are affected by the sequence of experiments. Moreover, as soon as each experimental run came to an end, all participants were asked to fill in a questionnaire regarding their perceived safety on all gap settings. Participants were also asked to respond to another questionnaire at the end, and this was followed by a brief discussion on the participants' experiences.



(a) Illustration of the scenario



(b) Top view of the merging road in the driving simulator

Figure 3.6: The *cut-in from an on-ramp* scenario used in **Paper V**

Figure 3.7: The "Sim IV" driving simulator used in **Paper V**.

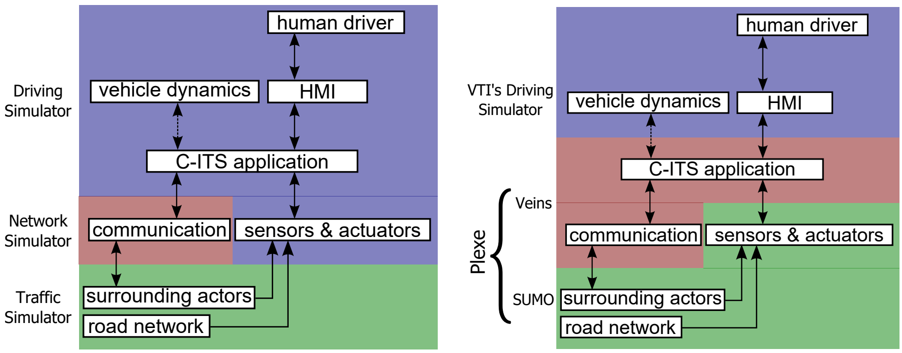
Of all 296 experiment runs, 69 collisions (23.31%) were observed when the participants cut-in between platooning vehicles. The paper evaluated the safety of CACC controllers in this situation with TTC and two other TTC-based safety indicators, namely Time Exposed Time-to-collision (TET) and Time Integrated Time-to-collision (TIT) proposed by *Michiel M. Minderhoud* and *Piet H.L. Bovy* in [23]. The paper concluded that the cut-in collisions are highly correlated with *i*) distance between the ego vehicle and the platooning vehicle following it (after the cut-in); and *ii*) difference between the speed of the vehicles at the moment the cut-in occurs.

The evaluation using TET showed that the *Ploeg* controller were less exposed to hazardous situations compared to the *Rajamani* controller. However, the TIT suggested that the situations experienced by the *Ploeg* controller were more severe.

Additionally, the paper concluded that it is necessary for CACC controllers to have a strategy for handling cut-in situations at merging points of highways. For instance, with extra sensors or procedures such as equipping vehicles with radar that looks to the side, increasing inter-vehicular gap at merging points, etc. With some assumptions, the paper suggested that all collisions observed in this study can be avoided by braking with a magnitude of -2.5 m/s^2 to the platooning vehicles, assuming that the vehicle coming from the on-ramp can be detected before it merges.

Chapter 4
C-ITS simulation framework

This thesis proposes a simulation framework for testing and evaluation of C-ITS applications. Considering simulation of C-ITS from the perspective of one vehicle, important models are identified as depicted in Fig. 4.1a. The actual implementation in this work is slightly different to the initial vision as illustrated in Fig. 4.1b. This chapter describes each model in the simulation framework and its implementation. Limitations and challenges in combining three different simulators are discussed, and the contributions of this thesis with respect to the C-ITS simulation framework are presented.



(a) The proposed setup from the perspective of one vehicle.

(b) The actual execution of the models, as proposed in **Paper II**.

Figure 4.1: C-ITS simulation models and simulators.

4.1 Simulation approaches

Figure 4.2 depicts the information flow specifically for platooning scenarios, when the ego vehicle is driven by a human driver in the driving simulator. If

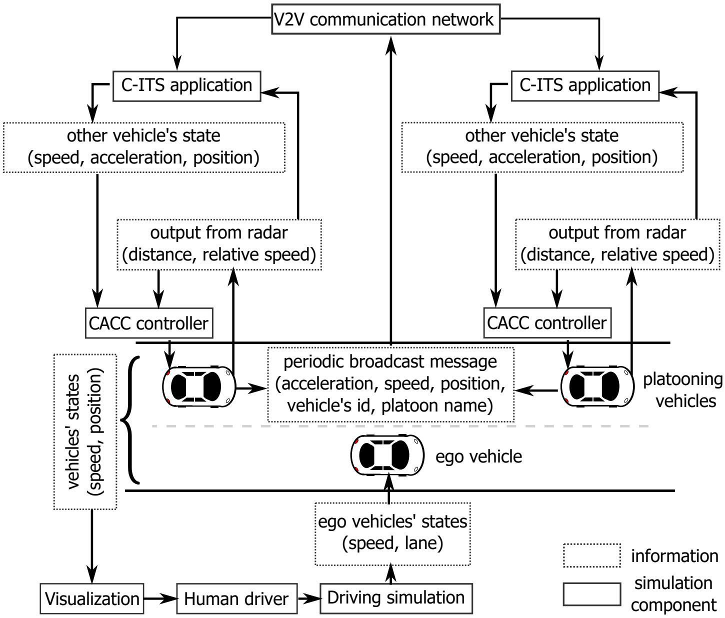
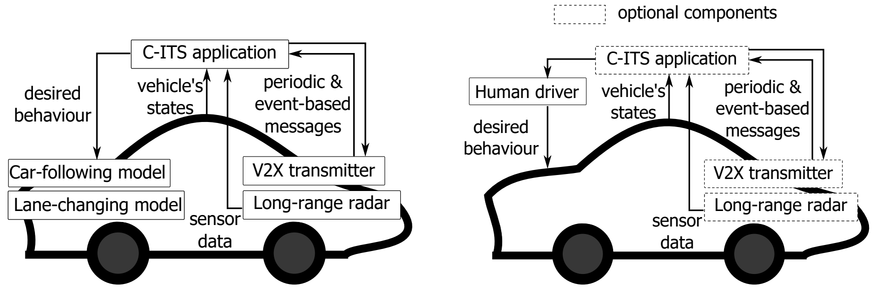


Figure 4.2: Flow of information between components in the simulation framework, when human driver is controlling the ego vehicle.

the human driver is not controlling the vehicle, the ego vehicle has same components as the platooning vehicles. Each platooning vehicle has its own C-ITS application component, which in this case is the platooning application. The information from each platooning vehicle is transmitted periodically at 10 Hz rate (every 0.1 seconds): this contains the acceleration, position, speed, vehicle identification number, and platoon name of that vehicle. Moreover, each vehicle is assumed to be equipped with a forward-looking radar that provides distance and relative speed to an object in front. These two sources of information are then gathered and processed by the C-ITS application. Relevant data such as states of the platoon leader and the preceding vehicle are sent to the CACC controller, which then regulates the speed of the vehicle.

If there is a human driver involved, the driver controls the ego vehicle via a driver interface in the driving simulator. Otherwise, the ego vehicle can be a vehicle that is a part of C-ITS and the person in the driving simulator could be there as a passenger or an operator of the vehicle. Each platooning vehicle is visualized to the human driver according to its states, i.e. position and speed. The ego vehicle's representation to the platooning vehicles is defined by its speed and lane in the driving simulator. Only speed and lane were used because of the limitation in the traffic simulator, which will be discussed in the Section 4.3 below.



(a) C-ITS vehicles. (b) Manually driven vehicles.

Figure 4.3: Components in vehicles in the simulation framework.

Figure 4.3 shows general components of vehicles in the simulation framework. For C-ITS vehicles such as platooning vehicles, their components are depicted in Fig. 4.3a. In case of the platooning vehicles in our studies, the C-ITS application component is the platooning application, and a CACC controller acts as the car-following model. All platooning vehicles are also assumed to be equipped with a V2X transmitter and a forward-looking radar. A lane-changing model defines lateral behaviour of the vehicles, although the platooning vehicles never change lane in our studies except for the platoon merging scenario in **Paper II**.

Figure. 4.3b illustrates the components for manually driven vehicles that are driven by human drivers. Most components in these vehicles are optional as human drivers have full control of the vehicles. In some use cases presented in **Paper II**, there is no manually driven vehicle. In other use cases such as in **Paper III**, **Paper IV**, and **Paper V**, there is only one manually driven vehicle. In these cases, the manually driven vehicle is assumed to have no V2X transmitter or C-ITS application. A forward-looking radar is assumed only for data measurement, i.e. it did not assist the driver in the studies. Nevertheless, the optional components can be included as parts of driver assistance systems, and thus allows flexibilities in future studies. For instance, a C-ITS application could be included to assist the driver, or a V2X transmitter can be equipped to the manually driven vehicle so that the driver could interact with other vehicles via wireless communication.

4.2 Simulation models

This thesis has aimed to make it possible to include all models in one simulation framework, although our main emphasis was not on all simulation models. The importance of each model with the selected abstraction levels in this thesis is described below. For some models, there are multiple simulators that can be used to execute the models: this is why this section also discusses our approach, i.e. why a specific simulator is selected for modelling each component. Please refer to Section 2.2 for a description of each simulator types considered in this thesis.

As shown in Fig. 4.1, the following models have been identified as being important to consider when simulating C-ITS.

C-ITS applications

Modelling of C-ITS applications is, of course, an essential part of C-ITS simulation. The original plan was to model these in a driving simulator. However, because C-ITS applications are required to communicate and interact with C-ITS applications in surrounding vehicles, driving simulators—which frequently offer no realistic modelling of wireless communication networks—are not ideal for modelling the C-ITS applications. On the other hand, network simulators are closely linked with the wireless communication network model and so they handle interactions between vehicles in the network, including the information that the vehicles are communicating, for example. Hence network simulators seem to be a more suitable option for implementation of the C-ITS application models. This is particularly true of *plexe-veins*, as it allows users to create their own applications that can be assigned to each individual vehicle, while message handling is dealt with by the simulator. Moreover, messages contain different information and data types specified by users, creating different packet sizes to be transmitted via the wireless communication channel.

Therefore, when modelling C-ITS applications in simulation, the environment for supporting interactions between the applications is an important factor to consider, in addition to their functionalities.

Human drivers

This is not exactly a simulation model, since the proposed framework can involve a real human driver in the simulation and all studies in this thesis use human drivers instead of driver models. Nevertheless, this can be replaced by a model of human drivers. In microscopic traffic simulators such as SUMO in particular, where each vehicle is modelled individually, the behaviour of human drivers is defined in two directions—longitudinal and lateral. The behaviour models for the longitudinal direction are often referred to as *car-following models*, which consider the speed selection and acceleration of vehicles. The lateral directions, i.e. lane change decisions, are described by means of *lane-changing models*.

Human-machine interfaces

Typically driving simulators include a form of human-machine interface (HMI) that allows drivers to interact with the ego vehicle, and vice versa. However, the test scenarios in this thesis included no driver interactions with surrounding vehicles via HMI, and information received was not displayed to drivers as the ego vehicle is assumed to have no wireless communication capability. Despite the option of including HMI in the driving simulator, including an HMI for test participants in the driving simulator was not considered a priority and so was not tested in this thesis.

Road networks

Basic modelling of road networks involves defining their geometry, i.e. lane width, length, slope, height, etc. In traffic simulators, speed limits and permitted directions of travel are typical additional factors used to describe road networks. Besides the aforementioned factors, road networks can be modelled in greater detail in respect of road conditions such as friction, materials, etc. These details are sometimes taken into account in road networks defined in driving simulators. Nevertheless, in this thesis, a road network in the *plexe-sumo* traffic simulator is used as a primary source. However, road networks are modelled in both driving and traffic simulators even though the human driver may not be driving the ego vehicle, because corresponding visualisation of road networks is also required in the driving simulator. As described in Fig. 3.3, the ego vehicle is controlled using the driving simulator and other surrounding vehicles are controlled by the traffic simulator. Ensuring accurate representation of the same road across different simulators, which do not use the same road formats, presents a challenge. This challenge will be discussed in Section 4.3 below.

Sensors and actuators

Another important aspect involves the modelling of sensors and actuators. These may have many different levels of detail, from simple to very sophisticated models. As regards sensors, this thesis assumes that each vehicle in the simulation is equipped with a forward-looking radar. There is an existing radar model in *plexe-sumo* which we used in this thesis. The radar in the traffic simulator measures distance and speed relative to the object in front in the lane currently occupied by the vehicle. There is no uncertainty in the measurement; and the range can be adjusted. This is a rather simple model of a radar system, but it is adequate for the platooning applications considered in this thesis. Realistic modelling of automotive sensor systems is a challenging research field in itself. Another motivation for using a simple model is to achieve real-time performance for the entire simulation framework. Greater detail and realistic models may enhance the validity of the simulation framework but may compromise its performance.

Regarding actuators, these are similar to sensors, that modelling realistic actuation of vehicles is a huge research field. This is not to be confused with the field of vehicle dynamics. One of the jobs of modelling actuators is to translate desired acceleration (due either to the driver pressing an accelerator pedal or to compliance with a control law in an automated function) into actual acceleration, taking into account factors such as engine, the limits of the vehicle, and so forth. With respect to this, *plexe-sumo* uses a first-order low-pass filter for modelling the actuation lag of vehicles, in order to imitate an engine. In the driving simulator, a vehicle dynamics model is used for the ego vehicle. This will be discussed in the vehicle dynamics section below.

Surrounding actors

Surrounding actors can be modelled in both driving and traffic simulators. In this thesis, surrounding actors are modelled using the traffic simulator *plexe-sumo*. The reasons for this are as follows. Firstly, *plexe-sumo* is already closely linked with the network simulator *plexe-veins*. Thus modelling surrounding actors in *plexe-sumo* also provides the option of making the actors "connected", i.e. equipping them with a wireless communication module and allowing them to be represented in the network simulator. Secondly, *plexe-sumo* has more diverse car-following models that already exist in the simulator, including the models for CC, ACC, and CACC. Moreover, a few lane-changing models are available in *plexe-sumo*.

That said, the behaviours of surrounding actors in VTI's driving simulation software are less diverse in normal driving situations. Their behaviour can also be specified, but this has to be done individually and conditions such as time or road position are required. This is more suitable for creating events such as a vehicle emerging from a junction as the ego vehicle approaches, a vehicle starting to brake hard after a certain simulation time, etc.

Vehicle dynamics

Vehicle dynamics is a field of study concerning how vehicles react to driver input, i.e. acceleration and steering, taking into account several factors such as the weight of the vehicle, centre of mass, characteristics of the vehicle's tyres, road friction, tyre-road interaction, wind resistance, etc. In driving simulators, vehicle dynamics are usually considered to provide a realistic driving experience as well as being an important field of research related to vehicle design. In this study, a simple vehicle dynamics model is used for the ego vehicle. A hardware-in-the-loop (HiL) simulator can be used in order to include a more sophisticated vehicle model that includes vehicle dynamics, e.g. in [10].

Wireless communication

Wireless communication is an important part of C-ITS, and all C-ITS applications rely on the quality of it. Therefore, realistic modelling of wireless communication is important. In this thesis, the wireless communication is modelled using the network simulator *plexe-veins*. In *plexe-veins*, the default network parameters are considered as shown in Table 4.1 below. Furthermore, users can define their own application-specific messages, which are of different lengths. In all studies in **Paper II**, **Paper III**, **Paper IV**, and **Paper V**, platooning vehicles transmit periodic messages at 10 Hz using the default network parameters; and although delays can be added as mentioned in [10], this was not considered in this thesis. Moreover, in the platoon merging scenario in **Paper II**, application-specific messages were added in order to simulate interaction between platooning vehicles. Once set up, the network simulator handles the transmission and receiving of messages during the simulation. These messages are then processed by the C-ITS applications implemented as referred to above.

Table 4.1: Network parameters in *plexe-veins*.

Parameter	**Value**
Path loss model	Free space ($\alpha = 2.0$)
PHY model	IEEE 802.11p
MAC model	1609.4 single channel (CCH)
Frequency	5.89 GHz
Bitrate	6 Mbit/s (QPSK R = $\frac{1}{2}$)
Access category	AC_VI
MSDU size	200B
Transmission power	20 dBm

4.3 Challenges and limitations

This section reports on the challenges and limitations involved in combining three different simulator types.

The first challenge considers synchronisation of the three different simulators. Veins, on which *plexe-veins* is based, consists of the network and traffic simulators in the framework in which they are already connected via a TCP connection. Between these two simulators, they are synchronised in a server-client fashion; in other words, *plexe-sumo* acts as a server and the *plexe-veins* acts as a client. As mentioned above, one can interact with SUMO simulation using TraCI messages. Therefore, to execute one time step in this framework, after *plexe-veins* connects to *plexe-sumo*, it send a TraCI message used for synchronisation to *plexe-sumo*, triggering *plexe-sumo* to execute one time step and return the subscribed data. After receiving the subscribed data, *plexe-veins* processes the data. Then this process is repeated until the end of the simulation at 100 Hz (0.01 second time step). Consequently, when the driving simulator is added to the simulation framework, a similar process is implemented with the driving simulator acting as the client and *plexe-veins* acting as the server for the driving simulator. This synchronisation is tested and reported in [6].

Secondly, lateral manoeuvres were considered in the simulation framework. This presented a challenge as vehicles may be anywhere in the simulation in the driving simulator. However, in the traffic simulator selected, this space is limited to the centre of a lane in road networks and so vehicles are always positioned in the centre of the lane. Moreover, lane changing in the traffic simulator is discrete, i.e. vehicles move to their new lane instantaneously when changing lane. Since the driving simulator also visualises vehicles in the traffic simulator, this lane-changing behaviour causes the driving simulator to visualise vehicles "jumping" from one lane to another. This issue is resolved visually in the driving simulation software. In other words, it displays smooth lane-change manoeuvres, instead of "jumps" when vehicles change lane in the traffic simulator. Nevertheless, vehicles in the traffic simulator still exhibit discrete lane-changing behaviours, and this is also applicable to the representation of the ego vehicle in the traffic simulator.

Thirdly, road networks have to be created in both *plexe-sumo* and the driving simulator. Although *plexe-sumo* seems to support data import and export from OpenDRIVE[1] format to the "SUMO format"—which is a description in XML format—this conversion was not working properly while the proposed simulation framework was being developed. Therefore, the road networks used in the studies were created manually on the basis of existing road networks in either of the simulators. Furthermore, the coordinate system for vehicle positions in the driving simulator follows the track coordinate system defined in the OpenDRIVE standard format, which describes the position in (s, t, h) coordinates. The letter s represents the longitudinal position along the reference line of the road, t describes the lateral position (positive to the left), and h indicates the elevation. On the other hand, SUMO uses a three-dimensional Cartesian coordinate system which describes the position in (x, y, z) coordinates in respect of the simulation "world". Hence SUMO coordinates are not dependent on any road. An alternative for representing the positions of vehicles is available in SUMO. This alternative represents positions using a longitudinal position from the start of the road and the lane number, which is very similar to the OpenDRIVE approach. However, this is based on the names of all road sections, which is still not synchronise between these two simulators at the moment.

Lastly, in SUMO, a gap in the adjacent lane is evaluated in terms of safety before the simulator executes a lane change manoeuvre because SUMO uses a collision-free model by default. This challenge arose while creating the platoon-merging scenario in [7]. The inter-vehicular gaps between platooning vehicles is shorter than usual, hence they were never regarded as safe by SUMO. Consequently, instead of merging into the gap between platooning vehicles, the merging vehicle ends up overtaking the platoon. This safety mechanism has been removed, allowing vehicles to change lane freely without being restricted

[1] `http://www.opendrive.org/`

by safety checks, and hence it was possible to create the scenarios in **Paper II**, **Paper III**, **Paper IV**, and **Paper V**.

4.4 Contributions

Including human drivers in the simulation is one of the important aspects to consider in C-ITS research using simulation. This thesis helps to enable this as presented above, while many proposed simulation tools used in C-ITS research lack this capability. Using human drivers in C-ITS simulation can provide several benefits. For example, data regarding human behaviours can be collected in order to improve or create new car-following models. Car-following models are commonly used as alternatives for modelling human behaviours that could be realistic in normal driving situations. However, behaviours of human drivers when they encounter and interact with automated vehicles or the C-ITS environment are not well-known as yet. Realistic car-following models for situations involving cutting in between platooning vehicles in particular, where inter-vehicular gaps are unusually small, may not be available. Creating new car-following models based on realistic data will contribute to future studies that use them. Another benefit as mentioned above is that the simulation framework provides a safe way in which to involve human drivers in dangerous situations, which may not be possible otherwise.

All appended papers in this thesis motivate, support, and present use cases where human drivers needed to be involved. **Paper I** elaborates on the importance to consider C-ITS from behavioural perspective, i.e. the perspectives of different drivers in *heterogeneous* traffic scenarios. It also suggest that interactions between actors are important to consider, especially in testing and evaluation of C-ITS. Thus **Paper I** presents demand upon methodologies which consider human drivers and their interactions for testing and evaluation of C-ITS. **Paper II** presents an improvement to the simulation framework so that a human driver can drive in the scenarios using driving simulation software. **Paper III**, **Paper IV**, and **Paper V** show applications of the simulation framework in evaluating highway platooning in cut-in situations involving a manually driven vehicle. These risky situations are the ones where it is important to involve a human driver using simulation.

The experiment in **Paper V** is the first time that the proposed simulation framework was used in a study involving a full-scale moving base driving simulator. Although motion[2] is available, it was not activated in this study so as to prevent motion sickness. Nevertheless, it has been shown that the simulation framework is able to work with an actual driving simulator, thereby permitting future C-ITS research in other domains such as studies in the field of human factors or relating to the human-machine interface.

[2]The driving simulator can be moved in two directions with pitch and roll motions, providing a more realistic driving experience.

Chapter 5
Safety analysis and evaluation of CACC controllers

Another main topic presented by this thesis relates to safety analysis and evaluation of CACC controllers. This chapter provides an overview of the field, specifically in relation to potential hazards when CACC controllers are used for highway platooning, and safety mechanisms that may be required. The contributions made by this thesis to the research field are presented at the end.

5.1 Hazards

As there is a short inter-vehicular gap between platooning vehicles, there is a high risk of collisions within the platoon and with vehicles surrounding it due to disturbances and failures. Hazards associated with platooning applications that may potentially lead to collisions are identified in [14]. In the author's opinion, these can be assigned to four categories: *i*) external disturbances; *ii*) internal component failures; *iii*) wireless communication; and *iv*) human factors. Despite all the hazards that could be considered in the proposed simulation framework, only external disturbance caused by conventional vehicles in cut-in situations is considered in this thesis.

External disturbances

Cut-in manoeuvres from surrounding traffic are one of the most prominent hazards according to [14]. This is an example of external disturbances; disturbances from outside the system. Other examples could include imperfect roads and extreme weather conditions or road blocks due to roadworks. Some of these external disturbances are normally encountered in real traffic situations. In this thesis, we focus on two different cut-in scenarios. We show that cut-in manoeuvres can cause hazardous situations, the results suggesting that CACC controllers used in platooning applications should have strategies for dealing with such situations.

Internal component failures

This category considers internal sensor failures in each platooning vehicle. The most important sensor for CACC controllers is a forward-looking sensor, which is usually a radar measuring the distance to the preceding vehicle and the relative speed of the vehicles. Failure to detect the preceding vehicle, or incorrect detection of it, could lead to a devastating collision. Besides radars, failures in computation components or actuators could result in the same consequences. Therefore, it is important to evaluate CACC controllers and analyse their safety under these failures conditions. Detailed models of sensors and components are required for analysis purposes. However, modelling sensors and all components in detail is not considered in this thesis. Nevertheless, this may be included in future work using the proposed simulation framework with HiL simulators, simulating the electrical architecture of vehicles, as presented in [10].

Wireless communication

Information from wireless communication is another important input with significant influence on the operation of CACC controllers. Improvements on ACC provided by CACC controllers are due to the information made available by wireless communication. Hence the operation of CACC controllers relies heavily on having the most recent information that is reliable and correct. Ensuring that the information has such properties, taking into account disruptions in wireless communication such as delays, packet loss, etc., is a huge research field in itself. Although the proposed simulation framework is capable of simulating an imperfect situation of this kind by manipulating parameters in wireless communication, it is not the main focus of this thesis, and is thus not studied in this work. Moreover, to be able to compensate for imperfections, CACC controllers should be aware of the situation: that the wireless communication is currently not at its peak quality of service.

Human factors

Human factors is another major aspect to consider. Hazards caused by neglecting human factors include accidents where a driver is unaware that a CACC controller is no longer in control, for example. Since CACC controllers are categorised as “level 1” in the levels of driving automation, drivers are still responsible for monitoring the driving environment and controlling of the vehicle. Short inter-vehicular distances between platooning vehicles can place stress on drivers, who have to monitor the driving at all times. A common concern in this area is the transition between automated and manual driving modes. Some studies have reported results relating to this issue, e.g. [63, 64, 65, 66].

5.2 Contributions

In this work, two CACC controllers were evaluated with regard to safety during platooning using the highway cut-in scenarios. This work contributes to the method for evaluating the controllers in cut-in situations by means of simulation. As also suggested in the Section 2.3, most reported evaluations of CACC controllers and platooning applications relate to string stability, traffic flow efficiency, and reduction of fuel consumption. Nevertheless, some attempts have been made in the last decade, as summarised in the survey by *J. Axelsson* [14]. Cut-in situations are one of the more common hazards referred to, according to the survey. The survey concluded that of the hazards identified in the literature, *"The most prominent hazards are related to the risk of a vehicle in the platoon running into the preceding vehicle, and another commonly mentioned risk is that caused by cut-ins from surrounding traffic."*.

Different reactions by CACC controllers to cut-in manoeuvres were observed. In **Paper III**, collisions were observed when the platoon's leader sped up when using the *Rajamani* controller, but not when the *Ploeg* controller was being used. This is because the control law of the *Rajamani* controller relies on information from the platoon's leader via V2V communication (see Table 3.1). Moreover, because we used the CACC controller as designed, there was no mechanism for dealing with the cut-in situations. Consequently, when the manually driven vehicle (which has no V2V communication capability) cuts in, the no.2 platooning vehicle (see Fig. 3.4) detects it using its radar but continues executing the platooning application. When the leader then increases its speed, the no.2 vehicle attempts to follow the leader and so collides with the vehicle that cuts in. Nonetheless, this depends on the "weight" assigned to each source of information in the controller design, which was not altered from the original implementation in Plexe.

On the other hand, the *Ploeg* controller was designed to control on the basis of information about the preceding vehicle, with only acceleration of the preceding vehicle obtained via V2V communication; the rest of the information is gathered from the radar, as summarised in Table 3.1. Although we did not observe any collision in our simulation, it could occur with other settings as long as the platooning application is not disabled. Therefore, as [12] suggests, assigning the platooning vehicle directly behind the non-connected cut-in vehicle as the new platoon leader for the platooning vehicles behind is recommended, followed by switching its controller to ACC.

The safety of cut-in situations is analysed in this work using TTC and its extensions (TET and TIT) as safety indicators in **Paper V**. In **Paper V**, the two CACC controllers were equally involved in collisions but at different gap settings. In the cut-in situations, TET suggests that the vehicles controlled using the *Rajamani* controller are more exposed to hazardous situations. However, TIT suggests that although the duration of exposure to hazard is less in vehicles controlled by the *Ploeg* controller, TTC values during these times are lower, indicating that the situations are more severe.

It seems that in most cases, both CACC controllers were able to handle the situations presented in **Paper III** and **Paper IV**, but not the situations in **Paper V**, where many collisions were observed. High collision rates were observed in **Paper V**, compared with much lower collision rates in **Paper III** (a few intended collisions) and **Paper IV** (none). The conclusion from **Paper V** is that collisions are highly correlated with the relative speed and the inter-vehicle gap to the manually driven vehicle (after cutting in). This conclusion also supports the use of TTC, TET, and TIT as these safety indicators consider the relative speed and distance between vehicles that are being evaluated.

Situations of this type, where the relative speed is high and results in a collision, were also observed in **Paper III**. However, **Paper III** suggested that "having cut-in manoeuvres with such large relative speed on a highway is not common, however not impossible", and that this issue should be investigated further. The same scenario was investigated in **Paper IV** and found only a small number of low-TTC values (using 3 seconds as the critical TTC value). The reason for this contradiction is due to the difference between the scenarios used in **Paper III** and **Paper V**. In **Paper III** and **Paper IV**, there were no other vehicles on the road in the two-lane highway scenario, and so drivers were able to take their time adjusting the speed of the ego vehicle and aligning the ego vehicle properly with the gap. On the other hand, the scenario in **Paper V** has limited space for cutting in as the acceleration lane is coming to an end, which forces the participants to make a decision in a shorter time compared to the other scenario. Therefore, according to the results presented in **Paper III**, **Paper IV**, and **Paper V**, cut-in situations at a merging point in **Paper V**, which are more time-critical, are more hazardous than the situations in **Paper III** and **Paper IV**, which are less time-critical.

Another aspect of safety evaluation that the **Paper V** has started to investigate is the perceived safety by other road users. As soon as each experimental run came to an end, all participants were asked to fill in a questionnaire regarding their perceived safety on all gap settings. Participants were also asked to respond to another questionnaire at the end, and this was followed by a brief discussion on the participants' experiences. Initial analysis of this data shows that participants clearly felt it was safer and more comfortable to cut-in with the 42.5-metre settings, and there was a tendency for them to feel unsafe and find it difficult to cut-in with the 22.5-metre settings. As human factors and user acceptance are not the main focus of this thesis, detailed analysis of these topics is suggested for the future work and will be discussed in Chapter 6. Nevertheless, perceived safety could be used as a guideline for defining a "proper" inter-vehicular gap between platooning vehicles. Consider the situations in **Paper V**, for instance: if the platooning vehicles are to create gaps and allow conventional vehicles to merge, we are aware that they may need to make the inter-vehicular gap larger than 22.5 metres.

5.3 Reflections

The highway cut-in test scenarios used in this thesis could be improved upon in terms of their realism. Firstly, let us consider the scenario used in **Paper III** and **Paper IV** involving cutting in from an adjacent lane. This scenario is still a fixed scenario, i.e. the manually driven vehicle always cuts in between the second and third vehicles in the platoon. Moreover, there were no other vehicles in the scenario. Improvements may involve extending this to form an *overtaking* situation, where test participants need to overtake the platoon from behind while surrounding traffic is potentially blocking the other lane. This is therefore a more likely scenario in a real traffic situation. Moreover, test participants then have less freedom to align themselves with the gap and adjust their speed, thereby potentially creating more diverse cut-in behaviours.

Similarly, surrounding traffic could be added to the cut-in from an on-ramp scenario in order to make it more realistic. The duration of the scenario could also be extended. The scenario currently ends at about 500 metres after the merging zone: extending this will allow us to observe participants' behaviours after cutting in, e.g. investigating whether they remain in the platoon after cutting in, how much time they spend between platooning vehicles, how many participants decide to change lanes and overtake the platoon, etc. Furthermore, the speed of platooning vehicles in the scenario, which is currently 120 km/h, can be reduced. Last but not least, the dimensions of the test road should be adapted to make them more realistic. The length of the on-ramp is currently 190 metres in the scenario in **Paper V**, but according to the Swedish Transport Administration [69] the recommended length for a road of this type is at least 250 metres.

Despite TTC and its extensions indicating that these are proper safety indicators for the cut-in scenarios, they require precise information for both vehicles under consideration and this may not be the case in reality. This was possible in the cases presented in this thesis, because simulation is used. Therefore, these safety evaluation indicators may not be applicable to other studies involving experiments with real vehicles. More work needs to be done on creating robust methodologies for safety evaluation.

Chapter 6
Conclusions

This thesis presents work relating to the testing and evaluation of C-ITS applications by means of simulation. The scope is then narrowed down to evaluation of the safety of highway platooning applications enabled by CACC controllers. In the scope of this work, six research questions and three thesis statements are listed in Chapter 1. Therefore, following a brief summary, this work is concluded by elaborating on the answers to the six research questions and ends with an indication of how the three thesis statements are supported. This is then followed by proposals for future research in this field.

6.1 Summary

This thesis proposes a simulation framework—combining driving, network, and traffic simulators—for testing and evaluation of C-ITS applications. The proposed simulation framework is able to involve human drivers in testing and evaluation processes, in addition to providing the models of C-ITS in sufficient detail. This introduces a new way of using simulators, in particular driving simulators. Typically, test participants use driving simulators to drive a vehicle being tested that is equipped with a new function that is also being tested. With the proposed framework and setup in this thesis, test participants drive a conventional vehicle together with the vehicles being tested (platooning vehicles). In this case, the conventional vehicle was disturbing the CACC controllers. This approach will be important for testing C-ITS applications and autonomous driving functions in the future.

Furthermore, we suggest that cut-in situations on highways are common and present an important hazard to platooning applications. This is one of the crucial scenarios to consider when evaluating safety of platooning applications. Hence we investigate two different highway cut-in situations: one from an adjacent lane, and another from an on-ramp. The results suggest that these situations are not always hazardous, cutting in from an adjacent lane being potentially less hazardous than cutting in from an on-ramp. We found that two important factors with a high degree of correlation with collisions in the case of the on-ramp approach are: *i*) distance between the ego vehicle and the platooning vehicle following it (after the cut-in); and *ii*) difference between the speed of the vehicles at the moment the cut-in occurs. Lastly, we suggest that safe highway platooning requires strategies for handling cut-in manoeuvres, as many collisions and hazardous situations may otherwise could occur as observed in the study in **Paper V**.

In summary, this thesis contributes to the following aspects of C-ITS research.

- Proposal of a simulation framework in which human drivers can be involved in the testing and evaluation of platooning applications.
- Further investigation of highway cut-in scenarios involving conventional vehicles, demonstrating how these scenarios can be set up in the proposed simulation framework.
- Presentation of safety evaluations of CACC and platooning applications in the highway cut-in situations.

6.2 Research questions

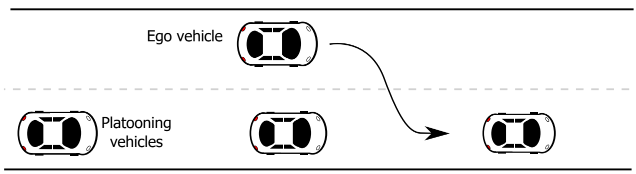
Let us commence with the first research question: **[RQ-1] What should be modelled and simulated for testing and evaluation of C-ITS applications?**

Paper I suggests that C-ITS should be considered from both behavioural perspectives (driver behaviours) and structural perspectives (technical components). Hence including human drivers in testing and evaluation is as important as having adequate representations of C-ITS actors. Therefore, considering from one vehicle's perspective, we proposed that the simulation should consider the following: C-ITS application, human driver, human-machine interface (HMI), road network, sensors and actuators, surrounding actors, vehicle dynamics, and wireless communication. To include the aforementioned components in the simulation, three different types of simulators are therefore needed: a driving simulator, a network simulator, and a traffic simulator—VTI's driving simulator, Veins, and SUMO respectively. The proposed and actual setups are illustrated in Fig. 4.1.

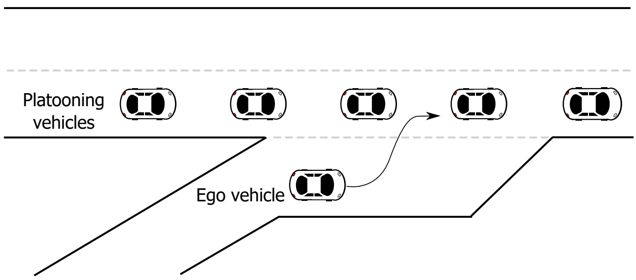
Therefore, important models to include in C-ITS simulation are identified as a response to **RQ-1**, and a simulation framework for testing and evaluation of C-ITS applications is proposed in **Paper II**. Furthermore, the simulation framework is able to consider the three dimensions of C-ITS proposed in **Paper I**: *i*) the number of actors in the system; *ii*) the driving tasks; and *iii*) the scope of goals. The traffic simulator allows different types of actors in the systems as each actor is modelled individually with its own C-ITS application, thus C-ITS applications which operates on different driving tasks can also be considered. The driving simulator permits studies with more focus on a human driver and the driving tasks of a vehicle. Network simulators enable actors to interact with each other, hence interaction can be studied. Lastly, the scope of goals can be evaluated by modelling C-ITS using the proposed simulation framework.

After proposing the simulation framework, CACC was selected as a function to be tested, leading to the second research question: **[RQ-2] How should CACC controllers be evaluated in order to ensure safety?**

The answer to this research question is twofold. Firstly, we propose the use of the simulation framework presented in **Paper II**. Secondly, we suggest beginning the investigation with cut-in situations, which are one of the most common hazardous situations but have not been studied thoroughly; in particular cutting in by conventional vehicles that have no automation and no wireless communication capabilities. This scenario was investigated in **Paper III**, **Paper IV**, and **Paper V** using the proposed simulation framework. Two highway cut-in scenarios were proposed and studied in this thesis: *i*) cutting in from an adjacent lane; and *ii*) cutting in from an on-ramp, which are illustrated in Fig. 6.1.



(a) *Cutting in from an adjacent lane*: a scenario where the conventional vehicle cuts in from an adjacent lane.



(b) *Cutting in from an on-ramp*: a scenario where the conventional vehicle cuts in from an on-ramp.

Figure 6.1: Illustration of the two highway cut-in scenarios considered in this thesis

Besides cutting in, many other factors and scenarios should be considered in order to ensure that CACC controllers operate safely. Of these, disruption in wireless communication, sensor failures, emergency braking, etc., are frequently discussed in research relating to safety of platooning.

[RQ-3] How can we assess the safety of highway platooning applications? This is an important research question in the context of platooning. Many attempts have been reported, as listed in [14]. However, more effort is required to address this properly as it continues to present challenges in the research field on account of the fact that many important scenarios are to be assessed in this regard. In this thesis, we focus on two different highway cut-in scenarios involving a conventional vehicle, as mentioned above. As regards these cut-in scenarios, we suggest that the safety of platooning applications in such scenarios can be assessed using time-to-collision (TTC) and its extensions, TET and TIT. This suggestion is also supported by the result from **Paper V**, where we found that relative speed and distance between vehicles are correlated with collisions in experiments.

[RQ-4] What defines safe CACC controller behaviours? As there are no standard definitions or guidelines for safe CACC controller behaviours as yet, we consider this definition from many different perspectives in order to define what could be regarded as safe behaviours for CACC controllers. First of all, from the point of view of the automated vehicle, the fact that the vehicle does not collide with anything is perhaps good enough. However, from the point of view of traffic safety in general, low TTC values and other *conflicts* according to the *safety surrogate measures* (SSM) in traffic may be considered unsafe.

From the perspective of other drivers, results from the questionnaire and discussions with participants in the experiments referred to in **Paper V** provide valuable feedback on this topic. The results do not represent *safe* behaviours exactly; rather, they represent *expected* behaviours from other drivers. This expands the scope to behaviours that are acceptable and not just safe, which is beyond the scope of this thesis. Nevertheless, these two topics are related to each other: in other words, it is important to have safe and acceptable behaviours with automated functions such as CACC. Thus social acceptance issues should be considered as important work to pursue in the future.

[RQ-5] Can a CACC controller handle all aspects of platooning alone? The answer to this question is no, particularly if we consider platooning applications that automate both longitudinal and lateral controls. However, the answer depends on the situation if the longitudinal dimension alone is considered. Under normal operating conditions with minimal communication disturbances, many CACC controllers have been proposed and mathematically proven to work under these conditions: see [60, 67, 68], for example. However, with the cut-in scenarios considered in this thesis, we have shown that different CACC controller behaviours were observed even though the situations are similar in **Paper III**, **Paper IV** and **Paper V**, and we can see that the controllers react differently to the situations. We conclude that a CACC controller alone as a single function, is insufficient to handle all aspects of platooning, especially when there are external disturbances.

This leads us to the next research question: **[RQ-6] How can CACC controllers used in platooning deal with disturbances caused by conventional vehicles driven manually?**

In this case, we consider only vehicles driven manually which have no wireless communication capability. One solution that has been proposed in the research literature [12] is that if a manually driven vehicle that has no V2V communication capability cuts into the middle of a platoon, the platoon should be split at the cut-in position as illustrated in Fig. 6.2. As the platoon does not know where the other vehicle will cut in, a robust algorithm is required to detect the cut-in manoeuvre, handle the split, and assign a new leader to the new platoon; and perhaps an additional protocol to reform the platoon when the manually driven vehicle has moved away. However, before reaching that decision point (whether or not the platoon should be split), we observed in **Paper V** that collisions can occur immediately after the other vehicle cuts in: there is very little time for the platooning vehicles to react.

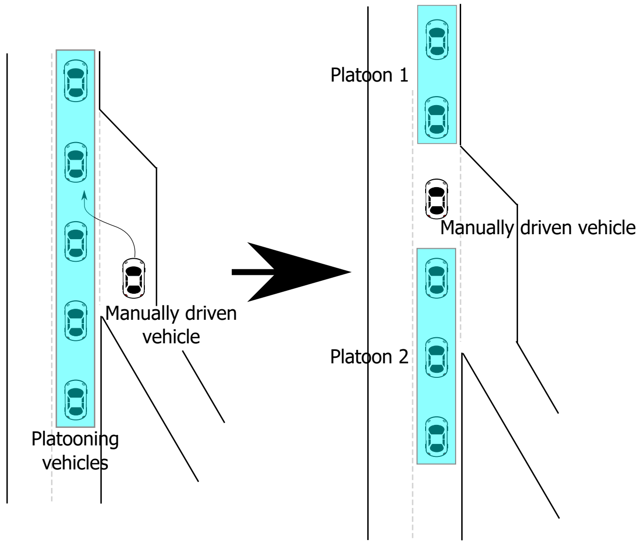


Figure 6.2: A method for handling vehicles cutting in, as suggested in [12].

Apart from splitting the platoon, a sensor (or multiple sensors) can be deployed at an on-ramp in order to detect oncoming vehicles and broadcast information via wireless communication. This would require additional sensors, and at least a wireless communication transmitter on a roadside unit. Moreover, platooning vehicles would need a protocol telling them how to react properly after receiving such information, e.g. whether or not the platoon should make a gap for a vehicle. Another alternative would be to have all platooning vehicles extending their inter-vehicular gaps when approaching merging points.

Such solution could improve safety, but it will compromise efficiency due to frequent acceleration and braking. Lastly, as we have shown in **Paper V**, collisions can be avoided if we assume that platooning vehicles can detect the manually driven vehicle and then adapt their speed accordingly. Regardless of the solution applied, all solutions would require more sophisticated functionality in the platooning vehicles so that they can identify which detected vehicles are relevant, thereby allowing them to adapt correctly. Moreover, the research literature includes no clear algorithms, guidelines, implementations, or protocols. We suggest that there has to be a way to deal with cut-in situations of this kind, such as by detecting that a vehicle is potentially cutting in at an earlier time and adapting to it. Although a number of strategies for handling such situations have been proposed as stated above, there are still no clear guidelines on how to implement them. The data collected from **Paper V** can be used as a baseline for assessing future improvements in order to judge whether these strategies would improve such situations.

6.3 Theses

[T-1] Safety of C-ITS applications can be evaluated by means of simulation.
Simulations have the advantage of being cost-effective and risk-free. This is particularly advantageous when evaluation of safety issues are the main concern, as the test scenarios could be dangerous and systems have to operate at their limits. Fidelity is arguably lower than with real experiments, such as experiments performed on proving grounds or in real traffic situations. Nevertheless, we suggest that simulation may be a valid and useful safety evaluation tool if adequate abstraction levels are taken into account for each model. Furthermore, the studies in **Paper IV** and **Paper V** show that simulation can be used to distinguish between a high-risk situation (cutting in from an on-ramp) and a low-risk situation (cutting in from an adjacent lane). A way of evaluating safety using safety indicators is also expanded upon in these papers.

[T-2] Testing and evaluation of C-ITS using simulation should include a way of involving human drivers.
A way of involving human drivers in testing and evaluation of C-ITS using an extended driving simulator is shown in this thesis. The importance of including a human driver is expanded upon in **Paper III**, **Paper IV**, and **Paper V**, where two different cut-in situations were investigated. Cut-in situations are regarded as one of the common hazards with regard to platooning, and we focused on in these in this thesis. One way of studying cut-in situations in detail is to use the proposed simulation framework that includes at least one human driver, such as the one proposed in **Paper II**. However, this is not the only test scenario that requires the involvement of a human driver. It will be necessary to study other test scenarios involving human drivers in the near future, when not all vehicles are fully automated. Therefore, we argue that it is important to have a way of involving human drivers in testing and evaluation of C-ITS.

[T-3] In the context of highway platooning, cut-in situations are hazardous and CACC controllers alone will be unable to handle such situations properly.

Results from the cut-in situation in **Paper III** and **Paper IV** did not suggest that the situation is hazardous. The results in **Paper V** suggested otherwise, however. Thus the first statement, *cut-in situations are hazardous*, is not well-supported, and it can be concluded that cut-in situations are not always hazardous. There are many factors that contribute to the hazardous consequences of cut-in situations. Nevertheless, this thesis confirms that a CACC controller alone is not enough to deal with cut-in situations properly, and hence additional functionalities are required for dealing with such situations.

6.4 Future work

Standards for functional safety, testing, and evaluation of C-ITS are still to be defined. As far as the author is aware, ongoing activities and related standards include the IEEE P2040 series of standards, which is supposed to be a "*Standard for Connected, Automated and Intelligent Vehicles*"; in particular IEEE P2040.2 entitled "*P2040.2—Standard for Connected, Automated and Intelligent Vehicles: Testing and Verification*". Apart from IEEE standards, ISO 26262 is a standard relating to the functional safety of electrical and electronic systems in vehicles. This is not directly applicable to the cooperative systems as suggested by *Nilsson et al.* [28]. Article [28] suggests an extension to the current standard and redefines some of the contents in order to address cooperative systems. This was then applied to platooning by way of example. According to ISO 26262, *severity*, *exposure*, and *controllability* are three factors contributing to risk that are used to determine the *Automotive Safety Integrity Level* (ASIL). TET and TIT, as used in this study, can be used to reflect on *exposure* and *severity* respectively. Thus future work could involve identifying a relationship between existing standards and these safety indicators; this could be important in more accurate definition of risk and systematic assessment of the safety of platooning.

Other future work may involve considering other functionalities that could co-exist in platooning vehicles. For example, in these particular test scenarios, an *automatic emergency braking system* (AEBS) may be triggered in the event of low TTC values. Therefore, cutting in could potentially cause unintentional full braking by platooning vehicles, consequently causing even worse situations. This may be beyond the scope of platooning applications, but it is important to consider these relationships with other functionality; i.e. how the application being tested operates together with other functions, particularly if that function will be common in future vehicles.

Solutions for dealing with cut-in situations have been proposed as referred to above, but how they should be implemented and executed in real situations is not clear. In future, implementing these solutions will demonstrate how they can be applied in real situations, and new challenges associated with them may be revealed. Moreover, assessing the benefits of these solutions is another important research for the future in this field.

As mentioned in [15], CACC is not primarily a safety system but it may improve safety. The main objectives for CACC development are to improve fuel efficiency and reduce traffic congestions. Regardless of the main objectives, safety should be guaranteed before such systems are deployed on public roads; and maintaining a larger inter-vehicular gap is one way of reducing risk. Therefore, it would be interesting in future to identify an optimal inter-vehicular gap in order to balance the efficiency and safety of CACC in a variety of situations.

Investigating acceptance by other road users began with **Paper V**, but the data collected needs to be analysed further. Furthermore, it is necessary to study acceptance of the "driver" in platooning vehicles. This can be done using the proposed simulation framework with the cut-in scenarios used in this thesis. Studies similar to [70, 71] can be carried out, involving CACC instead of ACC. The results of these studies will reflect on the trust and acceptance demonstrated by the driver and other road users towards CACC and platooning applications.

Lastly, as a tool for research, the proposed simulation framework also needs to be validated and developed further in order to address more realistic test scenarios. The validity of results is always questioned with regard to simulation studies, and so the simulation framework needs to be validated properly using data from the test track or real traffic scenarios, or comparing performance with a benchmark (if any), for instance. As regards development, vehicle manoeuvres in lateral dimension could be improved in the traffic simulator in order to have more realistic representations of surrounding traffic in the simulation framework. Furthermore, adding multiple human actors to one and the same simulation environment will provide a step forward for the simulation framework, since a fundamental aspect of C-ITS involves the interaction and cooperation of stakeholders.

References

[1] Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Cristofer Englund. Dimensions of Cooperative Driving, ITS and Automation. In *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pages 144–149. IEEE, 2015. (Not cited.)

[2] Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Arne Nåbo. A simulation framework for cooperative intelligent transport systems testing and evaluation. *Transportation Research Part F: Traffic Psychology and Behaviour*, 2017. (Not cited.)

[3] Maytheewat Aramrattana, Cristofer Englund, Jonas Jansson, Tony Larsson, and Arne Nåbo. Safety analysis of cooperative adaptive cruise control in vehicle cut-in situations. In *Proceedings of 2017 4th International Symposium on Future Active Safety Technology towards Zero-Traffic-Accidents (FAST-zero)*. Society of Automotive Engineers of Japan, 2017. (Not cited.)

[4] M. Aramrattana, T. Larsson, C. Englund, J. Jansson, and A. Nabo. Simulation of cut-in by manually driven vehicles in platooning scenarios. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 315–320, Oct 2017. (Not cited.)

[5] M. Aramrattana, C. Englund, T. Larsson, J. Jansson, and A. Nåbo. Safety Evaluation of Highway Platooning Under a Cut-In Situation Using Simulation. *IEEE Transactions on Intelligent Transportation Systems*, 2018. (submitted in August 2018). (Not cited.)

[6] Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Arne Nåbo. Extended Driving Simulator for Evaluation of Cooperative Intelligent Transport Systems. In *Proceedings of the 2016 Annual ACM Conference on SIGSIM Principles of Advanced Discrete Simulation*, SIGSIM-PADS '16, pages 255–258, New York, NY, USA, 2016. ACM. (Cited on page 35.)

[7] Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Arne Nåbo. Cooperative Driving Simulation. *The Driving Simulation Conference 2016 VR*, 2016. (Cited on page 36.)

[8] M. Aramrattana, J. Detournay, C. Englund, V. Frimodig, O. U. Jansson, T. Larsson, **W. Mostowski**, V. D. Rodríguez, T. Rosenstatter, and G. Shahanoor. Team Halmstad Approach to Cooperative Driving in the Grand Cooperative Driving Challenge 2016. *IEEE Transactions on Intelligent Transportation Systems*, 19(4):1248–1261, April 2018. (Not cited.)

[9] Maytheewat Aramrattana, **Raj Haresh Patel**, Cristofer Englund, Jérôme Härri, Jonas Jansson, and Christian Bonnet. Evaluating Model Mismatch Impacting CACC Controllers in Mixed Traffic using a Driving Simulator. In *29th IEEE Intelligent Vehicles Symposium (IV 2018)*. IEEE, 2018. (Not cited.)

[10] Maytheewat Aramrattana, Anders Andersson, Håkan Burden, Frida Reichenberg, and Niklas Mellegård. Testing Cooperative Intelligent Transport Systems in Driving Simulators. *The Driving Simulation Conference 2018 VR*, 2018. (Cited on pages 34, 35, and 40.)

[11] Patrizio Pelliccione, Avenir Kobetski, Tony Larsson, Maytheewat Aramrattana, Tobias Aderum, S Magnus Ågren, Göran Jonsson, Rogardt Heldal, Carl Bergenhem, and Anders Thorsén. Architecting cars as constituents of a system of systems. In *Proceedings of the International Colloquium on Software-intensive Systems-of-Systems at 10th European Conference on Software Architecture*, pages 5:1–5:7. ACM, 2016. (Not cited.)

[12] Vicente Milanés and Steven E. Shladover. Handling Cut-In Vehicles in Strings of Cooperative Adaptive Cruise Control Vehicles. *Journal of Intelligent Transportation Systems*, 20(2):178–191, 2016. (Cited on pages xi, 4, 23, 41, and 49.)

[13] Maytheewat Aramrattana. Modelling and Simulation for Evaluation of Cooperative Intelligent Transport System Functions, 2016. (Cited on pages 2 and 20.)

[14] J. Axelsson. Safety in Vehicle Platooning: A Systematic Literature Review. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1033–1045, May 2017. (Cited on pages 2, 6, 16, 39, 41, and 48.)

[15] Steven E Shladover, Christopher Nowakowski, and Xiao-Yun Lu. Using Cooperative Adaptive Cruise Control (CACC)to Form High-Performance Vehicle Streams. Definitions, Literature Review and Operational Concept Alternatives. *UC Berkeley: California Partners*

for Advanced Transportation Technology, 2018. Retrieved from https://escholarship.org/uc/item/3w6920wz. (Cited on pages 3, 7, 14, and 52.)

[16] SAE (On-Road Automated Driving (ORAD) committee). Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles, 2016. (Cited on pages 3 and 18.)

[17] Jacqueline Eckhardt. European Truck Platooning Challenge 2016 - Lessons Learnt, 2016. (Cited on pages 4 and 16.)

[18] 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, Feb 2014. (Cited on page 4.)

[19] Irene Cara and Jan-Pieter Paardekooper. The Potential of Applying Machine Learning for Predicting Cut-In Behaviour of Surrounding Traffic for Truck-Platooning Safety. In *25th International Technical Conference on the Enhanced Safety of Vehicles (ESV) National Highway Traffic Safety Administration*, 2017. (Cited on page 4.)

[20] S. Tsugawa, S. Kato, and K. Aoki. An automated truck platoon for energy saving. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4109–4114, Sept 2011. (Cited on page 4.)

[21] Carl Bergenhem, Henrik Pettersson, Erik Coelingh, Cristofer Englund, Steven Shladover, Sadayuki Tsugawa, and Magnus Adolfsson. Overview of platooning systems. In *Proceedings of the 19th ITS World Congress :*, pages 1–7, 2012. (Cited on pages 6 and 14.)

[22] Steven E Shladover, Christopher Nowakowski, Xiao-Yun Lu, and Robert Ferlis. Cooperative Adaptive Cruise Control: Definitions and Operating Concepts. *Transportation Research Record: Journal of the Transportation Research Board*, (2489):145–152, 2015. (Cited on pages 7 and 14.)

[23] Michiel M. Minderhoud and Piet H.L. Bovy. Extended time-to-collision measures for road traffic safety assessment. *Accident Analysis & Prevention*, 33(1):89 – 97, 2001. (Cited on pages 8, 12, and 27.)

[24] John C Hayward. Near miss determination through use of a scale of danger (traffic records 384). *Highway Research Board, Washington, DC*, 1972. (Cited on page 8.)

[25] Marcel Walch, Tobias Sieber, Philipp Hock, Martin Baumann, and Michael Weber. Towards Cooperative Driving: Involving the Driver in an

Autonomous Vehicle's Decision Making. In *Proceedings of the 8th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, Automotive'UI 16, pages 261–268, New York, NY, USA, 2016. ACM. (Cited on page 11.)

[26] ISO. ISO 26262 road vehicles – functional safety, 2011. (Cited on pages 12 and 20.)

[27] The U.S. Department of Transportation and the National Highway Traffic Safety Administration (NHTSA). A Vision for Safety 2.0., 2017. (Cited on page 12.)

[28] Josef Nilsson, Carl Bergenhem, Jan Jacobson, Rolf Johansson, and Jonny Vinter. Functional Safety for Cooperative Systems. In *SAE Technical Paper*. SAE International, 04 2013. (Cited on pages 12 and 51.)

[29] Md Sharikur Rahman and Mohamed Abdel-Aty. Longitudinal safety evaluation of connected vehicles' platooning on expressways. *Accident Analysis & Prevention*, 117:381 – 391, 2018. (Cited on page 12.)

[30] L. Codecá and J. Härri. Towards multimodal mobility simulation of C-ITS: The Monaco SUMO traffic scenario. In *2017 IEEE Vehicular Networking Conference (VNC)*, pages 97–100, Nov 2017. (Cited on page 13.)

[31] Michele Rondinone, Julen Maneros, Daniel Krajzewicz, Ramon Bauza, Pasquale Cataldi, Fatma Hrizi, Javier Gozalvez, Vineet Kumar, Matthias Röckl, Lan Lin, Oscar Lazaro, Jérémie Leguay, Jérôme Härri, Sendoa Vaz, Yoann Lopez, Miguel Sepulcre, Michelle Wetterwald, Robbin Blokpoel, and Fabio Cartolano. iTETRIS: A modular simulation platform for the large scale evaluation of cooperative {ITS} applications. *Simulation Modelling Practice and Theory*, 34:99 – 125, 2013. (Cited on page 13.)

[32] Christoph Sommer, Reinhard German, and Falko Dressler. Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. *IEEE Transactions on Mobile Computing*, 10(1):3–15, January 2011. (Cited on pages 13, 20, and 21.)

[33] M. Piórkowski, M. Raya, A. Lezama Lugo, P. Papadimitratos, M. Grossglauser, and J.-P. Hubaux. TraNS: Realistic Joint Traffic and Network Simulator for VANETs. *SIGMOBILE Mob. Comput. Commun. Rev.*, 12(1):31–33, January 2008. (Cited on page 13.)

[34] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler, and R. L. Cigno. Plexe: A platooning extension for Veins. In *2014 IEEE Vehicular Networking Conference (VNC)*, pages 53–60, Dec 2014. (Cited on pages 13, 20, and 21.)

[35] Y. Wang, J. Hu, and Q. Wang. A Simulation Study of Dynamic Route Guidance Based on Inter-Vehicle Communication. In *Communications and Mobile Computing (CMC), 2010 International Conference on*, volume 1, pages 397–403, April 2010. (Cited on page 13.)

[36] M. Tideman and M. van Noort. A simulation tool suite for developing connected vehicle systems. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 713–718, June 2013. (Cited on page 13.)

[37] Kugamoorthy Gajananan, Sra Sontisirikit, Jianyue Zhang, Marc Miska, Edward Chung, Sumanta Guha, and Helmut Prendinger. A cooperative its study on green light optimization using an integrated traffic, driving, and communication simulator. *Proceedings 36th-ATRF*, 2013. (Cited on page 13.)

[38] Helmut Prendinger, Marc Miska, Kugamoorthy Gajananan, and Alfredo Nantes. A Cyber-Physical System Simulator for Risk-Free Transport Studies. *Computer-Aided Civil and Infrastructure Engineering*, 29(7):480–495, 2014. (Cited on page 13.)

[39] Yunjie Zhao, Aditya Wagh, Yunfei Hou, Kevin Hulme, Chunming Qiao, and Adel W Sadek. Integrated traffic-driving-networking simulator for the design of connected vehicle applications: eco-signal case study. *Journal of Intelligent Transportation Systems*, pages 1–13, 2014. (Cited on page 13.)

[40] Y. Hou, Y. Zhao, A. Wagh, L. Zhang, C. Qiao, K. F. Hulme, C. Wu, A. W. Sadek, and X. Liu. Simulation-Based Testing and Evaluation Tools for Transportation CyberŰPhysical Systems. *IEEE Transactions on Vehicular Technology*, 65(3):1098–1108, March 2016. (Cited on page 13.)

[41] Michael P. Lammert, Adam Duran, Jeremy Diez, Kevin Burton, and Alex Nicholson. Effect of Platooning on Fuel Consumption of Class 8 Vehicles Over a Range of Speeds, Following Distances, and Mass. *SAE Int. J. Commer. Veh.*, 7:626–639, 09 2014. (Cited on page 14.)

[42] V. Turri, B. Besselink, and K. H. Johansson. Cooperative Look-Ahead Control for Fuel-Efficient and Safe Heavy-Duty Vehicle Platooning. *IEEE Transactions on Control Systems Technology*, 25(1):12–28, Jan 2017. (Cited on page 14.)

[43] A. A. Alam, A. Gattami, and K. H. Johansson. An experimental study on the fuel reduction potential of heavy duty vehicle platooning. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 306–311, Sept 2010. (Cited on page 14.)

[44] Assad Alam. Fuel-Efficient Distributed Control for Heavy Duty Vehicle Platooning, 2011. QC 20111012. (Cited on page 14.)

[45] S. E. Shladover. [path at 20Ůhistory and major milestones. *IEEE Transactions on Intelligent Transportation Systems*. (Cited on page 15.)

[46] R. Horowitz and P. Varaiya. Control design of an automated highway system. *Proceedings of the IEEE*, 88(7):913–925, July 2000. (Cited on page 15.)

[47] B. J. Harker. PROMOTE-CHAUFFEUR II amp;amp; 5.8 GHz vehicle to vehicle communications system. In *2001 ADAS. International Conference on Advanced Driver Assistance Systems, (IEE Conf. Publ. No. 483)*, pages 81–85, Sept 2001. (Cited on page 15.)

[48] H. Fritz. Longitudinal and lateral control of heavy duty trucks for automated vehicle following in mixed traffic: experimental results from the CHAUFFEUR project. In *Proceedings of the 1999 IEEE International Conference on Control Applications (Cat. No.99CH36328)*, volume 2, pages 1348–1352 vol. 2, Aug 1999. (Cited on page 15.)

[49] R. Ramakers, K. Henning, S. Gies, D. Abel, and H. M. A. Max. Electronically coupled truck platoons on German highways. In *2009 IEEE International Conference on Systems, Man and Cybernetics*, pages 2409–2414, Oct 2009. (Cited on page 15.)

[50] Tom Robinson, Eric Chan, and Erik Coelingh. Operating platoons on public motorways: An introduction to the SARTRE platooning programme. In *17th world congress on intelligent transport systems*, volume 1, page 12, 2010. (Cited on page 16.)

[51] E. van Nunen, M. R. J. A. E. Kwakkernaat, J. Ploeg, and B. D. Netten. Cooperative Competition for Future Mobility. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1018–1025, Sept 2012. (Cited on page 16.)

[52] J. Ploeg, E. Semsar-Kazerooni, A. I. Morales Medina, J. F. C. M. de Jongh, J. van de Sluis, A. Voronov, C. Englund, R. J. Bril, H. Salunkhe, Á. Arrúe, A. Ruano, L. García-Sol, E. van Nunen, and N. van de Wouw. Cooperative Automated Maneuvering at the 2016 Grand Cooperative Driving Challenge. *IEEE Transactions on Intelligent Transportation Systems*, 19(4):1213–1226, April 2018. (Cited on page 16.)

[53] C. Englund, L. Chen, J. Ploeg, E. Semsar-Kazerooni, A. Voronov, H. H. Bengtsson, and J. Didoff. The Grand Cooperative Driving Challenge 2016: boosting the introduction of cooperative automated vehicles. *IEEE Wireless Communications*, 23(4):146–152, August 2016. (Cited on page 16.)

[54] Hao Liu, Xingan (David) Kan, Steven E. Shladover, Xiao-Yun Lu, and Robert E. Ferlis. Impact of cooperative adaptive cruise control on multilane freeway merge capacity. *Journal of Intelligent Transportation Systems*, 22(3):263–275, 2018. (Cited on page 16.)

[55] 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, Dec 2006. (Cited on page 16.)

[56] W. J. Schakel, B. van Arem, and B. D. Netten. Effects of Cooperative Adaptive Cruise Control on traffic flow stability. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 759–764, Sept 2010. (Cited on page 16.)

[57] John A Michon. *A critical view of driver behavior models: what do we know, what should we do?* Springer, 1985. (Cited on page 18.)

[58] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker. Recent Development and Applications of SUMO - Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements*, 5(3&4):128–138, December 2012. (Cited on pages 20 and 21.)

[59] Rajesh Rajamani. *Vehicle dynamics and control.* Springer US, 2 edition, 2012. (Cited on page 21.)

[60] 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 *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 260–265, Oct 2011. (Cited on pages 21 and 48.)

[61] Axel Wegener, MichałPiórkowski, Maxim Raya, Horst Hellbrück, Stefan Fischer, and Jean-Pierre Hubaux. TraCI: An Interface for Coupling Road Traffic and Network Simulators. In *Proceedings of the 11th Communications and Networking Simulation Symposium*, CNS '08, pages 155–163, New York, NY, USA, 2008. ACM. (Cited on page 21.)

[62] Jonas Jansson, Jesper Sandin, Bruno Augusto, Martin Fischer, Björn Blissing, and Laban Källgren. Design and performance of the VTI Sim IV. In *Driving Simulation Conference*, 2014. (Cited on page 25.)

[63] M. H. Martens and A. P. van den Beukel. The road to automated driving: Dual mode and human factors considerations. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 2262–2267, Oct 2013. (Cited on page 41.)

[64] Christian Gold, Daniel Damböck, Lutz Lorenz, and Klaus Bengler. "Take over!" How long does it take to get the driver back into the loop? *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 57(1):1938–1942, 2013. (Cited on page 41.)

[65] Alexander Eriksson and Neville A. Stanton. Takeover Time in Highly Automated Vehicles: Noncritical Transitions to and From Manual Control. *Human Factors*, 59(4):689–705, 2017. PMID: 28124573. (Cited on page 41.)

[66] Natasha Merat, A. Hamish Jamson, Frank C.H. Lai, Michael Daly, and Oliver M.J. Carsten. Transition to manual: Driver behaviour when resuming control from a highly automated vehicle. *Transportation Research Part F: Traffic Psychology and Behaviour*, 27:274 – 282, 2014. Vehicle Automation and Driver Behaviour. (Cited on page 41.)

[67] J. Ploeg, E. Semsar-Kazerooni, G. Lijster, N. van de Wouw, and H. Nijmeijer. Graceful Degradation of Cooperative Adaptive Cruise Control. *IEEE Transactions on Intelligent Transportation Systems*, 16(1):488–497, Feb 2015. (Cited on page 48.)

[68] S. Öncü, J. Ploeg, N. van de Wouw, and H. Nijmeijer. Cooperative adaptive cruise control: Network-aware analysis of string stability. *IEEE Transactions on Intelligent Transportation Systems*, 15(4):1527–1537, Aug 2014. (Cited on page 48.)

[69] Trafikverket. *Krav för vägars och gators utformning*. 2015. ISBN: 978-91-7467-744-7, In Swedish. (Cited on page 43.)

[70] Annika F.L. Larsson, Katja Kircher, and Jonas Andersson Hultgren. Learning from experience: Familiarity with ACC and responding to a cut-in situation in automated driving. *Transportation Research Part F: Traffic Psychology and Behaviour*, 27:229 – 237, 2014. Vehicle Automation and Driver Behaviour. (Cited on page 52.)

[71] Bako Rajaonah, Françoise Anceaux, and Fabrice Vienne. Trust and the use of adaptive cruise control: a study of a cut-in situation. *Cognition, Technology & Work*, 8(2):146–155, Jun 2006. (Cited on page 52.)

Appendix A

Paper I
Dimensions of Cooperative Driving, ITS and Automation

Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Cristofer Englund

2015 IEEE Intelligent Vehicles Symposium (IV)
Seoul, South Korea, June 28 - July 1, 2015

(Link to the publication)

2015 IEEE Intelligent Vehicles Symposium (IV)
June 28 - July 1, 2015. COEX, Seoul, Korea

Dimensions of Cooperative Driving, ITS and Automation

Maytheewat Aramrattana[1,2], Tony Larsson[1], Jonas Jansson[2] and Cristofer Englund[1,3]

Abstract— **Wireless technology supporting vehicle-to-vehicle (V2V), and vehicle-to-infrastructure (V2I) communication, allow vehicles and infrastructures to exchange information, and cooperate. Cooperation among the actors in an intelligent transport system (ITS) can introduce several benefits, for instance, increase safety, comfort, efficiency. Automation has also evolved in vehicle control and active safety functions. Combining cooperation and automation would enable more advanced functions such as automated highway merge and negotiating right-of-way in a cooperative intersection. However, the combination have influences on the structure of the overall transport systems as well as on its behaviour. In order to provide a common understanding of such systems, this paper presents an analysis of cooperative ITS (C-ITS) with regard to dimensions of cooperation. It also presents possible influence on driving behaviour and challenges in deployment and automation of C-ITS.**

I. INTRODUCTION

With its potential benefits to the transport systems as presented in [1], cooperative intelligent transport system (C-ITS) have recently received a lot of attention. For example, in Europe three large projects that have been dealing with cooperative systems are CVIS, SAFESPOT, and COOPERS. Cooperative Vehicle-Infrastructure Systems (CVIS) [2] focused on vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication issues. While, SAFESPOT [3] aimed to enhance road safety via a "safety margin assistant" concept, which detects critical situations in advance, and the use of a "local dynamic map". COOPERS [4] was focused towards providing safety related real-time information and cooperative traffic management through infrastructure-to-vehicle (I2V) communication. A comparative study of these projects is presented in [5]. SARTRE [6], another European project dealt with platooning applications, through the concept of increased "driver comfort". Five years after the previous competition in 2011 [7], the grand cooperative driving challenge (GCDC) 2016 will be arranged by the i-GAME project [8]. The objective is to speed up real-life implementation of automated driving and interoperability of wireless communication. Besides i-GAME, another ongoing project is AutoNet2030 [9], working towards cooperative automated driving technology based on a distributed decision-making strategy.

[10] present an architecture for cooperative driving of automated vehicles. It consists of three layers *a*) vehicle control layer; *b*) vehicle management layer; and *c*) traffic control layer. The vehicle control layer typically is individual for each vehicle. It is connected to sensing and actuating systems, it sends data from sensors and vehicle state variables to the vehicle management layer. It also receives steering and vehicle speed commands from the vehicle management layer. The vehicle management layer is also implemented in the vehicles, placed in the middle between the vehicle control and the traffic control layer. It determines the movement of each vehicle in the C-ITS, with the data from the vehicle control layer of neighbouring vehicles through V2V communication. It also receives information from the traffic control layer via V2I communication. The traffic control layer consists of two parts; physical and logical. The physical part is located in the infrastructure, it consists of physical equipment like traffic signals, communication access and relay nodes, and roadside units. The logical part deals with regulations, rules, manners, common sense, and ethics in the human society. Considering the two parts, common criteria must be defined and communicated to neighbouring vehicles through the vehicle management layer.

Within C-ITS, information is shared between many actors such as vehicles, infrastructures, cloud services, etc. However, only sharing information is not enough to be considered a C-ITS, cooperation and interaction between the actors in the system is also required. In order to have a common understanding of what we mean by C-ITS, this paper presents an analysis of the topic from different perspectives in Section III. Introduction to driving automation is presented in Section II. Section IV elaborates on dimensions of C-ITS followed by its deployment challenges in Section V. Finally, Section VI conclude the paper.

II. LEVELS OF AUTOMATION

Recent research have focused on automated driving functions like adaptive cruise control (ACC), automated parking, etc. Levels of driving automation have also been defined by organizations such as SAE, BASt, NHTSA, and VDA. A comparison of these definitions is presented in Table I. Apart from ongoing research on automation functions like adaptive cruise control (ACC) and automated parking, several papers on cooperative systems in relation to automated driving concepts are published, see [10]–[13].

The following description will use SAE's definition as the basis. From level 0 to 2, the human driver has responsibility to monitor the environment. At level 2, the vehicle can take over steering, acceleration and deceleration in some driving modes. At level 3 and 4, the vehicle will monitor the environment, but only for some driving tasks. The differences

[1]School of Information Technology, Halmstad University, Box 823, 301 18, Halmstad, Sweden {maytheewat.aramrattana, tony.larsson}@hh.se
[2]the Swedish National Road and Transport Research Institute (VTI), SE-581 95, Linköping, Sweden jonas.jansson@vti.se
[3]Viktoria Swedish ICT, Lindholmspiren 3A, 417 56, Gothenburg, Sweden cristofer.englund@viktoria.se

TABLE I: Comparison between levels of automation released by SAE, BASt, NHTSA, and VDA.

	Organization			
Level	SAE	BASt	NHTSA	VDA
0	No Automation	Driver Only	No Automation	Driver Only
1	Driver Assistance	Driver Assistance	Function-specific Automation	Assisted
2	Partial Automation	Partial Automation	Combined Function Automation	Partial Automation
3	Conditional Automation	High Automation	Limited Self-Driving Automation	Conditional Automation
4	High Automation	Full Automation	-	High Automation
5	Full Automation	-	Full Self-Driving Automation	Full Automation

between 3 and 4 is the fall-back performance, in other words, who is responsible when the system fails. At level 3, the system still expect the human driver to handle the failure with a request to intervene. On the other hand, the vehicle will handle itself at level 4, for instance, when a failure occurs, the automation system still has to safely handle the vehicle. A request to the human driver may be made at this level, but if the driver does not respond, the system should be able to handle the situation. At full automation, level 5, the vehicle will handle all the driving responsibilities or tasks, including monitoring of the environment.

III. Cooperative ITS

In this paper, the scope of C-ITS encompass technical systems that applies to actors in the road transport system. Within this scope, C-ITS is defined according to the following definition:

Definition 1: C-ITS is a technical system that implements cooperative behaviour based on communication between two or more actors in the system.

Cooperative behaviour is in turn defined as:

Definition 2: A cooperative behaviour includes two or more actors working towards a common or mutually beneficial goal, purpose, or benefit; enabled by interaction and information exchange between the actors.

Cooperative behaviour involves actions such as sharing information, taking turns, following instructions from others, etc. Typical goals within the transport system context are, the improvement of safety and increased transport efficiency. When combined with driving automation, having more comfortable driving is another goal of C-ITS. The overall goal is to drive beyond the capability of a human driver or an autonomous vehicle. Thus, comfort as well as safety and efficiency are important goals. However, not every cooperative function must deal with all these goals. For example, a function like cooperative adaptive cruise control (CACC) can improve efficiency of the individual vehicles, but to be more efficient, vehicles could also drive closer to each other to reduce air resistance. This could however increase the risk of accident i.e. different goals can be in conflict with each other and may require different cooperative behaviours. Thus, applying the concept to the transport system needs to be considered carefully at different levels.

Apart from the exchange of system state information, interaction about intentions, planned behaviours, and agreements play an important role in C-ITS. In [14] cooperative driving is defined from a human-machine cooperation perspective, focused on the interaction between a vehicle and its human driver. They proposed five levels of cooperation for human-machine interaction. Those five levels deal with: *a*) *intention*; *b*) mode of *cooperation*; *c*) dynamic task and action *allocation*; *d*) the human-machine *interface*; and *e*) the *contact* between human and machine. Four out of the five levels were presented in the paper with an example of cooperative lane change scenario. Further evaluation of the concept was presented in [15].

A. Behavioural Perspectives

A critical review of different driver behaviour models is presented in [16]. According to the article, there are three levels of skill and control in driving, seen as a problem solving task: strategical, tactical, and operational. These three tasks relate to the driver's decision making and is often mentioned as basis for modelling of driver behaviour. The strategical level can be seen as a planning task, it involves things like cost and risk evaluation, route choice, trip goals. The tactical level is about deciding manoeuvres such as: overtaking, turning, gap adjustment based on the criteria made on strategic level. Moreover, negotiation is also involved at this level, for instance when making decision to cross an intersection, as well as monitoring of traffic since it is the basis for making decisions at this level. Lastly, the operational level handles more continuous and periodic routine tasks such as longitudinal and lateral control, based on environmental input. These are principles that any model of driver behaviour should take into account. Furthermore, information flow, switching and interaction between levels should also be considered. To bring this concept into C-ITS, the goal of cooperative functions could be on different levels but cooperative partners should have common goals on those levels to enable efficient cooperative behaviour.

As elaborated in [14] and [15], human-machine interaction is important for cooperative systems as long as the human driver is still involved in the driving task. Furthermore, at the early stage of C-ITS deployment, some vehicles might not have any communication and automation capabilities at all, some might have automation but not communication, and just a few would have both. How to communicate the intention between those three differently equipped categories of vehicles? How would the driving behaviour of autonomous vehicles be perceived and processed by the human driver in a manually driven car without communication? And vice versa. Those are important question from behavioural perspectives that needs to be addressed.

[17] investigated the effects of automation on tactical driving behaviour, depending on the trust in the system. Most

driving automation today works at the operational level, in which the function (when allowed to take over from the driver) handles longitudinal and lateral control, for example, ACC or lane keeping assist functions. On the tactical level, automation can be involved, e.g. in self-parking systems, but usually the vehicle only provides information to help the driver make decisions about driving tasks. Navigation systems are mentioned in [17] as one example of a function aiding strategic tasks, still it does not take control of the vehicle. Within C-ITS, automation of tactical tasks such as crossing of intersections is possible as presented in [11], [18], [19]. Furthermore, [20] elaborate on the possibility of having automation at the strategic or tactical level.

B. *Structural Perspectives*

From a structural perspective, actors in C-ITS consists of components aimed for: *a*) communication; *b*) sensor fusion; *c*) environment perception; *d*) decision making; *e*) actuators; and *f*) human driver interaction. Figure 1 illustrate these components from structural perspectives in relation to the driving tasks presented in section III-A.

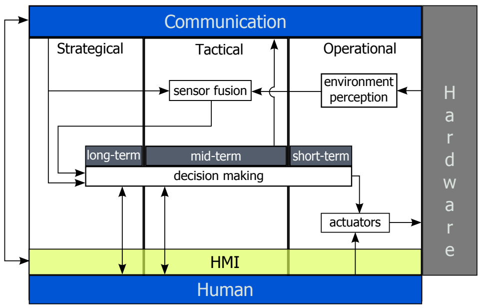


Fig. 1: Vehicle/infrastructure actors internal structure.

Access to one or more mechanisms for wireless communication is one of the key factors that enable C-ITS. Reliable and standardized communication techniques providing sufficient coverage and quality of service in different environments is an important enabler of C-ITS, and may eventually be achieved through a combination of vehicle-to-vehicle and vehicle-to-infrastructure (V2X) and cellular communication systems. ETSI [21], The European telecommunication standard institute, has published two technical specifications [22],[23], defining two types of messages namely cooperative awareness message (CAM) and decentralized environment notification message (DENM) respectively. These two message types are intended for the European C-ITS applications. CAM periodically provide information of presence, position and basic status of an ITS-station to neighbouring stations located within a single hop distance. Some use cases provided in [22] are: emergency vehicle warning, intersection collision warning, speed limit notification, collision risk warning, etc. If higher frequency is needed to ensure low reception latency after first contact, DENM with situation specific communication attributes can be used, e.g. providing road hazard warning related information. According to [23], examples of events that would trigger DENM are: collision risk warning, precipitation, road-work warning, accident, emergency electronic brake light, etc. In conclusion, these messages provide the basis for the communication protocols used in C-ITS. However, it still require some enhancements and extensions regarding application level information. For example, the need to include and standardize vehicle behaviour information as pointed out in [24]. Besides wireless communication, other alternatives such as light and sound signals from vehicles, messages on traffic signs, car horn, etc. are also included in this part. These alternatives are usually used to communicate intentions between cooperative partners as well as to interact with vehicles lacking wireless communication capability.

The sensor fusion part combines information from communication and environment perception. Usually this part implement a filter to certify information. The filter should detect false information and then ignore that information or inform the higher level about the failure. Thus, failures in sensing and communication are also handled in this part.

Environment perception or sensing could exist both in vehicles and infrastructures. Vehicles in C-ITS are usually equipped with sensors such as radar, camera, light detection and ranging (LiDAR), global positioning system (GPS), etc. A goal for this part is to perceive information about surroundings e.g. to detect and locate other vehicles, vulnerable road users (VRU), obstacles, cooperative partners, lane markings, etc. In case the vehicles maintain a local map of the surroundings, another goal is to locate itself in the environment through these sensors.

Decision making is an important part of C-ITS to select upon strategy and tactics of the systems based on the information gathered via sensors and communication. C-ITS can have either centralized decision making parts placed in the infrastructure or in a vehicle responsible for a group of vehicles. Alternatively, the decision making could be distributed and decentralized among vehicles and infrastructure. In a complex system both could be used at the same time in combinations such as distributed over the country but centralized within local areas. Decision making can be divided into short-term, mid-term, and long-term decision making. For instance, short-term decisions are, e.g. manoeuvres for collision avoidance, lane change, etc. They usually need information from the communication module in real-time, otherwise it could be dangerous to the system. For example, the driver receive the notification about a manoeuvre too late, and could not react in time, which might lead to an accident. On the other hand, route choice of a trip is an example of long-term strategic decision making.

At the highest level of automation, actuators, i.e. throttle, brake, and steering wheel, would be totally controlled by the system. However, at the lower level of automation, the human driver is still involved and have effects in this part as well. Especially at levels that are partly automated, the human drivers will have interactions with the decision making part via human-machine interface (HMI) possibly including haptic feedback by force on steering wheel. Moreover, according to the "convention on road traffic" from Vienna 1968 [25], which aim to set up international uniform traffic rules, *"every driver shall at all times be able to control his vehicle or to guide his animals"*. Thus, if the future policy will follow this rule, the human driver shall always have priority to decide and override the manoeuvre decided by the system.

Lastly, the human driver, interact with the system through its HMI. The human driver, responsible for all driving tasks, has the highest priority to decide and override the decisions from the automation system. Still, some systems such as advanced emergency braking system (AEBS), or anti-lock braking system (ABS), the system override the human driver's decision. By interactions between the human driver and the system through HMI, the driver will understand the intention of the system and vice versa. Moreover, having access to the communication part allows the driver to make requests to cooperative partners as well as to respond to requests. For example, as elaborated in [14], in the cooperative lane change scenario, although the request is initiated by the software function in a vehicle, the driver in another vehicle could decide, and confirm through HMI to the first vehicle that the request is accepted.

IV. Dimensions of Cooperative ITS

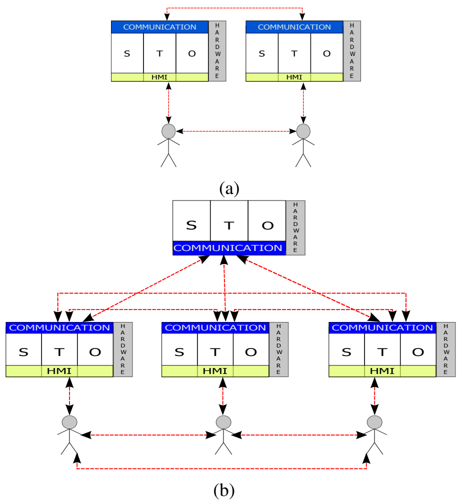


Fig. 2: Cooperative ITS with two and four actors respectively, interactions are indicated with red dotted lines.

Cooperation between two actors in C-ITS is illustrated in Fig. 2a with the red dotted lines representing possible interactions. The hardware box includes sensors, actuators, communication devices, computers, and user interfaces. There is usually at least one vehicle among the actors, and the other could be another vehicle or a road side unit. In case of cooperation between a vehicle and infrastructure, there is no interaction between the human operators. Cooperation between the vehicle and the infrastructure is usually aimed to assist the driver of the vehicle by providing extra information. In other words, it is typically a one-way communication from the infrastructure to the vehicle. Examples of use cases defined in [26] are: speed limit notification, traffic condition warning, point of interest notification, etc. The next step is when both actors are vehicles. This step includes more advanced scenarios such as cooperative lane change, motorway merging, intersection crossing, etc. Once the number of actors increase, the systems become more complex as illustrated in Fig. 2b. Moreover, if the vehicles are operating at different levels of automation, the interaction become even more complex.

The communication, which enables interaction and leads to cooperation, can be divided into three levels: *a*) human interaction; *b*) one-way wireless communication; and *c*) two-way wireless communication. Today, interaction between human drivers by means of conveying vehicle behaviour is performed via turn signal, vehicle horn, vehicle direction and position, etc. To enable interaction and communication within the transport system, drivers combine vehicle behaviour with eye contact and body language. Moreover, FM-radio sometimes acts as a road side unit providing warnings regarding traffic information. However, as of today, none of the above can communicate with automated vehicles. Thus, the one-way and two-way wireless communication provide channels to interact with and between automated vehicles. Normally, in one-way communication the warning would be sent to the vehicles and it depends on the driver or the automation system to react to the information. Hence, the action rely on the driving behaviour of the driver or the system. With two-way communication, more interaction such as acknowledgement and negotiation is possible. Therefore, it would be able to utilize the benefits of C-ITS.

Although complexity of C-ITS can be defined by many different factors as mentioned above, in this paper, the three dimensions considered important are: *a*) the number of cooperative actors; *b*) the driving task (planning horizon); and *c*) the scope of cooperative benefits.

Starting from two actors, which is the basis of C-ITS, adding more actors to the system will result in a more complex system as illustrated in Fig. 2b. Interaction between actors in the C-ITS is typically realized through wireless communication. More actors will require more reliable communication, more bandwidth and maybe even broader communication range. Furthermore, handling uncertainties created by the actors will also be an issue.

The cooperative function and its interaction behaviour also influence the complexity, depending on which kind of driving tasks it solves, i.e. operational, tactical or strategical. Moreover, depending on the function, different type of interaction is required. For example, CACC operates at the operational level and once the platoon is set up, no interaction between drivers is required unless there is a failure. On the other hand, cooperative lane change, which operates at the tactical level, requires interaction between vehicles and drivers at many different stages e.g. making lane change request, the driver determines situation and accept/reject the request, etc.

Another perspective is to classify the function based on its goal, i.e. comfort, economy, and safety. Comfort represents typical driver's assistance systems such as intelligent speed adaptation, traffic sign recognition, wrong-way driving warning, etc. Economy has the goal toward more efficient usage of road space and fuel consumption. For instance, CACC, platooning, highway merge function, etc. Lastly, functions with safety as the goal, usually have the highest complexity due to time and reliability constraints. For example, cooperative intersection collision avoidance systems (CICAS), cooperative lane change, etc.

Cooperative functions may sometimes fulfil two or more goals. The scope of these cooperative goals would have a significant impact on complexity. The scope is divided into three different levels: global, local, and individual. The global scope would give more priority to achieve better traffic flow and reduce congestion. For example, optimizing traffic flow of a whole highway. While, intersection, highway merging, cooperative lane change are examples at the local level. Although some "self-interested" agents could cheat and take advantage of cooperative systems as presented in [27], cooperation to make way for an emergency vehicle is a good example at the individual level.

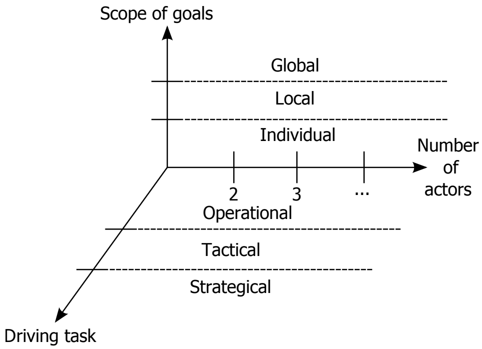


Fig. 3: Dimensions of cooperative ITS

Figure 3 illustrate three dimensions of cooperation. The complexity and the need for communication and cooperation grows as the system move away from the origin in any dimension. Automated driving functions help the C-ITS to achieve cooperation and vice versa.

V. Challenges

So far most research within cooperative driving and C-ITS deal with vehicles having the same level of automation. For instance, vehicles in the systems are all equipped with similar sensors, vehicles operate at the same level of automation, in other words, they have the same capabilities. On the contrary, considering real driving situations, cars with different capabilities are mixed in the traffic. There are many challenges already, even in current traffic situations, where driver's behaviour is a major difference. In the future traffic environment, where new and old cars meet, we would expect vehicles equipped with automated driving applications and communication facilities driving smoothly alongside older vehicles. Besides levels of automation, communicating vehicles from different companies, or different software version of the same cooperative function are other interesting scenarios. Seamless cooperation and interaction between such diverse vehicles are one of the challenges in C-ITS deployment. Apart from cooperation in the systems, interacting with non-cooperative vehicles, telling that there is cooperation, or automation going on, may sometimes be necessary. For example, using a special light signal to inform non-cooperative manually driven vehicles.

Safety is another important issue to address, which can be seen from many perspectives. First, perception failure or malfunction in a vehicle may mislead other vehicles in the system by feeding wrong information into the system. This issue is one example of hazardous events that could occur in the vehicle according to the standard ISO 26262 [28]. With more automation involved in the system as mentioned before, perception failure create risks which could lead to hazardous events. For instance, automated braking systems that suddenly brake the vehicle, or a lane keeping aid function that perform incorrect steering. Another perspective is safety, related to the transition to manual driving if the automation fails. For example, with an automated driving function like CACC, the driver might not always pay attention to the driving tasks. If the system fails and requires the driver to intervene in order to prevent hazardous events, one risk is that the driver is not alert enough to handle the situation. Thus, it could lead to an accident. There are some studies that already considered this issue, for example, [29] propose an architecture that separate applications into manageable and easy to test pieces and also use a communication protocol for collaborative vehicle control.

The larger the C-ITS becomes, the broader and more reliable communication coverage is needed. With such different capabilities as pointed out above and diversities among manufacturers, a standard set of rules are needed, especially in describing the behaviour of cooperative vehicles as elaborated in [24]. The paper used platooning, emergency vehicle warning and intersection scenarios as examples to illustrate lack of common abstraction to describe cooperative vehicle behaviour, for example, it is not clear in the current standard message format how the vehicles should manoeuvre.

One way of representing automation in cooperative driving is to apply the concept of multiagent system (MAS). Agents have suitable characteristics to represent actors in transport systems, which are autonomy, collaboration, and reactivity, as elaborated upon in [30]. Agent technology in traffic and transport systems are presented in [30]. Modelling and simulation are usually the main focus of agent-based applications. However, despite the long list of examples, only a few applications are implemented and deployed in real-world traffic. In conclusion, there are plenty of examples that relate cooperation with automation. Numerous promising simulation results were reported from those projects. Yet, only few were realized in real-world demonstrators. Thus, closing the gap between the simulation world and the real one is another challenge to be considered.

Last but not least, testing of cooperative driving functions is a challenging task within the area. C-ITS introduce numerous new possibilities and scenarios, testing all of them is nearly impossible. Therefore, ensure that "sufficient" tests have been done is seen as another challenge.

VI. Conclusion

This paper first presented C-ITS from two different perspectives: behavioural, and structural. From the behavioural perspective, C-ITS can operate or assist the driver on three levels of driving tasks: *a*) strategical *b*) tactical and *c*) operational. From the structural perspective, components within the actors in C-ITS, in relation to the behavioural perspective, are presented.

Moreover, the main factors to be considered for C-ITS classification are proposed as three dimensions of C-ITS: driving task, scope of goals, and number of actors. The scope of goals is the result of actors' behaviour and may be limited by the structure of the actors in the system. The number of actors reflects on connections and interaction complexity of the structures in the system. Besides, the relation between the driving task dimension, the behaviour perspective and the planning(time) horizon to solve the driving tasks, is seen as another perspective. Lastly, the challenges in the C-ITS deployment are also elaborated upon.

Interactions between cooperative partners and their drivers are a major contributor to successful cooperation. Especially automated coordination between vehicles with different capabilities is seen as one of the challenges.

In spite of the perspectives and challenges covered in this paper, implementing C-ITS is a great challenge with many issues. The presented analysis of C-ITS and dimensions of driving cooperation in relation to levels of automation is intended to provide a basis for future works regarding C-ITS.

References

[1] A. Stevens and J. Hopkin, "Benefits and deployment opportunities for vehicle/roadside cooperative ITS," 2012.

[2] "CVIS project." [Online]. Available: http://www.cvisproject.org/

[3] "SAFESPOT project." [Online]. Available: http://www.safespot-eu.org/

[4] "COOPERS project." [Online]. Available: http://www.coopers-ip.eu/

[5] G. Toulminet, J. Boussuge, and C. Laurgeau, "Comparative synthesis of the 3 main european projects dealing with cooperative systems (cvis, safespot and coopers) and description of coopers demonstration site 4," in *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*. IEEE, 2008, pp. 809–814.

[6] "SARTRE project." [Online]. Available: http://www.sartre-project.eu/

[7] J. Ploeg, S. Shladover, H. Nijmeijer, and N. van de Wouw, "Introduction to the special issue on the 2011 grand cooperative driving challenge," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, no. 3, pp. 989–993, 2012.

[8] "i-GAME project." [Online]. Available: http://www.gcdc.net/i-game

[9] "AutoNet2030 project." [Online]. Available: http://www.autonet2030.eu/

[10] S. Tsugawa, S. Kato, T. Matsui, H. Naganawa, and H. Fujii, "An architecture for cooperative driving of automated vehicles," in *Intelligent Transportation Systems, 2000. Proceedings. 2000 IEEE*. IEEE, 2000, pp. 422–427.

[11] G. R. de Campos, P. Falcone, and J. Sjöberg, "Autonomous cooperative driving: a velocitybased negotiation approach for intersections crossing," in *16th International IEEE Conference on Intelligent Transportation Systems*, 2013.

[12] R. Rajamani and S. Shladover, "An experimental comparative study of autonomous and co-operative vehicle-follower control systems," *Transportation Research Part C: Emerging Technologies*, vol. 9, no. 1, pp. 15–31, 2001.

[13] S. Kato, S. Tsugawa, K. Tokuda, T. Matsui, and H. Fujii, "Vehicle control algorithms for cooperative driving with automated vehicles and intervehicle communications," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 3, no. 3, pp. 155–161, 2002.

[14] M. Zimmermann and K. Bengler, "A multimodal interaction concept for cooperative driving," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 1285–1290.

[15] M. Zimmermann, S. Bauer, N. Lutteken, I. M. Rothkirch, and K. J. Bengler, "Acting together by mutual control: Evaluation of a multimodal interaction concept for cooperative driving," in *Collaboration Technologies and Systems (CTS), 2014 International Conference on*. IEEE, 2014, pp. 227–235.

[16] J. A. Michon, *A critical view of driver behavior models: what do we know, what should we do?* Springer, 1985.

[17] J. A. Hultgren, K. Kircher, and A. Larsson, "Tactical driving behavior with different levels of automation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, 2014.

[18] J. Kolodko and L. Vlacic, "Cooperative autonomous driving at the intelligent control systems laboratory," *Intelligent Systems, IEEE*, vol. 18, no. 4, pp. 8–11, 2003.

[19] J. Baber, J. Kolodko, T. Noel, M. Parent, and L. Vlacic, "Cooperative autonomous driving: intelligent vehicles sharing city roads," *Robotics & Automation Magazine, IEEE*, vol. 12, no. 1, pp. 44–49, 2005.

[20] C. Zinoviou, K. Katsaros, R. Kernchen, and M. Dianati, "Performance evaluation of an adaptive route change application using an integrated cooperative its simulation platform," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2012 8th International*. IEEE, 2012, pp. 377–382.

[21] "ETSI." [Online]. Available: http://www.etsi.org/

[22] "ETSI TS 102 637-2, Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Co-operative Awareness Basic Service," 2011.

[23] "ETSI TS 102 637-3, Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service," 2010.

[24] C. Englund, K. Lidström, and J. Nilsson, "On the need for standardized representations of cooperative vehicle behavior," in *Second International Symposium on Future Active Safety Technology toward zero-traffic-accident*. Nagoya, JAPAN: Society of Automotive Engineers of Japan, Inc., Sept. 2013, pp. 1–6.

[25] UNECE, "Convention on road traffic, of 8 november 1968," 1968.

[26] "ETSI TS 102 637-1, Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 1: Functional Requirements," 2010.

[27] R. Lin, S. Kraus, and Y. Shavitt, "On the benefits of cheating by self-interested agents in vehicular networks," in *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*. ACM, 2007, p. 183.

[28] *Road vehicles Functional safety*, ISO 26 262:2011, 2011.

[29] N. F. Maxemchuk, "Safety issues in collaborative vehicle control," in *Sarnoff Symposium, 2009. SARNOFF'09. IEEE*. IEEE, 2009, pp. 1–5.

[30] B. Chen and H. H. Cheng, "A review of the applications of agent technology in traffic and transportation systems," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, no. 2, pp. 485–497, 2010.

Appendix B

Paper II
A simulation framework for cooperative intelligent transport systems testing and evaluation

Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Arne Nåbo

Transportation Research Part F: Traffic Psychology and Behaviour
Available online: 31 August 2017

(Link to the publication)

ARTICLE IN PRESS

Transportation Research Part F xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Transportation Research Part F

journal homepage: www.elsevier.com/locate/trf

A simulation framework for cooperative intelligent transport systems testing and evaluation ☆

Maytheewat Aramrattana [a,b,*], Tony Larsson [a], Jonas Jansson [b], Arne Nåbo [b]

[a] School of Information Technology, Halmstad University, Box 823, SE-301 18 Halmstad, Sweden
[b] The Swedish National Road and Transport Research Institute (VTI), SE-581 95 Linköping, Sweden

ARTICLE INFO

Article history:
Received 30 December 2016
Received in revised form 1 July 2017
Accepted 9 August 2017
Available online xxxx

Keywords:
C-ITS
Driving simulator
Traffic simulator
Network simulator
Platooning

ABSTRACT

Connected and automated driving in the context of cooperative intelligent transport systems (C-ITS) is an emerging area in transport systems research. Interaction and cooperation between actors in transport systems are now enabled by the connectivity by means of vehicle-to-vehicle and vehicle-to-infrastructure (V2X) communication. To ensure the goals of C-ITS, which are safer and more efficient transport systems, testing and evaluation are required before deployment of C-ITS applications. Therefore, this paper presents a simulation framework—consisting of driving-, traffic-, and network-simulators—for testing and evaluation of C-ITS applications. Examples of cooperative adaptive cruise control (CACC) applications are presented, and are used as test cases for the simulation framework as well as to elaborate on potential use cases of it. Challenges from combining the simulators into one framework, and limitations are reported and discussed. Finally, the paper concludes with future development directions, and applications of the simulation framework in testing and evaluation of C-ITS.

© 2017 Elsevier Ltd. All rights reserved.

1. Introduction

During the past decades, information and communication technology have been incorporated into transport systems, i.e. intelligent transport systems (ITS), which aim for safer, and more efficient transport systems. During the same period, driving automation has been introduced to facilitate human drivers, because human errors are one of the major causes of road accidents. Driving automation starts from assisting human drivers in form of advanced driver assistance systems (ADAS). It has been classified into different levels of automation, the most commonly used is perhaps the ones proposed by Society of Automotive Engineers (SAE) (SAE International, 2014), although similar visions have been presented by other organizations such as National Highway Traffic Safety Administration (NHTSA), Federal Highway Research Institute (BASt), and German Association of the Automotive Industry (VDA). According to US Department of Transportation (2016), highly-automated vehicles are vehicles operating at SAE's driving automation level 3–5 and are responsible for monitoring the driving environment. Nowadays, several highly-automated vehicles can be seen both for private, and public uses. Cooperative intelligent transport systems (C-ITS), where actors in the systems are equipped with wireless communication modules for vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication, enabling interaction and cooperation between the

☆ This paper is an extension of the works originally reported in the Driving Simulation Conference 2016 and Proceedings of the 2016 annual ACM Conference on SIGSIM Principles of Advanced Discrete Simulation.

* Corresponding author at: School of Information Technology, Halmstad University, Box 823, SE-301 18 Halmstad, Sweden.
E-mail address: maytheewat.aramrattana@hh.se (M. Aramrattana).

http://dx.doi.org/10.1016/j.trf.2017.08.004
1369-8478/© 2017 Elsevier Ltd. All rights reserved.

actors in the systems. Especially, the wireless communication can serve as a way to communicate with highly-automated vehicles. Moreover, the wireless communication provides information beyond their own line-of-sight sensors to actors in the systems, which increase situation awareness of the actors, thus they can make better decisions based on access to external information. However, this increases the complexity of the systems, because to maximize the benefits, actors should be able to efficiently process and utilize the extra information. Furthermore, relying on external information not only make the systems more complicated, but also raises safety and security questions, e.g. *how to make sure that the received data are correct and up-to-date?* Therefore, to ensure that the C-ITS applications are able to cope with such situations, efficient ways to test and evaluate C-ITS are needed.

Computer simulation has been used to support research and development in many contexts, including transport systems and automotive engineering. It is a cost-effective and risk-free alternative for testing and evaluation of a system, complementing other test and evaluation methods. In the context of C-ITS studies, commonly used simulation tools are traffic simulators, driving simulators, and network simulators. Usually, combinations of the simulation tools are used, the most common approach is perhaps to use combinations between traffic and network simulators, such as in Rondinone et al. (2013), Sommer, German, and Dressler (2011), Piórkowski et al. (2008), Segata et al. (2014), Wang, Hu, and Wang (2010), Hikita, Kasai, and Yoshioka (2008). While human drivers are still expected to be involved in C-ITS, only a few research utilize driving simulator (Gajananan et al., 2013; Prendinger, Miska, Gajananan, & Nantes, 2014; Zhao et al., 2014) to involve human drivers. Driving simulators are feasible tools for studying human factors, which are normally missing from using only combinations of traffic and network simulators. On the other hand, with respect to C-ITS testing and evaluation, driving simulators normally have the following limitations—which are easily handled by traffic and network simulators: (a) they usually do not support realistic modelling of V2X communication; and (b) surrounding vehicles in driving simulators are often simplified and take almost no consideration of other vehicles. Thus, to simulate C-ITS, a combination of all three types of simulators is needed to execute simulation models of C-ITS, as illustrated in Fig. 1. The models are considered from one vehicle's perspective.

Therefore, this paper presents a simulation framework, consisting of driving-, traffic-, and network-simulators. The driving simulation software from the Swedish National Road and Transport Research Institute (VTI) is integrated to the traffic and network simulators, namely *Platooning Extension for Veins* (Plexe) (Segata et al., 2014). Details regarding each simulator are described in Section 2. The simulation framework is presented in Section 3. Section 4 shows application of the simulation framework on cooperative adaptive cruise control (CACC) scenarios, as example use cases. Remaining challenges and limitations, as well as future work, are presented and discussed in Section 5. Finally, the paper concludes in Section 6.

2. Background

To support simulation of wireless communication and more complex traffic models in driving simulation, this paper extends an existing driving simulation software from VTI with Plexe (Segata et al., 2014); a traffic and network simulator mainly aimed for simulation studies of platooning scenarios. For the traffic part, Plexe extends Simulation of Urban MObility (SUMO) (Krajzewicz, Erdmann, Behrisch, & Bieker, 2012), an open source microscopic traffic simulator. Furthermore, the net-

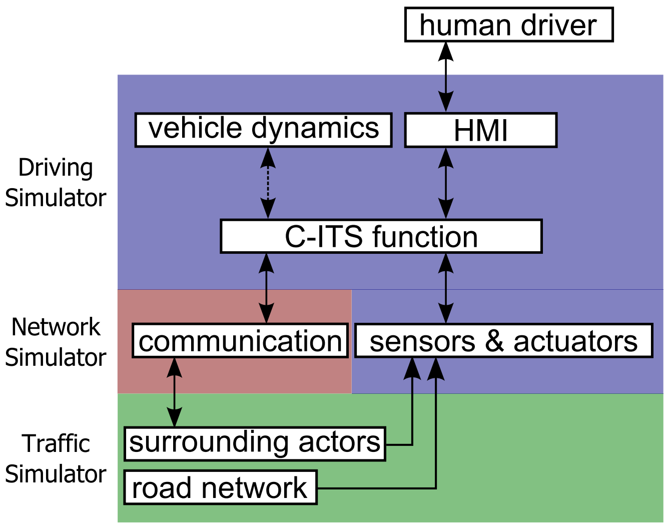


Fig. 1. Simulation tools and their responsible models in the context of C-ITS simulation, from the perspective of one vehicle in the systems.

Please cite this article in press as: Aramrattana, M., et al. A simulation framework for cooperative intelligent transport systems testing and evaluation. *Transportation Research Part F* (2017), http://dx.doi.org/10.1016/j.trf.2017.08.004

work simulator part is based on Vehicles in Network Simulation (Veins) (Sommer et al., 2011), a vehicular network simulator, which has been built on top of a discrete-event simulator, OMNeT++ (Varga & Hornig, 2008). More details about each simulation framework are described below.

Fig. 2 illustrates the current state of the simulation framework. The driving simulation software from VTI—VTI's software in Fig. 2—is responsible for interacting with human driver through human–machine interface (HMI), and vehicle dynamics models of the ego vehicle.

2.1. VTI's driving simulator

The VTI's driving simulation software is implemented in C++ and has been developed in house at VTI. There are three main modules; VISIR—the graphics rendering, SIREN—the sound software, and CORE—the kernel software running the main simulation loop, the vehicle dynamics model, scenario description, and cabin interface and HMI software. The software is designed for distributed simulation and can be executed either in desktop environment or in simulators with motion system at VTI such as Sim IV (Jansson et al., 2014). In this work, the software has been executed only in a desktop environment.

2.2. Plexe

As aforementioned, Plexe consists of means for traffic and network simulation, namely *plexe-sumo*, and *plexe-veins*. The latest release of *plexe-sumo* is based on SUMO version 0.29, however this work is based on the old version of *plexe-sumo*, using SUMO version 0.22. In this version, besides existing car-following models, cruise control (CC), adaptive cruise control (ACC), and cooperative adaptive cruise control (CACC) are available as car-following models. Two CACC controllers are available in Plexe: (a) the CACC controller from *Rajamani's book* (Rajamani, 2012, Chapter 7), referred to as *Rajamani* in this paper; and (b) the CACC controller proposed by Ploeg, Scheepers, van Nunen, van de Wouw, and Nijmeijer (2011), referred to as *Ploeg* in this paper. Microscopic traffic models implement a driving behaviour for each vehicle using a car-following model, which is normally differential equations regulating acceleration of the vehicle. The car-following models normally allow some degree of randomness, i.e. non-deterministic behaviour, but the models used in this paper are intended to simulate vehicles in automated driving mode. Therefore, surrounding vehicles in this work have deterministic driving behaviour. Moreover, to model the behaviour of the powertrain in vehicles, i.e. how vehicles accelerate and decelerate, a first order low-pass filter is applied on the desired acceleration of the vehicles in *plexe-sumo*, as shown in Fig. 3. In this case, the desired acceleration is calculated by CC model in *plexe-sumo*, as described in Rajamani (2012, Chapter 5).

The network simulator used in Plexe is based on an open-source vehicular network simulation framework, Veins. It is built on top of an event-based network simulator, OMNeT++. Veins interacts with SUMO using a Traffic Control Interface (TraCI) (Wegener et al., 2008) to obtain movements of the communication nodes, i.e. vehicles. TraCI is also used to control vehicles in SUMO, as well as for synchronization between SUMO and Veins. TraCI is implemented using a transmission control protocol (TCP) connection. The synchronization is done in a client–server scheme, where SUMO acts as a server waiting for a command from Veins, acting as a client, in order to proceed the simulation. In other words, at each time step, Veins trigger SUMO to execute one time step with a TraCI message. SUMO then returns vehicles' information that Veins subscribed

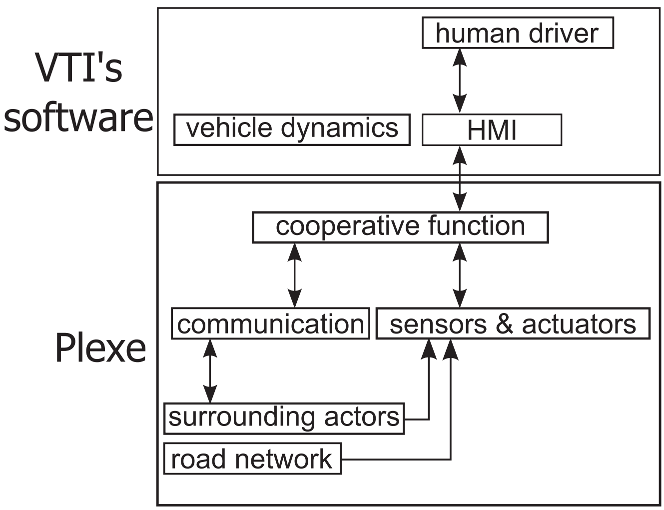


Fig. 2. Current set up of the simulation framework. The boxes indicate which simulator is responsible for which models.

ARTICLE IN PRESS

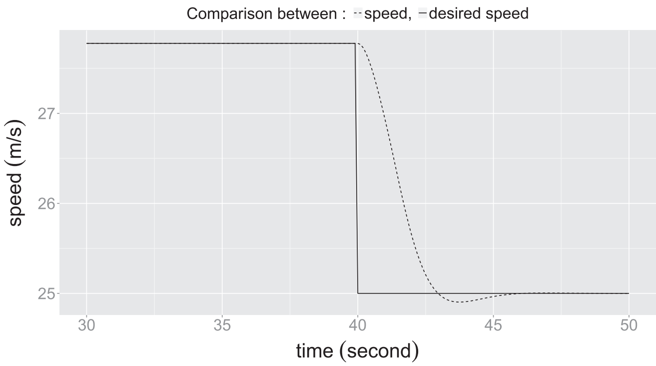


Fig. 3. Effect of the low-pass filter in *plexe-sumo* to the speed of a vehicles in simulation.

to, e.g., speed, position, etc. The same procedure is used for the Plexe version of both software. Furthermore, each vehicle in *plexe-veins* is equipped with a network interface according to IEEE 802.11p standard. The default communication parameters, listed in Table 1, defined by the author of Plexe are used. For simplicity, *plexe-veins* and *plexe-sumo* will be referred to as Veins and SUMO, respectively, for the rest of the paper.

2.3. Platooning

The simulation framework is intended to be used for testing and evaluation of many C-ITS applications. As a starting point, platooning have been selected for this study. Platooning is an application of CACC function, which is expected to be deployed soon in real traffic, in the authors' opinion. Thus, all use cases presented in Section 4 are related to CACC and its applications such as platooning. Platooning involves two or more vehicles driving in a platoon as shown in Fig. 4. A platoon usually consists of one lead vehicle (leader) in front, and one or more other vehicles following the leader, while keeping a desired safe distance (gap_{des}) between vehicles. The platoon leader is normally represented by index 0. Vehicle i normally refers to the ego vehicle and vehicle $i - 1$ is its preceding vehicle. x_i represents the position of the vehicle i. Vehicles driving close to each other in a platoon are expected to reduce their fuel consumption and provide better utilization of road spaces, as well as enhance driver's comfort. Definitions and operating concepts of CACC are summarized in Shladover, Nowakowski, Lu, and Ferlis (2015).

The two existing CACC controllers in Plexe, as mentioned above, are used in this paper. They have different definitions for the inter-vehicular gap. First, the *Rajamani* controller implements the "constant distance gap" concept. The *Rajamani* controller utilizes information from both its preceding vehicle and platoon's leader. All information regarding speed and acceleration are received via V2V communication, except the distance to the preceding vehicle, which is assumed to be measured by radar.

On the other hand, the *Ploeg* controller uses the "constant time gap" concept, which means that the gap is defined by the time gap (in seconds) between its own front bumper and the preceding vehicle's rear bumper. In other words, it is the amount of time between the ego vehicle's front bumper and preceding vehicle's rear bumper, when they pass a fixed point on the road, given the current speed. Therefore, the inter-vehicular gap depends on the vehicle's speed. Furthermore, only information from the preceding vehicle, i.e. speed, acceleration, and inter-vehicular distance, are used as control parameters. Among all the information, only the acceleration of preceding vehicle is obtained via V2V communication.

Table 1
Network parameters in *plexe-veins*.

Parameter	Value
Path loss model	Free space ($\alpha = 2.0$)
PHY model	IEEE 802.11p
MAC model	1609.4 single channel (CCH)
Frequency	5.89 GHz
Bitrate	6 Mbit/s (QPSK R = $\frac{1}{2}$)
Access category	AC_VI
MSDU size	200B
Transmit power	20 dBm

Please cite this article in press as: Aramrattana, M., et al. A simulation framework for cooperative intelligent transport systems testing and evaluation. *Transportation Research Part F* (2017), http://dx.doi.org/10.1016/j.trf.2017.08.004

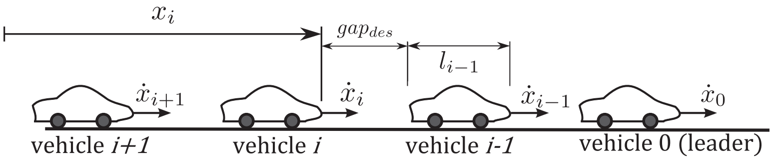


Fig. 4. A platoon of vehicles with parameters.

3. C-ITS simulation framework

The connections between VTI's driving simulation software and Plexe is shown in Fig. 5. The connection denoted "TraCI" is the existing connection between Veins and SUMO, used for synchronization and data exchange as described in Section 2. Two TCP connections are established between the driving simulation software and Plexe, namely TCP_{sync} and TCP_{app}. Both connections use the TraCI protocol defined in Wegener et al. (2008), only a few TraCI commands are added in this work, while message formats and data types remain as defined. The TCP_{sync}, is added for synchronization between the VTI's driving simulation software and Plexe. Through this connection, the driving simulation trigger the process between Veins and SUMO. Afterwards, the received vehicles' information in Veins are forwarded to the driving simulator. All simulators are running in real-time with 0.01 s time step.

Another TCP connection, TCP_{app}, is created for exchanging information from the application layer of Veins to the VTI's software. For example, to exchange information between Veins and a control logic implemented in the driving simulator to control a vehicle in the simulation, as presented in the second use case of Section 4.1.

In the VTI's driving simulation software, a plug-in was developed. It receives information such as vehicle name, speed, and position (Cartesian coordinate system). One of the vehicles in the simulation is chosen as ego vehicle. The driving simulation software then visualizes perspectives from the ego vehicle. Moreover, it implements realistic visualization of lane change manoeuvres. Since the car-following models in SUMO does not consider lateral acceleration, lane changes in SUMO occur instantaneously. Vehicles switch from one lane to another in one time step. When a lane change occurs in SUMO, the plug-in in VTI's driving simulator will display a smooth lane changing manoeuvre. The manoeuvre is executed using a proportional-integral-derivative (PID) controller that output yaw velocities based on the difference between the desired and current lateral positions.

For the platoon merging scenario in Section 4.2, the application and protocol layer of the example are extended to support the scenario. Apart from cooperative awareness messages (CAM) (ETSI TS 102 637-2) and decentralized environmental notification messages (DENM) (ETSI TS 102 637-3), the i-GAME project, organizer of the Grand Cooperative Driving Challenge (GCDC) 2016, defined an i-GAME cooperative lane change message (iCLCM). Thus, it is added to Plexe. Among 29 data fields in iCLCM, the data fields that will be discussed in this paper are listed in Table 2. Furthermore, to change lane in SUMO, a lane changing model is implemented in each vehicle. When the vehicle is told to change lane, the model makes a decision whether to change lane or not. It considers the space between vehicles in the other lane. If the space is not large enough, the vehicle will not change lane and instead try to speed up and overtake. Since vehicles are driving fairly close to each other in platooning applications, the lane changing model in SUMO often results in vehicles trying to speed up and overtake the platoon leader. Therefore, the lane changing model in SUMO is modified, to make decisions without considering the space in another lane. Hence, vehicles change lane immediately, when they are told to do so.

The most recent addition to the simulation framework is the support to involve a human driver. Previously in Aramrattana, Larsson, Jansson, and Nåbo (2016), Aramrattana, Larsson, Jansson, and Nåbo (2016), all vehicles in the scenarios are fully-automated, and the human driver is only acting as a spectator. This recent support allows a human driver to participate and control a vehicle in the same scenarios with connected and automated vehicles. Moreover, the human driver can also interact with surrounding vehicles through HMI, e.g. pushing a button on the steering wheel to send V2V message. To enable this, the longitudinal speed, and current lane information is sent from the driving simulation software to SUMO via the TCP_{app} connection. Although, the human driver can drive freely in the driving simulation software, the lane changes in

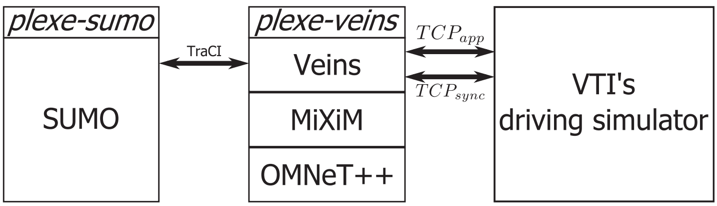


Fig. 5. Overview of the simulation framework.

Table 2
Part of i-GAME Cooperative Lane Changes Message (iCLCM).

Field	Description
stationID	ID of the vehicle
MIO_ID	ID of the most important object in front (MIO)
mergeRequestFlag	Is merge requested? (boolean)
STOMFlag	Is it safe to merge in front? (boolean)
mergingFlag	Is it started merging? (boolean)
FWDPairID	ID of the forward pair
BWDPairID	ID of the backward pair
firstVehicleFlag	Is this platoon leader? (boolean)
PlatoonID	PlatoonA = 1, PlatoonB = 2

SUMO is still instantaneous as mentioned above. Decisions to change lane are determined by the position of the front axel of the ego vehicle in the driving simulator, i.e. as soon as the front axel cross the lane marking in the driving simulator, that vehicle switch lane in SUMO instantaneously. Hence, this recent addition enables human factors studies in the C-ITS context using the simulation framework. Future research and remaining challenges will be further discussed in Section 5.

4. Simulation results

Two platooning scenarios are simulated to display potential use cases of the simulation framework. At the same time, they serve as test cases to evaluate the simulation framework. The results are presented in this section, and will be discussed further in the Section 5.

4.1. Scenario 1 – Simple platooning scenario

This scenario simulates a platoon of five vehicles running on a straight road as depicted in Fig. 6. The platoon leader starts with the speed of 100 km/h (27.78 m/s). At the 40 and 100 s simulation time, the speed of the platoon's leader is changed to 25 and 30 m/s respectively. And, after 60 s, a command to increase inter-vehicular gap is sent to all vehicles in the platoon. Default communication network parameters in Veins were used. This scenario is similar with the one presented in Aramrattana et al. (2016).

4.1.1. Use Case 1

The first use case compares two existing CACC controllers in Plexe. As aforementioned, the two controllers use different definitions for inter-vehicular gap. Therefore, the default time head way is set to 0.5 s. After receiving the "increase gap" command, the controllers then modify the parameter to 20 m and 1 s respectively. Fig. 7b and a illustrates speed and distance between vehicles plotted from data collected in Veins. This use case illustrates the capability of testing different control strategies with human drivers in the loop. For example, from the plot, it can be observed that the two controllers have different characteristics. With the driving simulator, one can observe behaviours of the controllers from the driver's seat. Thus, user's acceptance can be studied.

4.1.2. Use Case 2

The second use case perform a similar scenario, with all vehicles controlled by the *Rajamani* controller, except for the third vehicle in the platoon (vehicle number 2 in Fig. 6), that was controlled via the driving simulator executing the simple step function control logic shown in (1).

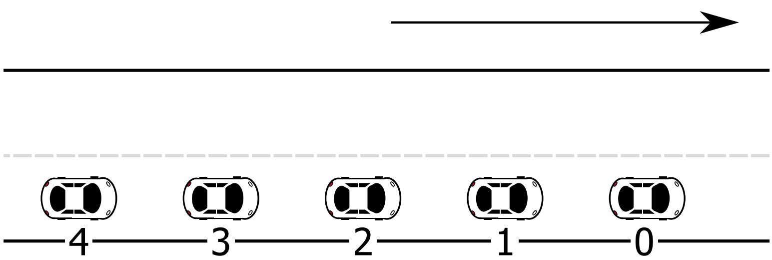
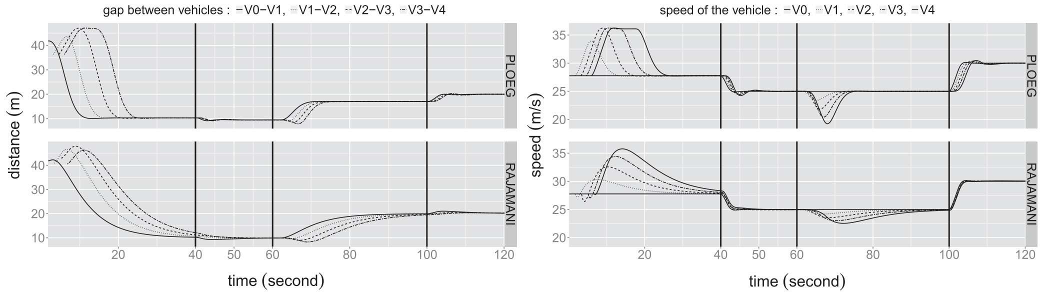


Fig. 6. Vehicles' identification in the simple platooning scenario.



(a) Distances between vehicles in the use case 1. (b) Speed profiles of the vehicles in the use case 1.

Fig. 7. Plots of vehicles' speed and distances between vehicles from simulation of the use case 1. Vehicles' maximum acceleration and deceleration are 3.0 and 5.0 m/s^2 respectively.

$$\dot{x}_i = \begin{cases} \dot{x}_{i-1} - 5\ \text{km/h}, & \text{if } gap_{des} < 12\ \text{m} \\ 120\ \text{km/h}, & \text{otherwise} \end{cases} \tag{1}$$

$\dot{x}_i$ represents desired speed of the ego vehicle and $\dot{x}_{i-1}$ is speed of its preceding vehicle. The desired speed was sent from the driving simulator to SUMO via Veins. The speed then went through the CC model in SUMO, which also implemented the first order low-pass filter for power train modelling as mentioned in Section 2. The result is compared with executing the same control logic in Veins with the distance in driving simulator used as a reference parameter in both cases. The result is presented in Fig. 8, which shows that the simulation environment is synchronized and in phase. Moreover, it elaborate on flexibility of the simulation framework, by showing that the same result can be obtained, regardless of whether the control logic is executed by the driving simulator or Veins. Therefore, in this simulation framework, different types of controllers are able to perform together in the same environment, not only limited to car-following models in SUMO. Moreover, the vehicle controlled by Eq. (2) can be seen as disturbance to the vehicles behind, as reflected in the distances plot between vehicle 2–3 and 3–4 in Fig. 8.

4.2. Scenario 2 – Platoon merging scenario

In this scenario, a simplified version of platoon merging scenario from the GCDC 2016 competition is implemented. The simulation run for 200 s, starting with two platoons: five vehicles on the right lane; four vehicles on the left lane. Fig. 9 illustrates the two platoons with vehicles' identification. The two platoons are driving at 60 km/h for about 3300 m. At the simulation time 50 s, the merge request message is sent out and the platoons start the pair-up process. The first vehicle enter the merging zone at 2000 m (about 150 s in simulation time). The platoons are merged and reformed to one platoon, and drive until the end of the simulation. Overview of the simulation is illustrated in Fig. 10.

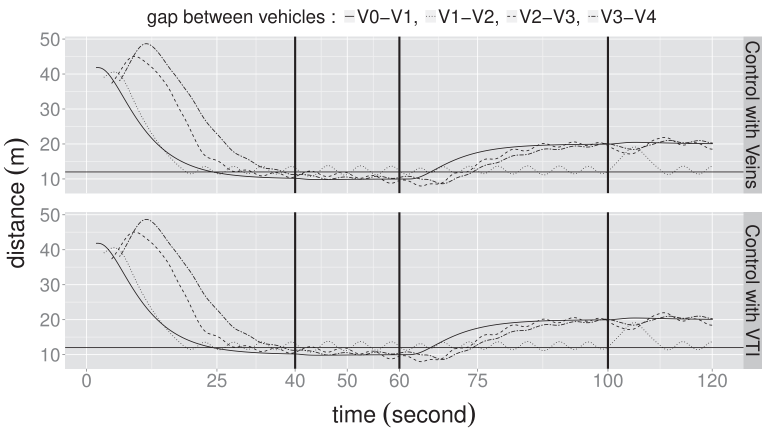


Fig. 8. Plots of distances between vehicles with the third vehicle of the platoon (vehicle 2) using the simple step function control logic described in (1) (Outputs of the control logic is also smoothened by the low-pass filter modelling drive chain dynamics). The horizontal line is drawn at 12 meters.

Please cite this article in press as: Aramrattana, M., et al. A simulation framework for cooperative intelligent transport systems testing and evaluation. *Transportation Research Part F* (2017), http://dx.doi.org/10.1016/j.trf.2017.08.004

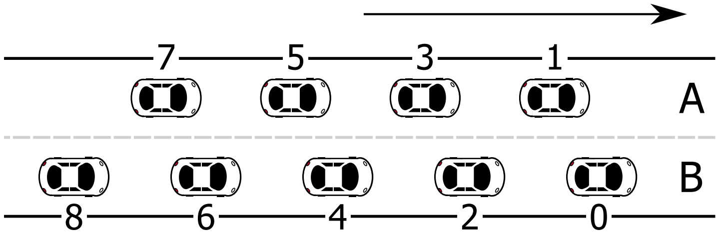


Fig. 9. Arrangement of vehicles with their identification in the platooning phase.

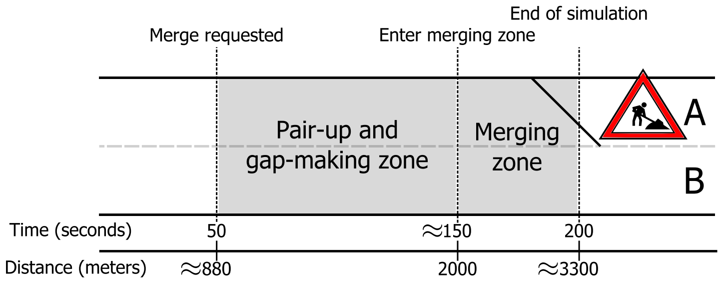


Fig. 10. Overview of the simulation.

The implementation is described as a state machine diagram in Fig. 11a for *platoon A* and Fig. 11b for *platoon B*. The scenario will be briefly described below, more detailed explanation of each state can be found in Aramrattana et al. (2016).

4.2.1. Platooning phase

In this phase, two platoons are driving at 60 km/h on a straight two-lane highway, "*platoon A*" driving on the left lane, "*platoon B*" driving on the right lane, as depicted in Fig. 9. Each platoon is led by the organizer's pace car (OPC)—vehicles number 0 and 1 in Fig. 9—which are not considered as a part of the respective platoons, and not participating in the merging. Thus, the platoon's leader, which holds the first vehicle (FV) flag for each platoon, is the vehicle immediately behind the OPC (vehicle number 2 and 3 in Fig. 9). While platooning, the desired distance between vehicles is defined in Eq. (2).

$$CACCgap = r + hv \tag{2}$$

CACCgap is a desired inter-vehicle distance in meter, r is a standstill distance (6 m is used as suggested in the GCDC documentation (Salunkhe, Nijssen, & Terken, 2016)), h is time headway, and v is velocity of the vehicle in m/s. In this scenario, *CACCgap* equals to 11 m is simulated.

4.2.2. Pair-up phases

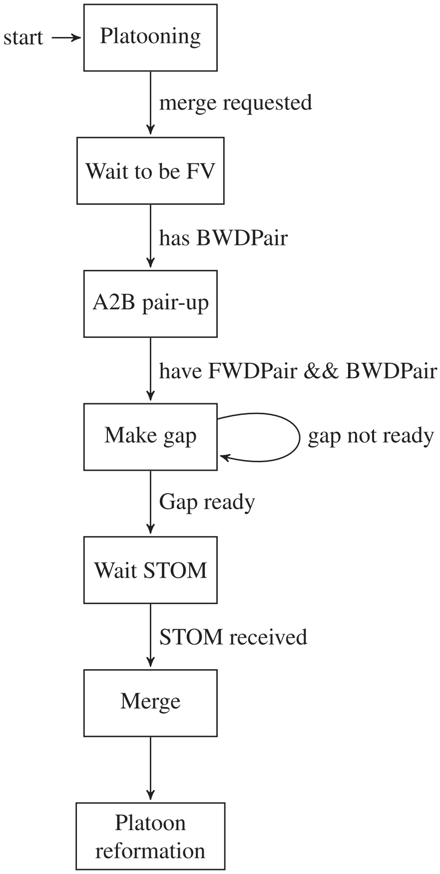
After that, the platoons received information about a road-work ahead, the vehicles in both platoons pair-up with their immediate neighbours and make the required gap with respect to each other, Gap_{FW} as illustrated in Fig. 12. For instance, vehicle number 3 will pair-up with vehicle number 2 and 4, vehicle number 4 pair with vehicle number 3 and 5. This is done based on information received via the wireless communication. A vehicle may have forward and backward pairs. Forward pair is defined as a vehicle in the other platoon that is, longitudinally, between the ego vehicle and its preceding vehicle, e.g. vehicle number 3 is the forward pair of vehicle number 4. Vice versa, backward pair is a vehicle in the other platoon that is, longitudinally, between the ego vehicle and its immediate follower (if any). Vehicle number 5 is the backward pair of vehicle number 4, for instance. If a vehicle has no follower such as vehicle number 8, it does not have the backward pair.

4.2.3. Gap making and merging phase

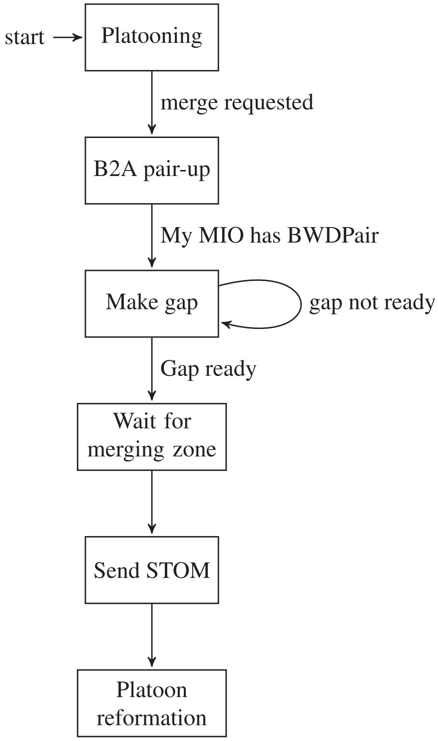
When a vehicle has identified its pairs, it will start to make gap in relation to its forward pair by adjusting the gap to its preceding vehicle. Eq. (3) is used for making the gap.

$$gap_{desired} = gap_{current} + K_p * (gap_{safe} - gap_{FW}) \tag{3}$$

$gap_{desired}$ is the desired *CACCgap*, $gap_{current}$ is the current distance to the vehicle in front, K_p is the controller gain, gap_{safe} is the desired distance to the forward pair, and gap_{FW} is the current gap between ego vehicle and its forward pair. Variables are illustrated in Fig. 12. In the simulation, the goal is to make safety distance (gap_{FW} in Fig. 12) of 20 m. Moreover, in the sim-



(a) State diagram for vehicles in the *platoon A*.



(b) State diagram for vehicles in the *platoon B*.

Fig. 11. State diagram for vehicles in the platoon merging scenario.

ulation, each vehicle in *platoon A* has 25 s to make the gap. After 25 s, the gap is assumed to be done. Then, after 2000 m in the simulation, the vehicles enter the merging zone, where the two platoons merge into one platoon on the right lane. After the platoons have merged, the inter-vehicle distance is reduced to the starting value, which is defined as 11 m.

4.3. Scenario 3 – Human driver

In this scenario, a human driver is driving the ego vehicle in the driving simulation software running on a desktop computer. The driver controls the ego vehicle with a steering wheel and pedals, the ego vehicle is assumed to have an automatic gearbox. Data is collected from both the driving simulation software and Plexe. Lateral positions and speed of the ego vehicle collected from both software are presented in Fig. 13a and b, respectively. The results show that simulators in the simulation framework are synchronized.

5. Discussion

Each individual simulator in this framework has not been fully utilized in this work due to limitations and remaining challenges listed below. Moreover, in this section, alternatives to overcome such limitations and challenges will be proposed along with future research directions.

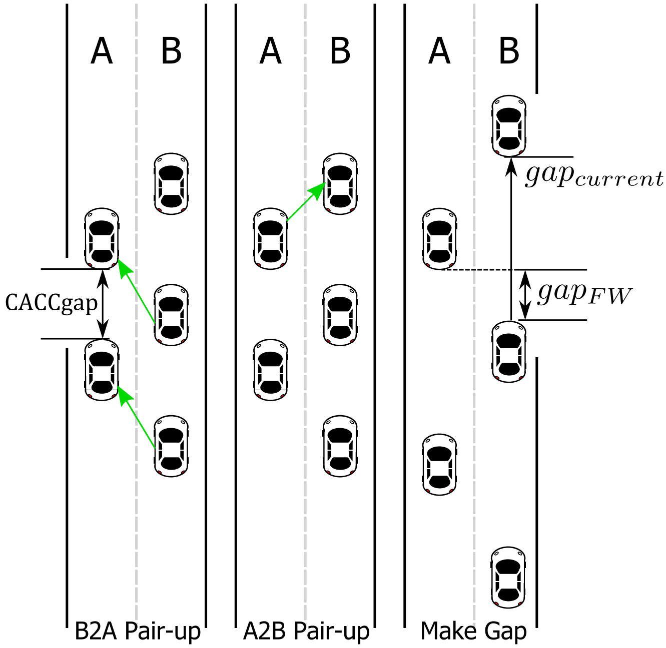
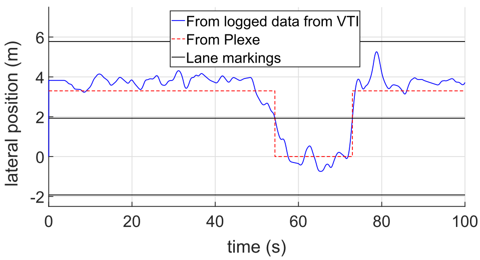
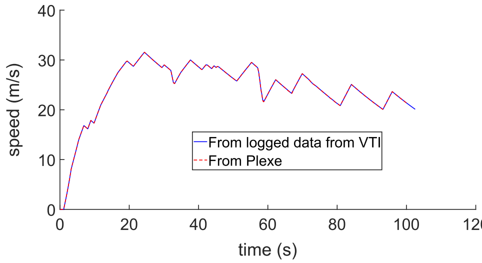


Fig. 12. Pair-up and gap-making phase.



(a) Lateral position of the ego vehicle in the scenario 3.



(b) Speed of the ego vehicle in the scenario 3.

Fig. 13. Ego vehicle's data collected from the driving simulation software and Plexe during the scenario 3.

5.1. Involving human drivers

It is important to have human drivers included in the loop while testing and evaluating C-ITS, because they are expected to be involved either as actual drivers, or drivers in stand-by mode. Therefore, in this work, the usages of the VTI's driving simulation software are twofold; (a) allow the human driver to visualize and experience platooning from the driver's perspective; and (b) allow the human driver to drive manually in the platooning scenarios. Although more scenarios have been enabled by the driving simulator, the scenarios which the driver switch from automated to manual driving during the simulation, and vice versa, are not yet considered. Thus, studies related to mode transition between automated and manual driving are not available in this simulation framework yet.

5.2. Lateral positions in traffic simulation

Since traffic simulator such as SUMO does not consider lateral acceleration of vehicles, lane-changing occurs instantaneously in one time step. In other words, vehicles "jump" from one lane to another in one time step. Moreover, all vehicles in SUMO are assumed to be driving ideally in the middle of a lane, which is not the case in the driving simulator, where the vehicles can be anywhere as decided by human drivers or automated driving control systems. This can be considered unrealistic from the testing and evaluation points of view, because C-ITS functions, CACC in this case, are implemented in SUMO. However, the applications under test hitherto are rather simple, the vehicles are assumed to be equipped with only a forward-looking radar, that detects an object as long as the object is in the same lane. With such limited field-of-view, having more details information regarding lateral position of the other vehicles does not significantly improve the object detection by radar. From the human driver's perspective, a realistic lane-changing manouevre is displayed in the driving simulator when a lane change occurs in SUMO.

To have more realistic lateral positions in a lane, SUMO recently released a "sub-lane" option, which divides each lane into several smaller sub-lanes. This can be at least a partial solution to this problem. However, this option has been added recently to the official *plexe-sumo*, while the simulation framework still uses the older version.

5.3. Validation of the simulation framework

Furthermore, the simulation framework needs to be validated more thoroughly to prove its correctness and usefulness. To begin with, the first-order low-pass filter used in *plexe-sumo* to imitate vehicles' power-train behaviour is compared with the logged data from a Volvo S60 car participated in the GCDC 2016. Fig. 14 illustrates the plot between speed profile of a simulated vehicle and logged real vehicle data. Both vehicles start from standstill and trying to accelerate to the desired speed of 30 km/h (about 8.33 m/s). Also, both vehicles have 2 m/s^2 as their maximum acceleration capability.

Nonetheless, the example above only validates one aspect of the simulation framework, i.e. the power-train model. Further validations on other models are needed as future work.

5.4. Future work

The work may be extended to consider also heterogeneous traffic scenarios, where combinations of connected, non-connected, automated, and non-automated vehicles share the road space. These scenarios can be expected in the early deployment of C-ITS. Studying how human drivers will perceive and adapt to such situations is an important issue to consider. Many research questions can be studied using the simulation framework, such as *How human drivers would interact with other autonomous vehicles? How to present information gathered via V2V to the driver in a good way? What is an acceptable inter-vehicular distance for platooning?*

Human behavior is often unpredictable and the actions a driver can perform may vary a lot. This can be seen as an uncertainty in the C-ITS system in a heterogenous traffic scenario, and must be considered in testing and evaluation, as well as the uncertainties in wireless communication and sensors. The former is rarely studied in the context of C-ITS compared to the latter issues. With a driving simulator included in the simulation framework, such disturbances can be studied, and their effects can be analysed.

Lastly, before the deployment of C-ITS applications, testing and evaluation is needed from many perspectives; for instance, user's acceptance and safety. A methodology on how to use simulation, especially this simulation framework, to

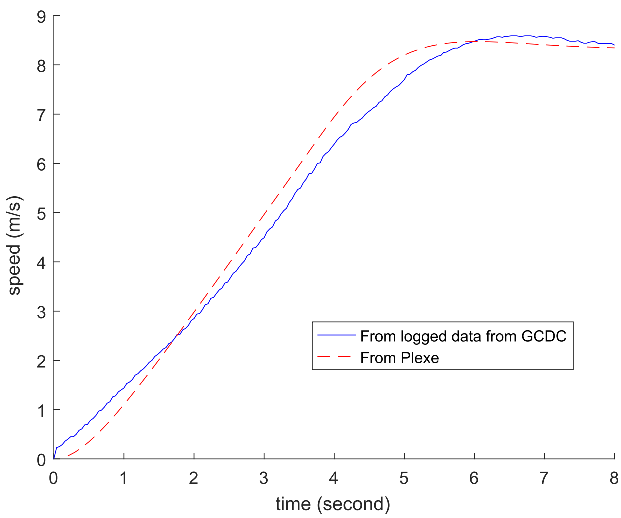


Fig. 14. Comparison between speed profile.

support testing and evaluation of C-ITS application is desired. Moreover, one of the final goals is to use the presented simulation framework for all C-ITS applications. However, it has so far not been used for simulating other C-ITS applications apart from the platooning scenarios presented in this paper. Thus, as a future work, the simulation framework may need further development to cover other C-ITS scenarios.

6. Conclusion

During transition towards cooperative intelligent transport systems (C-ITS), where vehicles in the systems are expected to be automated and connected, human drivers will still be involved in the systems in many ways. For instance, by driving a vehicle, interacting with autonomous vehicles, or monitoring the automated function in the vehicle. Therefore, involving human drivers in the loop is essential for testing and evaluation of C-ITS. This paper presents a simulation framework, which combines driving-, traffic-, and network-simulators. In particular, the simulation framework integrated the driving simulation software from VTI with an integrated traffic-network simulator, Plexe. The simulation results from the following three scenarios are presented; (a) the simple platooning scenario; (b) the simplified version of platoon merging scenario from GCDC 2016; and (c) the scenario where a human driver drives the ego vehicle. Although Plexe designs specifically for platooning application, it has potential to develop further to include other C-ITS applications. By combining three types of simulators, the simulation framework provides great opportunities to study several aspects of C-ITS. For example, studies from human driver's perspectives are enabled by the driving simulator; communication-related issues can be addressed by the network simulator; impacts of C-ITS on the traffic systems can be analysed using the traffic simulator.

The development of the simulation framework is still ongoing, and many challenges have to be addressed in the future work as discussed in Section 5. Apart from that, there are still several open questions concerning the usages of the simulation framework, and how to utilize the simulation results with respect to testing and evaluation of C-ITS.

Acknowledgement

The authors would like to thank Michele Segata from the University of Trento (Italy) for his valuable advices regarding modification and details of Plexe. Also, Jonas Andersson Hultgren and Bruno Augusto from VTI, for their guidances on the VTI's driving simulation software. This work is supported by SAFER – Vehicle and Traffic Safety Centre at Chalmers, as a part of Vehicle ICT Innovation Methodology (VICTIg) project.

References

Aramrattana, M., Larsson, T., Jansson, J., & Nåbo, A. (2016). Cooperative driving simulation. In *The driving simulation conference 2016 VR*.

Aramrattana, M., Larsson, T., Jansson, J., & Nåbo, A. (2016). Extended driving simulator for evaluation of cooperative intelligent transport systems. In *Proceedings of the 2016 annual ACM conference on SIGSIM principles of advanced discrete simulation, SIGSIM-PADS '16* (pp. 255–258). New York, NY, USA: ACM. http://dx.doi.org/10.1145/2901378.2901397.

ETSI. ETSI TS 102 637-2, Intelligent Transport Systems (ITS); vehicular communications; basic set of applications; Part 2: Specification of co-operative awareness basic service.

ETSI. ETSI TS 102 637-3, Intelligent Transport Systems (ITS); vehicular communications; basic set of applications; Part 3: Specifications of decentralized environmental notification basic service.

Gajananan, K., Sontisirikit, S., Zhang, J., Miska, M., Chung, E., Guha, S., & Prendinger, H. (2013). A cooperative its study on green light optimization using an integrated traffic, driving, and communication simulator. In *Proceedings 36th-ATRF*.

Hikita, T., Kasai, T., & Yoshioka, A. (2008). Integrated simulator platform for evaluation of vehicular communication applications. In *IEEE international conference on vehicular electronics and safety, 2008. ICVES 2008* (pp. 323–327). doi:http://dx.doi.org/10.1109/ICVES.2008.4640899.

Jansson, J., Sandin, J., Augusto, B., Fischer, M., Blissing, B., & Källgren, L., (2014). Design and performance of the VTI Sim IV. In *Driving simulation conference*.

Krajzewicz, D., Erdmann, J., Behrisch, M., & Bieker, L. (2012). Recent development and applications of SUMO - Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements, 5*(3&4), 128–138.

Piórkowski, M., Raya, M., Lugo, A. L., Papadimitratos, P., Grossglauser, M., & Hubaux, J.-P. (2008). TraNS: Realistic joint traffic and network simulator for VANETs. *SIGMOBILE Mobile Computing and Communications Review, 12*(1), 31–33. http://dx.doi.org/10.1145/1374512.1374522.

Ploeg, J., Scheepers, B. T. M., van Nunen, E., van de Wouw, N., & Nijmeijer, H. (2011). Design and experimental evaluation of cooperative adaptive cruise control. In *2011 14th international IEEE conference on intelligent transportation systems (ITSC)* (pp. 260–265). doi:http://dx.doi.org/10.1109/ITSC.2011.6082981.

Prendinger, H., Miska, M., Gajananan, K., & Nantes, A. (2014). A cyber-physical system simulator for risk-free transport studies. *Computer-Aided Civil and Infrastructure Engineering, 29*(7), 480–495. http://dx.doi.org/10.1111/mice.12068.

Rajamani, R. (2012). *Vehicle dynamics and control* (2nd ed.). US: Springer.

Rondinone, M., Maneros, J., Krajzewicz, D., Bauza, R., Cataldi, P., Hrizi, F., ... Cartolano, F. (2013). iTETRIS: A modular simulation platform for the large scale evaluation of cooperative {ITS} applications. *Simulation Modelling Practice and Theory, 34*, 99–125. http://dx.doi.org/10.1016/j.simpat.2013.01.007 <http://www.sciencedirect.com/science/article/pii/S1569190X1300018X>.

SAE International (2014). Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems (SAE J 3016).

Salunkhe, H., Nijssen, B., & Terken, J. (2016). DEL_i-GAME_D7.1 Proposal for GCDC judging criteria and their evaluation (Draft). <http://gcdc.net/images/doc/D7.1.Judging.criteria.draft-2016-03-17.pdf>.

Segata, M., Joerer, S., Bloessl, B., Sommer, C., Dressler, F., & Cigno, R. L. (2014). Plexe: A platooning extension for veins. In *2014 IEEE vehicular networking conference (VNC)* (pp. 53–60). doi:http://dx.doi.org/10.1109/VNC.2014.7013309.

Shladover, S. E., Nowakowski, C., Lu, X.-Y., & Ferlis, R. (2015). Cooperative adaptive cruise control: Definitions and operating concepts. *Transportation Research Record: Journal of the Transportation Research Board* (2489), 145–152.

Sommer, C., German, R., & Dressler, F. (2011). Bidirectionally coupled network and road traffic simulation for improved IVC analysis. *IEEE Transactions on Mobile Computing, 10*(1), 3–15. http://dx.doi.org/10.1109/TMC.2010.133.

US Department of Transportation (2016). Federal automated vehicles policy. <https://www.transportation.gov/AV>. Accessed 22 December 2016.

Varga, A., & Hornig, R., (2008). An overview of the OMNeT++ simulation environment. In *Proceedings of the 1st international conference on simulation tools and techniques for communications, networks and systems & workshops, Simutools '08, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium* (pp. 60:1–60:10). <http://dl.acm.org/citation.cfm?id=1416222.1416290>.

Wang, Y., Hu, J., & Wang, Q. (2010). A simulation study of dynamic route guidance based on inter-vehicle communication. In *2010 international conference on communications and mobile computing (CMC)* (Vol. 1, pp. 397–403). doi:http://dx.doi.org/10.1109/CMC.2010.61.

Wegener, A., Piórkowski, M., Raya, M., Hellbrück, H., Fischer, S., & Hubaux, J.-P. (2008). Traci: An interface for coupling road traffic and network simulators. In *Proceedings of the 11th communications and networking simulation symposium, CNS '08* (pp. 155–163). New York, NY, USA: ACM. http://dx.doi.org/10.1145/1400713.1400740.

Zhao, Y., Wagh, A., Hou, Y., Hulme, K., Qiao, C., & Sadek, A. W. (2014). Integrated traffic-driving-networking simulator for the design of connected vehicle applications: Eco-signal case study. *Journal of Intelligent Transportation Systems*, 1–13.

Appendix C

Paper III
Simulation of Cut-In by Manually Driven Vehicles in Platooning Scenarios

Maytheewat Aramrattana, Tony Larsson, Cristofer Englund, Jonas Jansson, and Arne Nåbo

2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)
Yokohama, Japan, October 16-19, 2017

(Link to the publication)

Simulation of Cut-In by Manually Driven Vehicles in Platooning Scenarios

Maytheewat Aramrattana*,†, Tony Larsson*, Cristofer Englund*,‡, Jonas Jansson† and Arne Nåbo†
*School of Information Technology, Halmstad University, Halmstad, Sweden
Email: {maytheewat.aramrattana, tony.larsson}@hh.se
†The Swedish National Road and Transport Research Institute (VTI), Linköping, Sweden
Email: {jonas.jansson, arne.nabo}@vti.se
‡RISE Viktoria, Gothenburg, Sweden
Email: cristofer.englund@ri.se

***Abstract*—In the near future, Cooperative Intelligent Transport System (C-ITS) applications are expected to be deployed. To support this, simulation is often used to design and evaluate the applications during the early development phases. Simulations of C-ITS scenarios often assume a fleet of homogeneous vehicles within the transportation system. In contrast, once C-ITS is deployed, the traffic scenarios will consist of a mixture of connected and non-connected vehicles, which, in addition, can be driven manually or automatically. Such mixed cases are rarely analysed, especially those where manually driven vehicles are involved. Therefore, this paper presents a C-ITS simulation framework, which incorporates a manually driven car through a driving simulator interacting with a traffic simulator, and a communication simulator, which together enable modelling and analysis of C-ITS applications and scenarios. Furthermore, example usages in the scenarios, where a manually driven vehicle cut-in to a platoon of Cooperative Adaptive Cruise Control (CACC) equipped vehicles are presented.**

I. Introduction

Cooperative Intelligent Transport Systems (C-ITS) is a strong trend in the development of the future transport systems, where the actors are equipped with wireless communication modules that enable them to communicate, interact, and co-operate. The overall goal with C-ITS is to improve safety, comfort, and efficiency [1]. The actors in C-ITS are vehicles and road infrastructure such as traffic lights, eventually it can also include pedestrians and bicyclists in the future[1]. Apart from connectivity, which is needed to achieve maximum benefits from the systems, different levels of automation and intelligent adaptivity of vehicles and infrastructure may also be part of C-ITS.

Modelling and simulation are often used to support design and evaluation of C-ITS applications [2]–[5] including CACC/Platooning applications [6]–[10]. While combinations of traffic and network simulators are commonly used to study such applications [6], [8], [10], driving simulators are not common. Major reasons for this could be fewer available open source driving simulation software, compared to traffic and network simulators. And, most of the driving simulators are coupled with hardware which requires extra space and cost. Moreover, the main purpose of driving simulator studies is focused on the driver, efforts are put on details of the ego vehicle and its driving behaviour, rather than studying a system of vehicles. However, driving simulators can offer realistic human driver behaviour and interaction, which is required in order to study the traffic scenarios with human drivers involved.

The main study cases of this paper are applications in the context of Cooperative Adaptive Cruise Control (CACC) and especially platooning. CACC and automated platooning has slightly different operational concepts depending on the inter-vehicle gap regulation strategy, as stated in [11]. However, in this paper the terms will be used interchangeably, referring to an application, that utilize vehicle-to-vehicle (V2V) communication to maintain a stable platoon[2] of vehicles. The goals of the application can include, e.g., to improve safety, maintain string stability[3], reduce fuel consumption, and improve driver comfort. Most of the studies assume homogeneous traffic, where all vehicles are identical, connected, and automated. Even so, considering real traffic scenarios on public roads is not yet the case. In early deployment of C-ITS applications, heterogeneous traffic—where combinations of vehicles with different capabilities are involved in the system—are to be expected [12]. Such heterogeneous traffic scenarios are not often studied, especially the ones involving human driven vehicles.

In particular, this study focuses on scenarios when a manually driven vehicle intervene with a platoon while doing a cut-in manoeuvre, which frequently occurs in today's traffic. In relation to ACC[4]/CACC operations, a few studies have considered this type of scenario. *V. Milanés & S. E. Shladover* [13] have presented effects of the cut-in and cut-out in long strings of CACC vehicles, with results from simulations and on-road experiments. Moreover, *Annika F.L. Larsson, et al.* [14] presented a study on reaction time of drivers in an ACC-equipped vehicle, when a cut-in happens.

Therefore, the contributions of this paper are twofold:

[1]Only vehicles are studied in this paper.

[2]In this paper, the term platoon is also refers to a string of vehicles that are operating with the CACC function activated.

[3]The effects of distance error, or disturbances do not amplify as it propagates backwards to the following vehicles in the platoon.

[4]Adaptive Cruise Control.

- We present a simulation framework with possibility to manually drive a vehicle in a C-ITS scenario.
- We show simulation results of the behaviour of two different CACC controllers in a cut-in scenario using the simulation framework.

The remainder of the paper is organized as follows. Section II introduces a C-ITS simulation framework with possibility to involve a manually driven car in C-ITS scenarios. This section also presents challenges, and results from including a human driver in the simulation framework. Section III defines the cut-in scenario, and parameters related to the CACC controllers and the simulation study. Section IV presents results from the scenario, where the manually driven car cut-in between vehicles in the platoon. Future work is presented in Section V. Finally, the paper is concluded in Section VI

II. THE C-ITS SIMULATION FRAMEWORK

A. Background

The C-ITS simulation framework consists of driving-, traffic-, and network simulators. The driving simulator is executed by the driving simulation software from the Swedish National Road and Transport Research Institute (VTI). And Plexe [6]—Platooning Extension for Veins—is used for traffic and network simulation. The software structure and included simulation models are illustrated in Fig. 1.

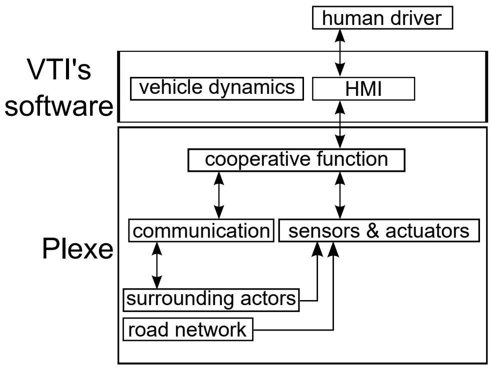


Fig. 1: Simulation models that are considered in the C-ITS simulation framework.

VTI's driving simulation software is developed in-house at VTI. It is implemented in C++, and the same software kernel can be run on a desktop computer, or the computer-controlled moving-base driving simulators at VTI [15]. Plexe is developed based on the microscopic traffic simulator, SUMO (Simulation of Urban Mobility) [16], and the network simulator, Veins [17]. Plexe extends both SUMO and Veins to support more realistic simulation of platooning scenarios, by considering vehicle dynamics in the form of actuation lag (modelled by a low pass filter), and the V2V communication protocol stack used to send messages according to the IEEE 802.11p standard.

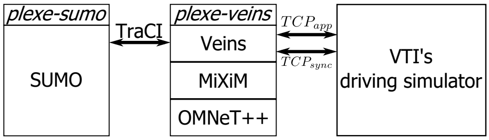


Fig. 2: The TCP connections between simulation software in the C-ITS simulation framework.

As illustrated in Fig. 2, the different simulation software used in the framework are connected via three transmission control protocol (TCP) connections: *a*) TraCI (traffic control interface), an existing connection for SUMO and Veins to interact according to [18]; *b*) TCP_{sync} facilitating synchronization between the driving simulator and Plexe; and *c*) TCP_{app} for exchanging information between the driving simulator and Plexe. Please refer to our previous work [19], [20] for more details.

The C-ITS simulation framework has been presented in [19], [20]. The framework is developed to support testing and evaluation of platooning applications, with a human driver in the loop. Previously, the human driver was included just as an operator or observer of the fully-automated platooning scenarios. By extending the existing framework, the human driver can now be more involved by driving a vehicle in the simulation framework. This development enables analysis of more complex platooning scenarios which are not commonly studied. For example, cut-in by *non-V2V-equipped vehicles* in platooning scenarios [13].

B. Involving a Human Driver

In driving simulators, it is common that the driver has freedom to drive anywhere in the simulated environment, this is one of the main features of every driving simulator. In contrast, the traffic simulation represents vehicles as always driving in the middle of their lane. The only lateral movement modelled in the traffic simulation is the lane-changing behaviour, which occurs instantaneously within one time step. Driving behaviour is usually restricted in most of the microscopic traffic simulators. For instance, behaviour of each vehicle in SUMO is mainly controlled by a car-following and a lane-changing models.

Car-following models in microscopic traffic simulations are usually defined by ordinary differential equations. In SUMO, the "active" car-following model can be changed during the simulation. However, at one simulation time point, only one model can be active for a vehicle. The car-following model regulates longitudinal velocity of vehicle(s), by taking into account parameters such as distance to the preceding vehicle, current acceleration of the ego vehicle, maximum acceleration, etc. These parameters are different for each car-following model.

By default, car-following models in SUMO are collision free. In other words, the car-following models considers the distance to the preceding vehicle, and determined a "safe speed". Even when a command is sent to control the speed of the vehicle, the car-following model has priority to override it, if that speed is higher than the safe speed. Because the main purpose of traffic simulators has been to study traffic flow behaviour rather than hazard or traffic safety issues.

There is no such restriction in car-following models proposed by the Plexe framework, thus enabling the vehicles to drive closely in a platoon. Thus, to overcome this restriction, the *Cruise Control* car following model presented in [6] is modified to be used in this paper. Alternatively, the simulation parameters in SUMO can be set to ignore this safety check. The same problem arises for changing lane in SUMO. The vehicle will refuse to change lane if the space is not perceived as large enough for the lane-changing model. This has been solved as presented in [20].

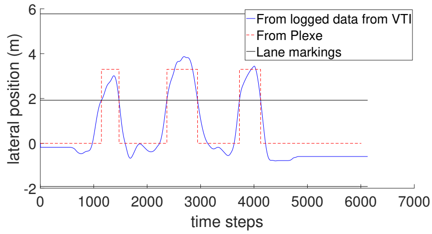


Fig. 3: Lateral positions of the manually driven vehicle within the simulation framework.

To facilitate manual driving in the simulation, information about longitudinal speed and current lane of the ego vehicle are sent from the driving simulator to Plexe via the TCP_{app} connection (as illustrated in Fig. 2). The simulation framework run at 0.01 second time step (100 Hz). Figure 3 shows plots of the logged lateral positions at each time step from Plexe and the driving simulator. The information about the ego vehicle's current lane comes from the driving simulation software. There, the lane change occurs when the front axle of the ego vehicle crosses the lane marking in the driving simulator. However, in Plexe, the lane change is instantaneous as depicted by the red dashed plot in Fig. 3.

III. The Cut-In

A cut-in scenario by a manually driven vehicle is achieved using the simulation framework presented in the previous section. The cut-in scenario is simulated on a two-lane straight highway. As illustrated in Fig. 4, four vehicles are simulated, three cars that are operated by a CACC function (the vehicle no. 1, 2, and 3 in Fig. 4) driving on the right lane, and one manually driven car (the vehicle no. 4 in Fig. 4) on the left lane. During the simulation, the driver of the manually driven car (no. 4) performs a cut-in manoeuvre, merging and placing the car in front of the second vehicle in the platoon (as shown in Fig. 4). Further, it is assumed that the manually driven car does not have automated driving capability, but may have an ability to transmit V2V communication messages such as to communicate its intention, or broadcasting *Cooperative Awareness Messages* (CAMs). Table I lists the configuration that the manually driven vehicle may have. In this paper, only one configuration marked by '⋆' in Table I is studied. That is the cut-in scenario by a manually driven vehicle without V2V communication (non-communicating vehicle).

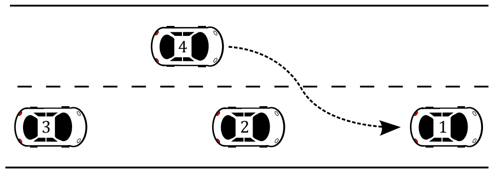


Fig. 4: The manual cut-in scenario. The vehicle no. 4 is driven by a human driver, the other cars are operated by the CACC function in the simulator.

TABLE I: Possible configurations for the human-driven car. The '⋆' indicates the one that considered in this paper.

	Sharing Intention	
Sending CAM	Yes	No
Yes		
No		⋆

The following two existing CACC controllers in Plexe are used for the platooning vehicles, namely:

- **Rajamani** is the constant-distance gap controller, which is implemented by the author of Plexe following the book by *Rajamani* [21, Chapter 7]
- **Ploeg** is the constant-time gap controller, which is implemented by the author of Plexe following the work by *Ploeg et al.* [22]

Apart from the different control strategies, there are a few more differences between these two controllers. The *Rajamani* controller uses information about the speed and acceleration of the platoon leader as control parameters, while the *Ploeg* controller only uses information about the preceding vehicle. Moreover, the *Rajamani* controller obtains the speed of the preceding vehicle from the V2V communication, while the *Ploeg* controller obtains the information from the radar. The differences are summarized in Table II. Nevertheless, during the operation of both CACC controllers in our simulation, each follower receives the information about speed, acceleration, and position of the platoon leader and its preceding vehicle via V2V communication. A desired inter-vehicle distance of 17.5 meters (0.6 second headway time) was chosen for the CACC controllers. All vehicles are equipped with radar that detects an object in front. If there is no such object, nothing is detected and the radar returns -1 as output. As mentioned in Section II-B, the lane change in Plexe happens instantaneously, when the front axle of the vehicle in the driving simulator crosses the lane marking (see Fig. 3). Therefore, in this paper the radar is assumed to have a relatively wide angle covering the driving lane, i.e. it will detect the cut-in vehicle as soon as the cut-in vehicle's front axle cross the lane marking.

TABLE II: List of the information used by the two CACC controllers and their source.

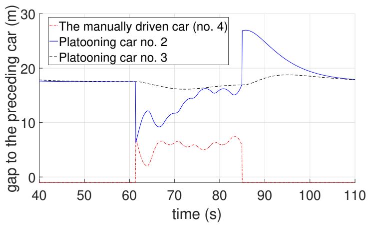
Control Parameters	Controller	
	Rajamani	*Ploeg*
Platoon leader's speed	V2V	-
Platoon leader's acceleration	V2V	-
Preceding vehicle's speed	V2V	Radar
Preceding vehicle's acceleration	V2V	V2V
Distance to preceding vehicle	Radar	Radar

IV. RESULTS

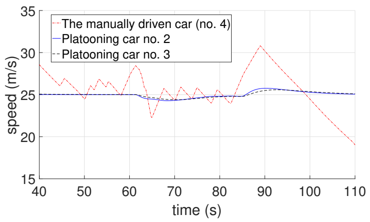
In this section, vehicle no. 4 in Fig. 4 will be referred to as the "*manually driven car*", and the platooning vehicles no. 1, 2, and 3 will be referred to as the *platoon leader*, *second vehicle*, and *last vehicle* respectively, as illustrated in Fig. 4.

A. Conventional Car Cut-In

The results of the cut-in scenario using two CACC controllers with desired inter-vehicle gap of 17.5 meters (0.6 second time headway) are presented in this subsection. At the steady-state, the platoon leader is always driving with constant speed of 90 $\mathrm{km/h}$ (25 $\mathrm{m/s}$). Since the platoon leader is not affected by the cut-in manoeuvre it is excluded from all the plots in this subsection.



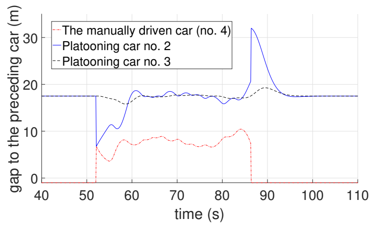
(a) Plots of the distances measured by radar to the object in front.



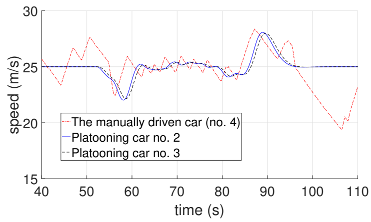
(b) Plots of each vehicle's speed.

Fig. 5: Results from the cut-in scenario using *Rajamani* controller with desired gap of 17.5 m.

Figure 5 shows the behaviour of the *Rajamani* controller, when the manually driven car cut into the platoon. Figure 5a depicts measurements from the radar in each of the vehicles. The cut-in occurs at approximately 61 second simulation time, where the measurements from radar change instantaneously for the manually driven car and the second vehicle. The inter-vehicle distance is reduced to 6.35 meters for the second vehicle. Consequently, the second vehicle adjust its speed, and is able to prevent a collision. At approximately 85 seconds, the manually driven car leave the platoon giving a big gap in front of the second vehicle. Thus, it speeds up to close the gap and maintains a stable platoon. One can observe that the controller is string stable by looking at the behaviour of the last vehicle (platooning car no. 3), which does not amplify the error caused by the second vehicle.



(a) Plots of the distances measured by radar to the object in front.



(b) Plots of each vehicle's speed.

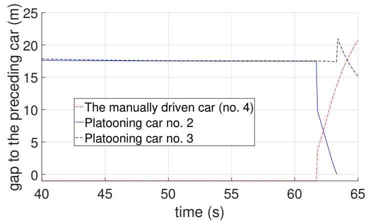
Fig. 6: Results from the cut-in scenario using *Ploeg* controller with desired gap of 0.6 second headway time.

For the *Ploeg* controller, its behaviour during the cut-in scenario is presented in Fig. 6. In this case, the cut-in occurs at approximately 52 seconds simulation time, and the inter-vehicle distance is reduced to 6.75 meters for the second vehicle. Consequently, the second vehicle adjusts its speed, and is also able to prevent a collision. The manually driven car then leave the platoon at approximately 86 seconds. Again, the string stability of the controller can be observed.

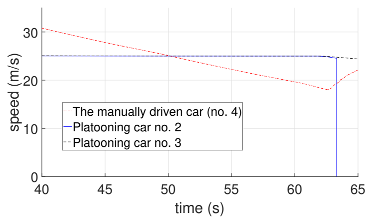
B. Collision

Plots in Fig. 7 presents a collision, when the manually driven car cut into a platoon operated by the *Rajamni* controller at 17.5 meters inter-vehicle distance. Even though the inter-vehicle gap at the cut-in point ($\approx$ 61 seconds time) is 9.76 meters, which is higher than that of the same scenario above, the large difference in speed causes the collision (see Fig 7b at 63 seconds).

On the other hand, the *Ploeg* controller was able to handle the situation, when the cut-in vehicle has a difference in speed, as shown in Fig. 8 (at 107 seconds simulation time).

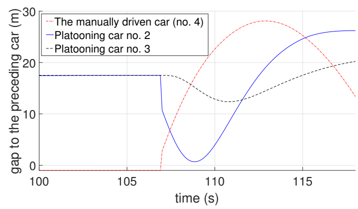


(a) Plots of the measured distance to the preceding vehicle.

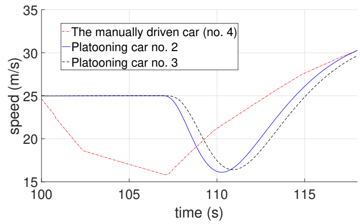


(b) Plots of each vehicle's speed.

Fig. 7: Results from the cut-in scenario, when a collision occurs after the cut-in. Platooning vehicles are using *Rajamani* controller with desired gap of 17.5 m.



(a) Plots of the measured distance to the preceding vehicle.



(b) Plots of each vehicle's speed.

Fig. 8: Results from the cut-in scenario using the *Ploeg* controller with a desired gap of 0.6 second headway time.

Nevertheless, it requires the second vehicle to apply maximum deceleration of 4.5 $\mathrm{m/s^2}$, in order to prevent a collision. The maximum deceleration is considered as an emergency braking manoeuvre, according to [23]. Furthermore, this hard-braking manoeuvre could be dangerous to the vehicles behind the platoon, and might result in a rear-end collision.

Furthermore, we simulated a similar situation with the same speed difference, but with a desired inter-vehicle gap of 30 meters. The *Rajamani* controller is then able to handle the situation and prevent the collision. However, the collision still occurs when the platoon leader speeds up, because the controller's dependency to the platoon leader's speed and acceleration. The platooning vehicles speed up to follow the platoon leader, and collide into the manually driven car. Hence, to prevent a collision, switching the active controller to ACC when a cut-in is detected by non-communicating vehicles, as suggested in [13] is required. Also, the vehicle that the manually driven car merged in front of have to be assigned as the new platoon leader for each platooning vehicle behind it.

In contrast, the *Ploeg* controller reacts differently to the platoon leader speeding up, the collision did not happen in this case. Because most of the controller's decision is based on the information from the radar, which detects the manually driven car. Although the collision did not happen in our simulation, it is still possible under certain amount of leader's acceleration, and other parameter settings than the ones used in this simulation. Therefore, even though the *Ploeg* controller behave similarly to ACC, i.e. using radar to detects inter-vehicle gap and relative speed, switching to conventional ACC when the cut-in is detected is preferable.

V. Future Work

Many aspects of cut-in scenarios are still to be explored as follows. First, having cut-in manoeuvres with such large speed difference on a highway is not common, however not impossible. The results in Section IV-B show that the different designs of CACC controller may achieve similar efficiency in terms of string stability, but it can have different effects on the safety of the driver. Therefore, in the evaluation of the CACC controller design, safety should also be considered, in addition to string stability. Further investigation on a hazard and risk analysis framework for evaluation of safety-related issues is required.

Detecting the cut-in by a non-communicating vehicle is a challenge in itself. Observing a sudden change in distance measured by the radar as shown in the scenario above is one way to detect the cut-in event. The platooning vehicles may, as a complement, be equipped with a camera system, or other sensors such as LiDAR. Then, sensor fusion algorithms can be applied to better detect, or even predict the cut-in manoeuvre. Furthermore, considering future scenarios when conventional vehicles are equipped with V2V communication modules. They can still be driven manually, but have a possibility to broadcast CAM. In addition, intentions to cut-in, or change lane, can be broadcast. Although the standard CAM does not include the *intention* of the vehicle, this has been proposed in the i-GAME Cooperative Lane Change Message (iCLCM) in the Grand Cooperative Driving Challenge 2016 [1]. To fully utilize the intention message, extension on the platooning application with a strategy to handle the message is required.

Moreover, the more common cut-in events are perhaps the ones happening at the on-ramps to highways, which is more time-critical due to the limited space in the acceleration lanes. Using V2V communication to communicate the intention as discussed above could be necessary to facilitate the cut-in, especially for long platoons. If we were to facilitate such cut-in, many questions need to be answered. For instance, *How should the platoon react to the intention message? Should the platoon make a gap to support the cut-in, how big gap will the human drivers accept?* The proposed C-ITS simulation framework has potential to be used as a tool to study such questions.

Last but not least, the behaviour of the manually driven car varies between different simulation runs and drivers, but the scenarios are fixed in this paper, i.e. the manually driven car always cut-in between the first and second vehicle in the platoon. Hence, other realistic traffic scenarios must also be considered, e.g. cut-in at different parts of a platoon, different time headway settings, etc. Moreover, using this simulation framework, human drivers' behaviour can be gathered to create a car-following model that includes the cut-in behaviour (as also desired in [13]).

VI. Conclusions

This paper presents the recent development of a C-ITS simulation framework consisting of driving-, traffic-, and network simulators. The development allows human drivers to drive in the C-ITS simulation framework, which was not previously possible. Allowing manually driven vehicles by human drivers in a C-ITS simulation enables vast future studies related to C-ITS applications that involve human drivers. One of such applications include studying the cut-in scenario by a manually driven vehicle in a platooning application as presented in this paper. Preliminary results on a collision case study suggest that a radar-based CACC controller showed promising performance compared to a V2V-based controller with respect to collision avoidance. Safety issues and future works related to the cut-in scenario are presented and discussed as the results.

Acknowledgment

The authors would like to thank Jonas Andersson Hultgren at VTI for his guidance and support on the driving simulation software. Also, Michele Segata from the University of Trento (Italy), for his valuable advices regarding modification and details of Plexe.

References

[1] C. Englund, L. Chen, J. Ploeg, E. Semsar-Kazerooni, A. Voronov, H. Hoang Bengtsson, and J. Didoff, "The Grand Cooperative Driving Challenge (GCDC) 2016 - boosting the introduction of Cooperative Automated Vehicles," *IEEE Wireless Communication Magazine*, no. August 2016 Issue, 2016.

[2] F. J. Ros, J. A. Martinez, and P. M. Ruiz, "A survey on modeling and simulation of vehicular networks: Communications, mobility, and tools ," *Computer Communications*, vol. 43, pp. 1 – 15, 2014.

[3] C. Englund, L. Chen, and A. Voronov, "Cooperative speed harmonization for efficient road utilization," in *2014 7th International Workshop on Communication Technologies for Vehicles, Nets4Cars-Fall 2014*, 2014, pp. 19–23.

[4] Y. Zhao, A. Wagh, Y. Hou, K. Hulme, C. Qiao, and A. W. Sadek, "Integrated traffic-driving-networking simulator for the design of connected vehicle applications: eco-signal case study," *Journal of Intelligent Transportation Systems*, pp. 1–13, 2014.

[5] M. Guériau, R. Billot, N.-E. E. Faouzi, J. Monteil, F. Armetta, and S. Hassas, "How to assess the benefits of connected vehicles? A simulation framework for the design of cooperative traffic management strategies ," *Transportation Research Part C: Emerging Technologies*, vol. 67, pp. 266 – 279, 2016.

[6] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler, and R. L. Cigno, "Plexe: A platooning extension for Veins," in *2014 IEEE Vehicular Networking Conference (VNC)*, Dec 2014, pp. 53–60.

[7] J. Larson, K. Y. Liang, and K. H. Johansson, "A Distributed Framework for Coordinated Heavy-Duty Vehicle Platooning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 419–429, Feb 2015.

[8] D. Jia and D. Ngoduy, "Platoon based cooperative driving model with consideration of realistic inter-vehicle communication," *Transportation Research Part C: Emerging Technologies*, vol. 68, pp. 245 – 264, 2016.

[9] L. Zhao and J. Sun, "Simulation Framework for Vehicle Platooning and Car-following Behaviors Under Connected-vehicle Environment," *Procedia - Social and Behavioral Sciences*, vol. 96, pp. 914 – 924, 2013.

[10] D. Jia, K. Lu, J. Wang, X. Zhang, and X. Shen, "A Survey on Platoon-Based Vehicular Cyber-Physical Systems," *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 263–284, Firstquarter 2016.

[11] S. E. Shladover, C. Nowakowski, X.-Y. Lu, and R. Ferlis, "Cooperative Adaptive Cruise Control: Definitions and Operating Concepts," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2489, pp. 145–152, 2015.

[12] M. Aramrattana, T. Larsson, and C. Jansson, Jonas Englund, "Dimensions of Cooperative Driving, ITS and Automation," in *Intelligent Vehicles Symposium (IV)*. Seoul: IEEE, 2015.

[13] V. Milanés and S. E. Shladover, "Handling Cut-In Vehicles in Strings of Cooperative Adaptive Cruise Control Vehicles," *Journal of Intelligent Transportation Systems*, vol. 20, no. 2, pp. 178–191, 2016.

[14] A. F. Larsson, K. Kircher, and J. A. Hultgren, "Learning from experience: Familiarity with ACC and responding to a cut-in situation in automated driving," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 27, Part B, pp. 229 – 237, 2014, vehicle Automation and Driver Behaviour.

[15] J. Jansson, J. Sandin, B. Augusto, M. Fischer, B. Blissing, and L. Källgren, "Design and performance of the VTI Sim IV," in *Driving Simulation Conference*, 2014.

[16] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent Development and Applications of SUMO - Simulation of Urban MObility," *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, pp. 128–138, December 2012.

[17] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, January 2011.

[18] A. Wegener, M. Piórkowski, M. Raya, H. Hellbrück, S. Fischer, and J.-P. Hubaux, "TraCI: An Interface for Coupling Road Traffic and Network Simulators," in *Proceedings of the 11th Communications and Networking Simulation Symposium*, ser. CNS '08. New York, NY, USA: ACM, 2008, pp. 155–163.

[19] M. Aramrattana, T. Larsson, J. Jansson, and A. Nåbo, "Extended Driving Simulator for Evaluation of Cooperative Intelligent Transport Systems," in *Proceedings of the 2016 Annual ACM Conference on SIGSIM Principles of Advanced Discrete Simulation*, ser. SIGSIM-PADS '16. New York, NY, USA: ACM, 2016, pp. 255–258.

[20] ——, "Cooperative Driving Simulation," *The Driving Simulation Conference 2016 VR*, 2016.

[21] R. Rajamani, *Vehicle dynamics and control*, 2nd ed. Springer US, 2012.

[22] 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 *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Oct 2011, pp. 260–265.

[23] 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," 2012.

Appendix D

Paper IV
Safety Analysis of Cooperative Adaptive Cruise Control in Vehicle Cut-in Situations

Maytheewat Aramrattana, Tony Larsson, Jonas Jansson, and Arne Nåbo

2017 4th International Symposium on Future Active Safety Technology towards Zero-Traffic-Accidents (FAST-zero)
Nara, Japan, September 18-22, 2017

(Link to the publication)

Safety Analysis of Cooperative Adaptive Cruise Control in Vehicle Cut-in Situations

Maytheewat Aramrattana[1,3], Cristofer Englund[1,2], Jonas Jansson[3], Tony Larsson[1], and Arne Nåbo[3]

1) The School of Information Technology, Halmstad University
PO Box 823, SE-301 18, Halmstad, Sweden (E-mail: maytheewat.aramrattana@hh.se)
2) RISE Viktoria
Lindholmspiren 3A, SE-417 56 Göteborg, Sweden
3) The Swedish National Road and Transport Research Institute (VTI),
VTI, SE-581 95 Linköping, Sweden

ABSTRACT: Cooperative adaptive cruise control (CACC) is a cooperative intelligent transport systems (C-ITS) function, which especially when used in platooning applications, possess many expected benefits including efficient road space utilization and reduced fuel consumption. Cut-in manoeuvres in platoons can potentially reduce those benefits, and are not desired from a safety point of view. Unfortunately, in realistic traffic scenarios, cut-in manoeuvres can be expected, especially from non-connected vehicles. In this paper two different controllers for platooning are explored, aiming at maintaining the safety of the platoon while a vehicle is cutting in from the adjacent lane. A realistic scenario, where a human driver performs the cut-in manoeuvre is used to demonstrate the effectiveness of the controllers. Safety analysis of CACC controllers using time to collision (TTC) under such situation is presented. The analysis using TTC indicate that, although potential risks are always high in CACC applications such as platooning due to the small inter-vehicular distances, dangerous TTC (TTC < 6 seconds) is not frequent. Future research directions are also discussed along with the results.

KEY WORDS: cooperative adaptive cruise control, modelling and simulation

1. Introduction

Nowadays, there is extensive research on connected and automated vehicles. Higher levels of automation are being added to vehicles, as well as the capability for vehicles to be connected via wireless communication, which often is referred to as vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. By using wireless communication technologies, automated vehicles interact with each other and parts of the road infrastructure, forming cooperative intelligent transport systems (C-ITS). Expected benefits from such systems are massive in terms of improved energy efficiency, safety, and sustainability of the transport systems. As soon as 2019, European Commission is planning to deploy sets of Day 1, and Day 1.5 C-ITS services [(1)]. Apart from Europe, projects aiming towards C-ITS deployment can be seen around the world, e.g. the Connected Vehicle Pilot Deployment Program in USA[(2)], and other projects in Korea, Australia, etc.

Cooperative adaptive cruise control (CACC) is one example of a C-ITS function, which has the potential to be deployed in the near future. One application of CACC is platooning, where vehicles automatically follow each other with small inter-vehicular distance. A string of such vehicles is often referred to as a “platoon”. The goals of platooning are, for example, to make more efficient use of the road space by reducing the inter-vehicular distance, and to reduce fuel consumption by the reduced aerodynamic drag obtained by the short distance between the vehicles. In theory, several control strategies for CACC have been proposed. CACC controllers are mainly designed to enable small-inter vehicular distance, while maintaining string stability (a platoon is string stable when the effects from control errors, or disturbances, do not amplify backwards to the followers in the platoon). Details regarding definitions and operation concepts of CACC is described in [(3)]. Furthermore, a few highway platooning scenarios have been demonstrated such as in the Grand Cooperative Driving Challenge [(4)(5)], and the European Truck Platooning Challenge [(6)].

In most CACC-related literature, a homogeneous platoon is often assumed, i.e. all vehicles in the platoon are identical, connected, and automated. However, during the early deployment phase of C-ITS, there will be a mixture of vehicles with different automation and connectivity capabilities driving in the same traffic environment. Furthermore, in real traffic scenarios, cut-in manoeuvres can be frequently expected, both from connected and non-connected vehicles. Cut-in manoeuvres can potentially reduce the benefits of platooning, and are thus not desired from a safety perspective. In spite of that, cut-in scenarios are not often considered in research related to CACC or platooning. To the authors’ knowledge, only a few publications in the area have considered cut-in manoeuvres in their studies. For instance, the work from *Milanés and Sladover* [(7)] proposed an approach to handle the cut-in by switching the controller to adaptive cruise control (ACC) when the cut-in is detected. On road experiments as well as simulation results are presented in their paper, analysing effects of the cut-in on string stability. From human driver’s perspective, *Larsson et al.* [(8)] studied response time of the human driver under a cut-in situation. Even though the study was with ACC-equipped vehicle, learning how the human drivers would react to the cut-in is of interest.

Therefore, this paper presents safety analysis of two different CACC controllers in a simulated cut-in scenario on a highway. During the scenario, depicted in Fig. 1, a non-connected vehicle performs a cut-in manoeuvre into the gap between the first, and the

Copyright © 2017 Society of Automotive Engineers of Japan, Inc. All rights reserved

second vehicle of the platoon. The non-connected vehicle is driven by a human driver using the driving simulation software from the Swedish National Road and Transport Research Institute (VTI). Data from seven participants are collected from the simulations.

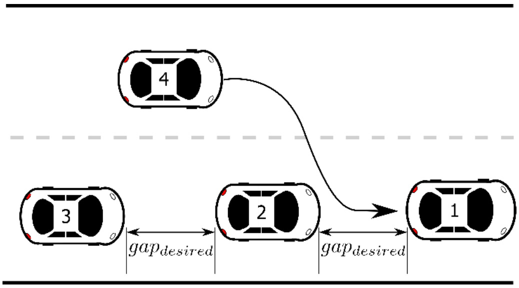


Fig. 1 A vehicle cut-in scenario, where vehicle number 4 is cutting in between vehicle 1 and 2.

The rest of the paper is organized as follows. Section 2 describes the simulation framework used in this study, along with details about the two CACC controllers. Set up of the simulation study is explained in Section 3. The results are presented in Section 4. Section 5 presents the safety analysis with the discussions in Section 6. Finally, Section 7 concludes the paper.

2. Background

The C-ITS simulation framework used in this paper has been developed previously [9][10]. The framework is a combination of driving-, traffic-, and network simulators. As mentioned above, the driving simulation software is developed by VTI. It is a driving simulation software, which can be run either on a desktop computer, or a complete moving-base driving simulator [11]. For the traffic and network simulators, *The Platooning Extensions for Veins* (Plexe) [12] is used. Plexe is a traffic and network simulation framework, based on the traffic simulator, *Simulation of Urban Mobility* (SUMO), and the network simulator, *Vehicle in Network Simulation* (Veins). Plexe extends SUMO and Veins with support for studying platooning applications. An overview of the C-ITS simulation framework and existing models are illustrated in Fig. 2.

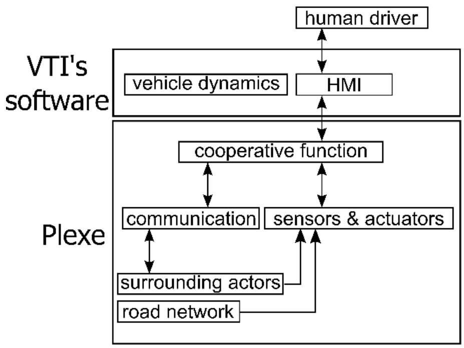


Fig. 2 Overview of the C-ITS simulation framework.

In the C-ITS simulation framework, the human driver can either drive manually or visualize the automated vehicle from the driver perspectives. Moreover, the human driver can interact with the other vehicle in the simulation, e.g. pushing a button on the steering wheel to send V2V communication messages. The rest of the vehicles are automatically controlled by car-following models in SUMO. For simulation of platooning scenarios, Plexe added a number of car-following models to SUMO, including the implementation of two existing CACC controllers.

The first controller defined in Chapter 7 of [13], will be referred to as the *Rajamani* controller in this paper. The second CACC controller was proposed in [14], and will be referred to as *Ploeg* controller in this paper. The two controllers have different control strategies, and rely on different information sources, as summarized in Table 1. The *Rajamani* controller uses a “constant distance gap” strategy, which means that the controller is designed to maintain a desired inter-vehicular distance to the vehicle in front. On the other hand, the *Ploeg* controller uses a “constant time headway” strategy. The time headway is defined as the time gap between when the rear bumper of the preceding vehicle, and the front bumper of the ego vehicle, reach the same fixed point on the road. In this case, the actual inter-vehicular distance depends on the speed of the ego vehicle. Moreover, as listed in Table 1, the *Rajamani* controller relies more on information exchanged via V2V communication, while the *Ploeg* controller, to a larger extent, trust on information from the radar.

All vehicles are assumed to be equipped with a forward-looking radar, which will detect an object that is in front of it in the same lane. Lane changing in SUMO is discrete, i.e. vehicles instantaneously switch from one lane to another. Such lane changing decisions are determined by the position of the front axle of the ego vehicle in the driving simulation. In other words, as soon as the front axle of the ego vehicle crosses the lane marking, the corresponding vehicle in SUMO is switched to another lane. Furthermore, the platooning vehicles (vehicle no. 1,2, and 3) are assumed to be equipped with V2V communication modules according to the IEEE 802.11p standard.

Table 1 Control parameters of the two existing CACC controllers in Plexe.

	Controller	
Control parameters	*Rajamani*	*Ploeg*
Control strategy	distance	time headway
Platoon leader’s speed	V2V	-
Platoon leader’s acceleration	V2V	-
Preceding vehicle’s speed	V2V	Radar
Preceding vehicle’s acceleration	V2V	V2V
Distance to preceding vehicle	Radar	Radar

3. Simulation Setup

Seven participants have participated in the study. Upon arrival, each participant was given time to familiarize with controlling the vehicle in the driving simulation software. Then, the scenario is explained to the participant.

According to Fig. 1, the participant is driving the vehicle no. 4, and is asked to perform a cut-in between vehicle no. 1 and 2. Vehicle no. 1, 2, and 3 are connected and automated vehicles driving with CACC functionality. The leader of the platoon

Copyright © 2017 Society of Automotive Engineers of Japan, Inc. All rights reserved

(vehicle no.1) drives at a constant speed of 90 km/h. Two existing CACC controllers in Plexe were used with two different desired inter-vehicular distances of 30 meters (1.2 seconds headway at 90 km/h) and 17.5 meters (0.7 seconds headway at 90 km/h).

Fig. 3 The setup for driving simulation.

Two desktop computers were used to run the simulation framework, one for the Plexe framework, and the other for the driving simulation software. The driving simulation setup for the participants is depicted in Fig. 3. The simulation is run for 100 seconds, and data is collected both from the Plexe framework and from the driving simulator. At each simulation run, the same CACC controller and desired distance is chosen for all vehicles in the platoon. Each participant has driven each scenario at least once.

4. Cut-in Manoeuvres

Fig. 4 and Fig. 5 illustrate all cut-in manoeuvres recorded by the driving simulation software, and the average cut-in manoeuvres from the recorded data, respectively. The plots show five seconds before and after each cut-in occurs. The horizontal black lines, which is drawn at the lateral position approximately -2, 2, and 6 meters, represents the position of the lane marking.

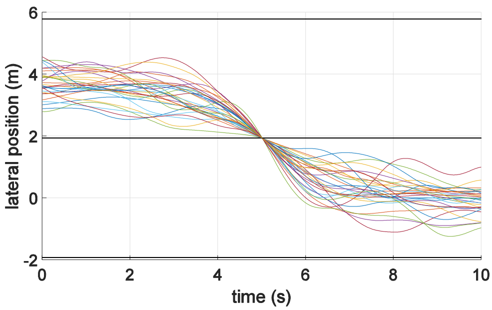


Fig. 4 Traces from all participants during lane change.

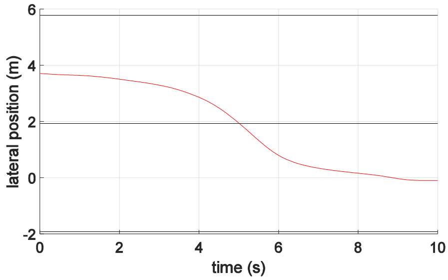


Fig. 5 Average of the traces from all participants during lane change.

Furthermore, the results show that the inter-vehicular distance of the platoon seems to have effects on the human cut-in behaviour.

All cut-in manoeuvres are then divided into two groups based on the scenario, i.e. when the desired inter-vehicular distance of the platoon 30 meters, and 17.5 meters. All cut-in manoeuvres when the platoon's inter-vehicular distance is 30 meters, and 17.5 meters are depicted in Fig. 6, and Fig. 8, respectively. The average of the cut-in manoeuvres for the 30 meters distance, and 17.5 meters distance, are illustrate in Fig. 7, and Fig. 9, respectively. According to the figures, a few participants tend to be closer to the lane marking before performing the cut-in when the inter-vehicular gap is 17.5 meters.

Table 2 Summary of the distances measured from vehicle no.2 to the vehicle no.4 when the cut-in occurs.

CACC gap / Cut-in distance (m)	30m (n = 13)	17.5m (n = 13)
Maximum	18.96	9.86
Minimum	2.73	2.26
Mean	10.87	5.99
SD	5.10	2.45

Table 3 Summary of the distances measured from the ego vehicle (no.4) to the vehicle no.1 when the cut-in occurs.

CACC gap / Cut-in distance (m)	30m (n = 13)	17.5m (n = 13)
Maximum	23.28	11.24
Minimum	7.04	3.64
Mean	15.53	7.59
SD	4.85	2.49

Copyright © 2017 Society of Automotive Engineers of Japan, Inc. All rights reserved

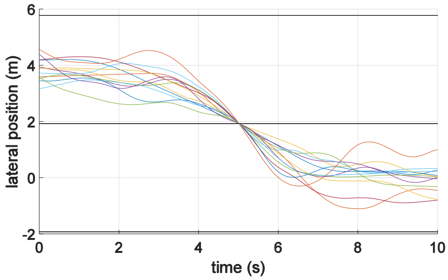


Fig. 6 Traces of vehicles performing cut-in manoeuvres when the platoon's desired inter-vehicular distance is 30 meters.

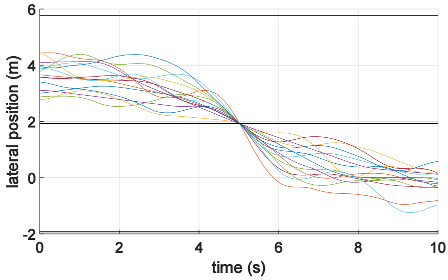


Fig. 8 Traces of vehicles performing cut-in manoeuvres when the platoon's desired inter-vehicular distance is 17.5 meters.

Results in Table 2 and Table 3 suggest that, on average, the drivers leave more gap in front than behind the ego vehicle while performing a cut-in manoeuvre.

5. Safety Analysis

In traffic systems, time to collision (TTC) is often used as a safety indicator [15]. The following definition of TTC will be used in this paper.

$$\mathrm{TTC} = \frac{X_{i-1}(t) - X_i(t) - l_i}{\dot{X}_i(t) - \dot{X}_{i-1}(t)} \qquad \forall\, \dot{X}_i(t) > \dot{X}_{i-1}(t) \qquad (1)$$

Where X_i is the position of vehicle i, $\dot{X}_i$ is the speed of vehicle i, and l_i is the length of vehicle i. The ego vehicle is indicated with the index i, and the preceding vehicle is indicated with the index $i-1$. In this case, vehicle no. 2 is the vehicle i and vehicle no. 4 is the vehicle $i-1$. The length of all vehicles is 4 meters.

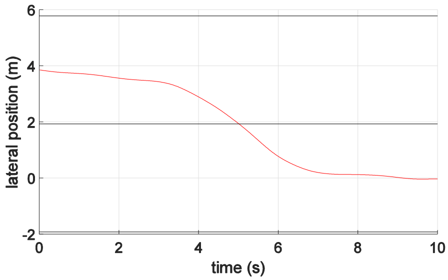


Fig. 7 Average trace of vehicles performing cut-in manoeuvres when the platoon's desired inter-vehicular distance is 30 meters.

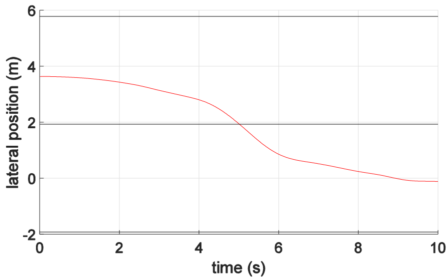


Fig. 9 Average trace of vehicles performing cut-in manoeuvres when the platoon's desired inter-vehicular distance is 17.5 meters..

According to [15], a TTC more than 6 seconds can be considered as safe. In other words, TTC more than 6 seconds is enough to ensure that the follower will not be forced to perform any dangerous avoidance manoeuvres, even with a standstill obstacle (unless the vehicle is driving faster than 130 km/h). Among all collected datasets there are 2 cases, where the TTC is less than 6 seconds, one for each controller when the desired inter-vehicular distance is set to 17.5 meters. Moreover, in these two situation, the TTC is less than 3 seconds, which is enough to trigger the emergency braking system, according to [16] ("*The emergency braking phase shall not start before TTC is equal to, or less than, 3.0 seconds*"). The following subsections describe the scenarios with the two different controllers.

5.1 The *Rajamani* Controller

The inter-vehicular distance measured from vehicle number 2 when the cut-in occurs is 2.37 meters, and the distance is 11.19 meters measured from the ego vehicle (vehicle number 4). The measured distances, and speed are presented in Fig. 11, and Fig. 12, respectively. Fig. 10 shows the TCC after the cut-in occurs at 44.9 seconds.

Copyright © 2017 Society of Automotive Engineers of Japan, Inc. All rights reserved

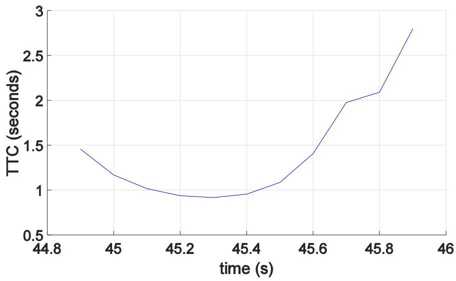


Fig. 10 The TTC when the vehicle no. 2 is operated by the *Rajamani* controller.

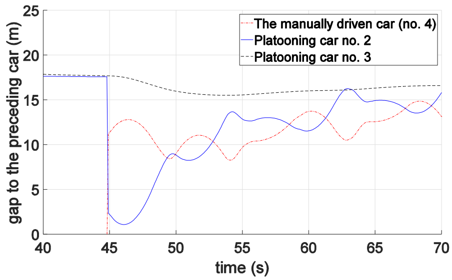


Fig. 11 Plots of the distances measured from vehicle no. 2, 3, and 4 during the cut-in. Vehicle 2 and 3 are operated by the *Rajamani* controller.

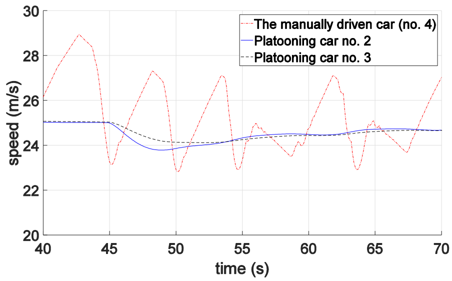


Fig. 12 Speed of vehicle no. 2, 3, and 4 during the cut-in. Vehicle no. 2 and 3 are operated by the *Rajamani* controller.

5.2 The *Ploeg* Controller

In this case, the inter-vehicular distance measured from vehicle number 2 when the cut-in occurs is 2.26 meters.

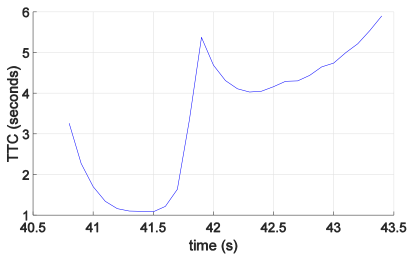


Fig. 13 The TTC when the vehicle no. 2 is operated by the *Ploeg* controller.

The distance measured from the ego vehicle is 11.24 meters. The measured distance, and speed during the scenario are presented in Fig. 14, and Fig. 15, respectively. Fig. 13 shows the TCC after the cut-in occurs at 40.8 seconds. This short TTC period is longer than that of the *Rajamani* controller.

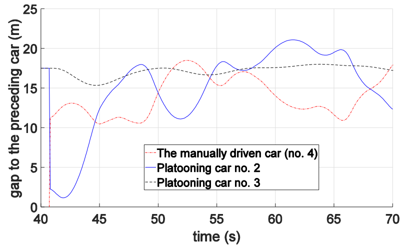


Fig. 14 Plots of the distances measured from vehicle no. 2, 3, and 4 during the cut-in. Vehicle no. 2 and 3 are operated by the *Ploeg* controller.

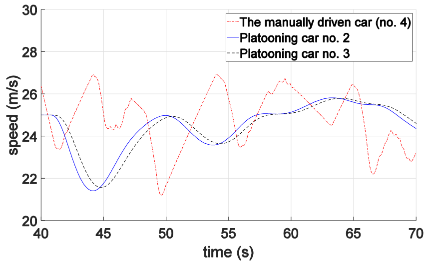


Fig. 15 Speed of the vehicle no. 2, 3, and 4 during the cut-in. Vehicle no. 2 and 3 are operated by the *Ploeg* controller.

6. Discussion

Although not considered in this study, an emergency braking system can be triggered at a TTC less than 3 seconds, as mentioned above. With sufficient V2V communication coverage, a coordinated emergency braking manoeuvre can be used to ensure safety within the platoon. However, such emergency braking may compromise safety of other road users, which are not aware of the emergency braking manoeuvre.

As pointed out in [(15)], TTC and time headway are different. A short time headway indicates potential danger; while a small TTC represents actual danger. In an application like platooning, where vehicles are driving with small inter-vehicular distance, time headway is almost always small. The 30 meters inter-vehicular distance, corresponds to the time headway of 1.2 seconds, can be considered as small. This is comparable to the time headway of adaptive cruise control (ACC) functions that are available nowadays. Furthermore, 0.7 seconds time headway, which corresponds to 17.5 meters inter-vehicular distance, indicates more potential danger. Such short time headway can sometimes lead to small TTC, as presented in the results in Section 5, for instance.

The two controllers used as examples are certainly not designed to handle cut-in manoeuvres as their main goal. Furthermore, one can observe different behaviours among the two controllers in Fig. 11, and Fig. 14. The *Ploeg* controller reacts faster to maintain the desired inter-vehicular distance, while on the other hand, the *Rajamani* controller seems to take longer time to recover after the

Copyright © 2017 Society of Automotive Engineers of Japan, Inc. All rights reserved

cut-in. However, such behaviours are heavily influenced by the speed variation of the human driver after the cut-in, which is not the same in this case. Thus, to carefully analyse the controllers' behaviour, a more repeatable cut-in manoeuvre is required. For instance, by replaying the same logged data to different controllers.

Nevertheless, both controllers perform well to handle the situations and prevent a collision, given that the controllers are not aware that the cut-in occurs. One solution to handle the cut-in is to switch to ACC, when the cut-in is detected, as suggested in [7], which is reasonable given that there is no way to anticipate or detect the cut-in manoeuvre before it occurs.

In this study, regardless of the cut-in manoeuvre, the controllers have the same perception, i.e. the sudden change detected in the radar measurements when a cut-in occurs. If the platoon can be aware of the cut-in. For example, if vehicle no. 4 has connectivity, but no automation capabilities. Scenarios where a request message is sent before the cut-in, similar to the usage of turning indicators, can be simulated. Consequently, as future work, benefits of having such information beforehand can be studied. Also, different algorithms to handle the request message and react to it in a safe and efficient manner can be developed.

The driving simulation setup in this study is arguably simple, because the main test subject is the CACC controllers, and the human driver is acting as disturbance to the platoon. As future work, running the same study on a moving-base driving simulator such as Sim IV (see Fig. 16), research questions related to human factors can be studied. For instance, *how big inter-vehicular gap is acceptable to the human driver?* Moreover, cut-in manoeuvres can be collected, and used to create a realistic car-following model with cut-in manoeuvre for testing CACC controllers.

Fig. 16 The Sim IV, moving-base driving simulator located at the VTI site in Gothenburg, Sweden.

At the beginning of the simulation, the manually driven vehicle starts at zero speed, while the platoon starts at 100 km/h, thus the participants need to accelerate and catch up with the platoon. Therefore, large time variation can be observed in Table 4, because each participant has their own approach to control the vehicle. In all cases, the participants took approximately 42-43 seconds on average, until he or she performs the first cut-in manoeuvres in between the vehicle no. 1 and no. 2, as presented in the Table 4.

Table 4 Summary of the time taken before the participants perform the first cut-in manoeuvres.

CACC gap / Cut-in time (s)	all	30m	17.5m
Maximum	73.54	71.29	73.54
Minimum	29.26	29.26	30.39
Mean	42.41	43.59	42.44
SD	10.78	11.52	11.05

7. Conclusions

This study explores behaviour of two different CACC controllers under a cut-in scenario by a non-connected vehicle. The non-connected vehicle is driven by human driver via a desktop driving simulator. Results from seven participants are presented in the paper.

Safety analysis of the two CACC controllers using TTC is discussed. Although, potential risks are always high in CACC applications due to the small inter-vehicular distance, dangerous TTC (TTC < 6 seconds) is not commonly observed. This is perhaps because of the CACC controllers' speed regulation; or TTC is not the best measure to analyse safety of CACC operations. However, under this specific scenario, the TTC results indicate that the two controllers can handle the situation fairly well, even though they are not designed specifically to do so.

Several future research directions are discussed in Section 6. However, the authors' main interest is to find a methodology for determining safe operation of CACC controller in mixed traffic scenarios. Thus, this study can be regarded as a pilot study towards a safety analysis framework for evaluating safety of the operations of CACC controllers in mixed traffic scenarios. In the future work, the scenario will be extended to include, e.g. speed variation of the platoon leader, V2V communication capability on the cut-in vehicle, realistic driving simulator setup, etc.

References

(1) European Commission, "COM/2016/0766, A European strategy on Cooperative Intelligent Transport Systems, a milestone towards cooperative, connected and automated mobility," 2016.
(2)"Connected Vehicle Pilot Deployment Program (https://www.its.dot.gov/pilots/)."
(3) S. E. Shladover, C. Nowakowski, X.-Y. Lu, and R. Ferlis, "Cooperative Adaptive Cruise Control: Definitions and Operating Concepts," *Transp. Res. Rec. J. Transp. Res. Board*, no. 2489, pp. 145–152, 2015.
(4) J. Ploeg, S. Shladover, H. Nijmeijer, and N. van de Wouw, "Introduction to the special issue on the 2011 grand cooperative driving challenge," *Intell. Transp. Syst. IEEE Trans.*, vol. 13, no. 3, pp. 989–993, 2012.
(5) C. Englund *et al.*, "The Grand Cooperative Driving Challenge (GCDC) 2016 - boosting the introduction of Cooperative Automated Vehicles," *IEEE Wirel. Commun. Mag.*, no. August 2016 Issue, 2016.
(6)"European Truck Platooning Challenge (https://www.eutruckplatooning.com)."
(7) V. Milanés and S. E. Shladover, "Handling Cut-In Vehicles in Strings of Cooperative Adaptive Cruise Control

Copyright © 2017 Society of Automotive Engineers of Japan, Inc. All rights reserved

Vehicles," *J. Intell. Transp. Syst.*, vol. 20, no. 2, pp. 178–191, 2016.
(8) A. F. L. Larsson, K. Kircher, and J. A. Hultgren, "Learning from experience: Familiarity with ACC and responding to a cut-in situation in automated driving," *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 27, Part B, pp. 229–237, 2014.
(9) M. Aramrattana, T. Larsson, J. Jansson, and A. Nåbo, "Extended Driving Simulator for Evaluation of Cooperative Intelligent Transport Systems," in *Proceedings of the 2016 Annual ACM Conference on SIGSIM Principles of Advanced Discrete Simulation*, 2016, pp. 255–258.
(10) M. Aramrattana, T. Larsson, J. Jansson, and A. Nåbo, "Cooperative Driving Simulation," in *The Driving Simulation Conference 2016 VR*, 2016.
(11) J. Jansson, J. Sandin, B. Augusto, M. Fischer, B. Blissing, and L. Källgren, "Design and performance of the VTI Sim IV," in *Driving Simulation Conference*, 2014.
(12) M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler, and R. L. Cigno, "Plexe: A platooning extension for Veins," in 2014 IEEE Vehicular Networking Conference (VNC), 2014, pp. 53–60.
(13) R. Rajamani, Vehicle dynamics and control, 2nd ed. Springer US, 2012.
(14) 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 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), 2011, pp. 260–265.
(15) K. Vogel, "A comparison of headway and time to collision as safety indicators," Accid. Anal. Prev., vol. 35, no. 3, pp. 427–433, 2003.
(16) 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 emer." 2012.

Copyright © 2017 Society of Automotive Engineers of Japan, Inc. All rights reserved

Appendix E

Paper V
Safety Evaluation of Highway Platooning Under a Cut-In Situation Using Simulation

Maytheewat Aramrattana, Cristofer Englund, Tony Larsson, Jonas Jansson, and Arne Nåbo

IEEE Transactions on Intelligent Transportation Systems
August 2018 (Submitted)

Safety Evaluation of Highway Platooning Under a Cut-In Situation Using Simulation

Maytheewat Aramrattana[1,2], Cristofer Englund[1,3], Tony Larsson[1], Jonas Jansson[2],and Arne Nåbo[2]
[1]School of Information Technology, Halmstad University, Sweden
[2]The Swedish National Road and Transport Research Institute (VTI), Sweden
[3]RISE Viktoria, Göteborg, Sweden

Platooning refers to an application, where a group of connected and automated vehicles follow a lead vehicle autonomously, with short inter-vehicular distances. At merging points on highways such as on-ramp, platoons could encounter manually driven vehicles, which are merging on to the highways. In some situations, the manually driven vehicles could end up between the platooning vehicles. Such situations are expected and known as "cut-in" situations. This paper presents a simulation study of a cut-in situation, where a platoon of five vehicles encounter a manually driven vehicle at a merging point of a highway. The manually driven vehicle is driven by 37 test persons using a driving simulator. For the platooning vehicles, two longitudinal controllers with four gap settings between the platooning vehicles, i.e. 15 meters, 22.5 meters, 30 meters, and 42.5 meters, are evaluated. Results summarizing cut-in behaviours and how the participants perceived the situation are presented. Furthermore, the situation is assessed using safety indicators based on time-to-collision.

***Index Terms*—driving simulator, highway platooning, cut-in, cooperative adaptive cruise control, safety evaluation, time-to-collision.**

I. Introduction

In the recent years, more efforts are put on realization of Cooperative Intelligent Transportation Systems (C-ITS) applications in real traffic scenarios, especially highway platooning. Although the highway platooning concept has been proposed and demonstrated since the 1990s as presented in [1], [2], [3], it has just recently reached the point where it is being tested on public roads, for instance in The European Truck Platooning Challenge 2016. This due to advancements in wireless communication and vehicle automation. A platoon usually refers to a group of vehicles, which are autonomously following each other with short inter-vehicular distances. This group of vehicles will be referred to as *platooning vehicles* hereafter. Typically controlled by Cooperative Adaptive Cruise Control (CACC) functions [4], platooning vehicles utilize information from wireless communication and the vehicles' sensors to automate (longitudinal) speed of the vehicles. Although platooning may also include automation in the lateral dimension [5], this paper will only consider platoons with longitudinal automation. Meaning that the CACC controller automates longitudinal control, and each vehicle is responsible for itself (as opposed to the centralized approach where one controller automates the whole platoon). Thus, in this paper, *platooning* refers to the aforementioned type of platoon, controlled by CACC controllers.

The expected goals of platooning include improving safety and driver comfort, more efficient use of road space, improved fuel efficiency of platooning vehicles, as also listed in [4]. Benefits of platooning on traffic flow have been predicted using simulation in [6]. They are reported in [7], which concludes that CACC can potentially improve traffic throughput and increase highway capacity near a lane drop, especially with more than 60% CACC-penetration rate in the traffic. Another publication [8] also suggests that CACC can promptly damp shockwaves in traffic. Potentials in saving fuel are also extensively studied, especially in platoons of heavy duty vehicles, e.g. the articles [9], [10], [11], [12] have suggested that the possible fuel reduction can be in the range of 3-12%. However, safety issues related to platooning are not as often studied as summarized in [13].

To achieve the platooning goals above on public roads, one of the important tasks is to ensure that platoons are sufficiently safe to operate in real traffic situations. Unless platoons have their own dedicated lane, they have to share roads with other vehicles that may not have any means of communication and/or automation. Hence, it is necessary to investigate how the platoons react to the situations, and how their behaviours are perceived by other road users when they encounter each other on the road. An example is the cut-in situation, which can be frequently expected, especially at a merging point such as on-ramps to highways. Cut-in situations can potentially reduce benefits of platooning, and are thus not desired from a safety point of view. Also, according to [13], the cut-in situations are commonly mentioned as hazards in platooning.

Therefore, this paper focuses on safety evaluation of a highway cut-in situation by a manually driven vehicle. Two existing CACC controllers are investigated in a driving simulation study with human participants driving the manually driven vehicle. We use simulation in this work because of the risky situations that we are presenting to the participants, and due to the difficulties to obtain a fleet of CACC-equiped vehicles. An existing simulation framework [14], which integrates driving, network, and traffic simulators is used. The contributions of this paper are as follows:

- Results from simulation of a cut-in scenario at an on-ramp of a highway, where a manually driven vehicle encounters

Corresponding author: Maytheewat Aramrattana (email: maytheewat.aramrattana@hh.se).

- a platoon at four different inter-vehicular gap settings: 15, 22.5, 30, and 42.5 meters.
- Results from a questionnaire on perceived safety collected during the simulation study. The results are from 37 participants, who experienced the scenario with all settings.
- A safety analysis on the collected data using Time-to-collision (TTC) and its extensions.

The rest of the paper is organized as follows. Section II presents state-of-the-art related to analysis of cut-in situations. Section III describes the simulation tool and scenario. Details about the experiment and participants are explained in Section IV. Section V presents the results, while the safety analysis using TTC is presented in Section VI. The scenario, results, and analysis are discussed in Section VII. Lastly, the future work are presented in the Section VIII, and the paper concludes in Section IX.

II. Related Work

Despite a lot of research related to the highway platooning application, there are not much work that focus on highway platooning with cut-in or merging situations. A selection of them will be mentioned in this section to depict the state-of-the-art in this area.

Our study focuses on scenarios where the cut-in vehicle is manually driven and is not communicating with surrounding vehicles, because this is likely to happen in the early deployment phases, when the penetration rate of CACC-equipped vehicles is low. Also, because this kind of scenarios is not often considered in current research. Nevertheless, the cut-in by non-connected vehicles is studied in [15], [16], [17], [18], [19]. In [15], the CACC is tested on public roads with four experimental vehicles. Although the CACC controller in the vehicles can have 0.6, 0.9, and 1.1 seconds gap settings, the cut-in was tested at the 1.1 seconds gap setting due to safety reasons. Similar tests were reported in [16], with additional simulation results and different length of the platoon. An impact on multi-lane highway capacity at a merging point is reported in [17]. The study used microscopic traffic simulation to model CACC and manually driven vehicles. The results were presented with respect to the market penetration rate of CACC-vehicle from 0-100%. We have presented studies related to cut-in scenarios in [18], [19], which considered cut-ins from an adjacent lane on a highway. The number of participants in the two previous papers are lower, and the driving simulation set up is less realistic compared to the one presented in this paper.

On the other hand, in the case when the merging vehicle is a CACC-equipped vehicle, there are existing investigations and suggested solutions in the literature. For examples, CACC in highway merging control is simulated, analysed, and compared to Adaptive Cruise Control (ACC) in [20]. As suggested in [7], having an infrastructure at the merging point that informs the platoon to enlarge the gap, hence coordinates the merging manoeuvre. Adding a "cooperative-merging" application to the CACC system is also suggested in the same paper. The findings in [15], [16] suggested that the CACC-equipped cut-in vehicles become part of an existing platoon after the cut-in.

Regarding safety evaluation of platooning applications, a recent survey [13] summarized the topic well. However, to the authors' knowledge, there is a lack of studies dedicated to the safety evaluation of cut-in situations. Therefore, safety indicators used in this paper are inspired by the proposed road traffic safety indicators, namely Time-to-collision (TTC). TTC is first introduced in [21], it can be used as a safety indicator as suggested in [22]. Moreover, this paper will also considers extensions of TTC such as Time Exposed Time-to-collision (TET) and Time Integrated Time-to-collision (TIT) [23]. These safety indicators were used in a platooning safety evaluation in [24], where the safety of a dedicated lane for platoon is investigated.

III. The Simulation

A. Simulation Tools

The simulation framework used in this work, presented in [14], includes driving, network, and traffic simulators for realistic simulation of C-ITS scenarios, especially platooning applications.

This study used the moving-base driving simulator, "Sim IV", at the Swedish National Road and Transport Research Institute (VTI) located in Gothenburg, Sweden. The driving simulation software kernel is developed in-house at VTI. In the *Sim IV*, the cabin inside can be a passenger car, or a truck. The passenger car cabin (Volvo XC60) is used in this study. *Sim IV* provides more than 180 degrees forward field of view, which gives a more realistic driving experience compared to a desktop simulator, as used in the previous studies [18], [19]. The moving base was not used in this study because the motion cues did not add any value in this specific scenario. More details about the *Sim IV* can be found in [25].

Vehicle in Network Simulation (Veins) [26] is used as a network simulator, and *Simulation of Urban MObility* (SUMO) [27] is used as a traffic simulator. The version proposed by the author of Plexe—Platooning Extension for Veins [28]—is used in this work. Hereafter *plexe-veins* will refer to the Plexe version of Veins, and *plexe-sumo* will refer to the Plexe version of SUMO. Plexe adds CACC controllers as car-following models into SUMO, and enable a way to interact with the controllers through the TraCI interface, which is an existing protocol for on-line interaction with SUMO simulations [29]. As an addition to Veins, Plexe implements a wireless communication interface according to IEEE 802.11p standard. Please refer to [28] for a more detailed description.

Although all vehicles are present in both the VTI's driving simulator and SUMO, the vehicles are divided into two groups: platooning vehicles, which are controlled by SUMO; and the cut-in ego vehicle, which is controlled by the driving simulator. Figure 1 illustrates which simulator is responsible for controlling which vehicle(s) in the simulation, marked by colour of the vehicles. Green vehicles (on top) are controlled by CACC controllers defined in SUMO, while the white vehicle (on the bottom) is controlled by a person in the driving simulator.

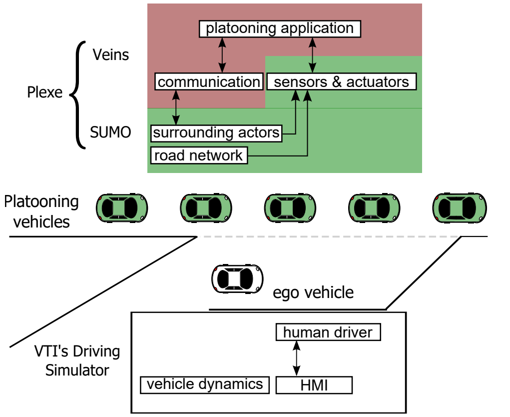
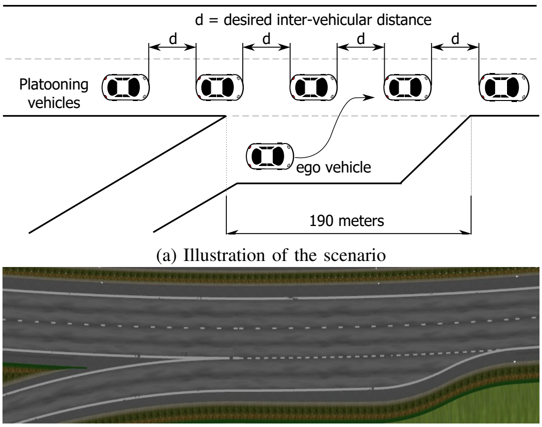


Fig. 1: Role of each simulators in the highway cut-in scenario.

B. Simulation Scenario – Highway Cut-in Scenario

This study investigates a scenario when a manually driven vehicle encounters a platoon at a merging point on a two-lane highway. The *manually driven vehicle* or *ego vehicle* refers to the vehicle in the simulation, that is driven by the participants in this study. The ego vehicle starts from standstill at an one-lane on-ramp, which leads to the highway. After that, when the ego vehicle has reached a certain speed and position on the road, a platoon of five vehicles is released on the rightmost lane of the highway. The ego vehicle then encounters the platoon at the merging point, as illustrated in Fig. 2a. These five vehicles in the platoon will hereafter be referred to as *platooning vehicles*.



(a) Illustration of the scenario

(b) Top view of the merging road in the driving simulator

Fig. 2: The highway cut-in scenario

In Plexe, the platooning vehicles are either controlled by *a*) the CACC controller proposed by *Rajamani* [30, Chapter 7]; or *b*) the CACC controller presented by *Ploeg et al.* in [31], these will be referred to as the *Rajamani* and *Ploeg* controllers respectively for the rest of this paper.

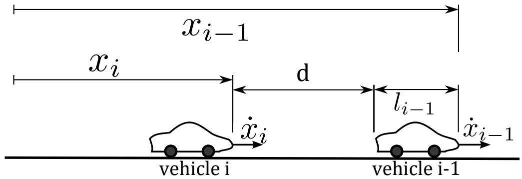


Fig. 3: Notation for two vehicles following each others.

Moreover, four desired inter-vehicle gaps were selected for the platooning vehicles. Therefore, there are eight different combinations as summarized in Table I. In the table, the gap settings described the desired inter-vehicular gap, "d", between platooning vehicles, which is measured from the front bumper of the ego vehicle to the rear bumper of the preceding vehicle, as illustrated in Fig. 2a. The time headway indicates the difference in time between two vehicles (that are following each other), when they pass the same fixed point. It is normally used as an indicator for traffic safety [22]. If we assume the two vehicles driving at constant speed, vehicle i follows vehicle $i-1$ (see Fig. 3), time headway is defined as follows:

$$h = \frac{x_{i-1}(t) - x_i(t)}{\dot{x}(t)} = \frac{d(t) + l_{i-1}}{\dot{x}(t)}$$

where h is time headway in seconds, $x_i(t)$ is the position of the vehicle i at time t, $x_{i-1}(t)$ is the position of the vehicle $i-1$ at time t, and $\dot{x}(t)$ is the velocity in (m/s).

TABLE I: Combinations of the experimental runs

Desired gap	**CACC Controller**	
(approx. time headway at 120 km/h)	*Rajamani*	*Ploeg*
15 meters (0.6 seconds)	#1	#5
22.5 meters (0.8 seconds)	#2	#6
30 meters (1 seconds)	#3	#7
42.5 meters (1.4 seconds)	#4	#8

IV. The Experiment

A. Assumptions and Definitions

All vehicles are assumed to be 4-meter long passenger cars. In *plexe-sumo*, the modelled platooning vehicles is given a maximum acceleration capability of 1.5 m/s^2, while the maximum deceleration is limited to 5.88 m/s^2. Furthermore, the manually driven vehicle has the limits at 5 m/s^2 and 9 m/s^2, for acceleration and deceleration respectively. Maximum speed of all vehicles is 41 m/s.

Vehicle-to-vehicle (V2V) communication between the platooning vehicles are assumed to have no packet loss, the default parameters in *plexe-veins* are used as summarized in the Table II. Each platooning vehicles broadcasts *platooning beaconing messages* at 10 Hz rate, containing information about the vehicles' identity, speed, acceleration, and position

TABLE II: Network parameters in *plexe-veins*

Parameter	Value
Path loss model	Free space (α = 2.0)
PHY model	IEEE 802.11p
MAC model	1609.4 single channel (CCH)
Frequency	5.89 GHz
Bitrate	6 Mbit/s (QPSK R = $\frac{1}{2}$)
Access category	AC_VI
MSDU size	200B
Transmit power	20 dBm

(in Cartesian coordinates). In this work, the ego vehicle is assumed to have no V2V communication capabilities.

Besides V2V communication, platooning vehicles are assumed to be equipped with a forward looking radar, that detects an object in front. The radar's field-of-view cover only the vehicle's current lane. The information utilized by the CACC controller are listed in the Table III

TABLE III: List of the information used by the two CACC controllers and their sources.

Control Parameters	Controller	
	Rajamani	*Ploeg*
Platoon leader's speed	V2V	-
Platoon leader's acceleration	V2V	-
Preceding vehicle's speed	V2V	Radar
Preceding vehicle's acceleration	V2V	V2V
Distance to preceding vehicle	Radar	Radar

Moreover, the platooning vehicles always drive in the rightmost lane of the highway, while maintaining a pre-defined inter-vehicular gap and 120 km/h speed according to the control law of the CACC controller.

TTC is defined as the time remaining until collision, if both vehicles maintain its current speed and heading. The calculation of TTC follows the equation described below. If we assume that vehicle i follows vehicle $i-1$ and we use their front bumpers as reference points for their positions, then

$$TTC = \frac{x_{i-1} - x_i - l_{i-1}}{\dot{x}_i - \dot{x}_{i-1}} = \frac{d(t)}{\dot{x}_i - \dot{x}_{i-1}}, \quad \forall \dot{x}_i > \dot{x}_{i-1} \qquad (1)$$

where TTC is time-to-collision, x_{i-1} is the longitudinal position (along the road) of the vehicle $i-1$, l_{i-1} is the length of the vehicle $i-1$, and $\dot{x}_{i-1}$ is the speed of the vehicle $i-1$. Parameters used for this calculation is illustrated in the Fig. 3.

B. Participants and Procedures

The participants in the experiment were recruited through the VTI's database of people who are interested in driving simulator studies. The desired participants are those who:

- Are between 20 and 65 years old.
- Have a driving license for passenger car (category B)
- Have had a driving license for at least 2 years[1]
- Drive at least 500 km per year[2]
- Drive on a highway[3] regularly (at least once a week)

In this study, 39 people participated in the experiment. They are 21 female and 18 male between 19 and 59 years old, with an average of 43 years old. Everyone completed the experiment, although only the data from 37 participants were analysed and presented in this paper. Due to a technical mistake during the experiments, data from two participants are not usable for analysis.

An introduction was given before the participants started to use the driving simulator. The participants were informed that they will encounter a platoon of automated vehicles, on a highway with 120 km/h speed limit. Also, they were asked to merge onto the highway as they would normally do. However, they were not given information about gap settings. After the introduction, each participant was given about 2 minutes of test drive to familiarize with the control of the driving simulator, which simulates a passenger car with automatic gearbox. Finally, each participant drove the scenario 8 times with different settings (see Table I) in a pre-defined order[4]. They were asked to fill in a questionnaire regarding to their opinions about the inter-vehicular gap of the platooning vehicles immediately after each run.

Data from all vehicles in the simulation is collected in the driving simulation software at 200 Hz rate (0.005 seconds time step). Collected information include road, position, and speed, for all vehicles. Additional data are available for the ego vehicle, i.e. steering wheel angle, acceleration, brake activation status, throttle pedal position, and turning indicators' status.

V. Results

This section presents the results from 37 participants, which are 17 men and 20 women with an average age of 42.7 years. The results regarding cut-in behaviours are presented in three parts: *a*) results for all runs; *b*) results when there is a collision; and *c*) results when the participants successfully cut-in without a collision. After that, the results are analysed from perspectives of the human driver and the automated platoon.

A. Overall Behaviours

All cut-in behaviours are presented in Table IV. Although the participants did 8 runs, they experience the same gap setting twice, once for each CACC controller. The different controllers are not noticeable from the participants' perspective. Thus, results from the same gap settings are combined together. Overall, there are many participants that went ahead of the platoon in the shortest gap setting (15 meters). Apart from the shortest gap, the majority of participants cut-in between the platooning vehicles. Regarding the speed, it seems that the bigger the gap, the higher the speed. Moreover, on average, participants tend to leave a bigger space in front of the ego vehicle than behind, except for the biggest gap setting (42.5 meters). At the cut-in, it can be observed from the statistics about cut-in distances, that most of the collisions at the moment of cut-in are rear-end collisions (indicated by a negative value in the cut-in distances).

[1]All but one participant have had their driving license for more than 3 years.

[2]85% of the recruited participants have driven at least 10000 km per year

[3]A road with at least 100 km/h speed limit.

[4]To prevent order effects, balanced Latin Square design is used to generate a sequence for each participant.

TABLE IV: Summary of All Cut-in Behaviours by the Ego Vehicle.

Gap settings in meters (time headway)	Cut-in behaviours (n)			Cut-in speed[1](m/s)			Cut-in distance measured from the ego vehicle					
	In front	Between	Behind	min.	max.	avg.	Distance in front (m)			Distance behind (m)		
							min.	max.	avg.	min.	max.	avg.
15 meters (0.6 seconds)	41	14	19	23.35	33.64	30.20	-2.56	13.20	6.81	-2.20	13.56	3.97
22.5 meters (0.8 seconds)	16	52	6	22.54	34.69	30.91	4.05	19.98	10.32	-1.48	14.45	7.92
30 meters (1 seconds)	0	61	13	23.36	36.70	31.73	3.82	28.47	13.47	-2.97	21.68	12.31
42.5 meters (1.4 seconds)	0	63	11	25.43	37.26	32.19	4.52	39.75	17.98	-1.25	33.48	20.27

[1] Consider only when the participants cut-in between platooning vehicles.
* min. = minimum; max. = maximum; avg. = average

B. Collisions

There are 72 collisions in total from 296 runs, 69 of them occur when the participant cut-in between the platooning vehicles. Thus, the crash rate is approximately 23.31%. Table V summarizes cut-in behaviours when there is a collision. At the 15 meters gap setting, only one participants managed to avoid a collision after cutting in between platooning vehicles. The percentage of collisions decreases as the gap size increases. Negative values in the *cut-in distances behind* indicate that a platooning vehicle collides into the side of the ego vehicle as the ego vehicle cut-in. Similarly, negative values in the *cut-in distances in front* mean that, when the ego vehicle changes lane, it collides into a platooning vehicle at the moment of cut-in. Furthermore, it can be observed from the Table V that, on average at the time of cut-in, the participants reached lower speed and left smaller gap behind the ego vehicle compared to the overall behaviours in the Table IV.

On the other hand, platooning vehicles try to maintain 120 km/h speed and a pre-defined inter-vehicular distance. Also, the platooning vehicle detects the manually driven vehicle with its forward looking radar only when it has already cut-in. Therefore, in some cases it was too late for the platooning vehicle to brake and avoid the collisions. Also, in some cases, the participants did not see the platooning vehicle and collide with it by surprise while cutting in. This is because of a limited field-of-view behind the ego vehicle in the driving simulator, which makes it difficult for the participants to see a vehicle in a blind spot.

In this dataset, TTC is calculated at each time step (every 0.005 seconds), up until the collision (if any). Thus, for each experiment run, there is an array of TTC over time. The Table VI summarizes TTC before the collisions and speed differences during cut-ins. In the cases of collisions, TTCs are calculated only until the collision point.

In Table VI, some of the data points are presented as not available (n/a). There are two reasons when TTC is not available. First, it could be because there are no "unsafe" TTC in those situations. In this paper, 6 seconds is decided as an upper bound for TTC due to the fact that nobody considers TTC larger than 6 seconds to be dangerous [22] (see Section VI for more details). Second, the vehicles collide before they are in "car-following" state, i.e. the ego vehicle collides into the side of a platooning vehicle or vice versa.

One hypothesis is that *the speed differences is the main reason for collisions*. Speed difference at collisions is calculated by subtracting the ego vehicle's speed with the vehicle that it collides with. From the Table VI, speed difference is a negative value on average, which shows that the ego vehicle usually has lower speed when the collisions happen. This fact is confirmed by looking at the histogram of speed differences in Fig. 4.

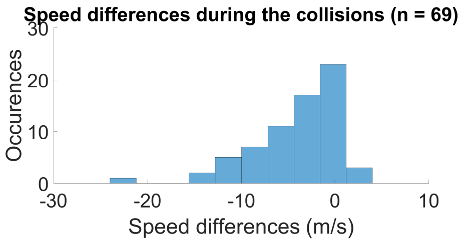


Fig. 4: Histogram of speed differences during the cut-in collisions (n = 69).

To further test this hypothesis, we calculate correlation in our data between collisions and four other factors, which are: *i*) ***gap settings***: desired inter-vehicular gap between platooning vehicles; *ii*) ***cut-in speed***: absolute of the difference between 33.33 m/s (120 km/h) and the speed of the ego vehicle when it cut-in; *iii*) ***gap in front***: gap in front of the ego vehicle at the moment after it has cut-in; and *iv*) ***gap behind***: gap behind the ego vehicle at the moment after it has cut-in.

Thus, according to the Table VII, *cut-in speed*, *gap behind*, and the *gap settings* are highly correlated with collisions. Actually, it is the distance behind the ego vehicle when cut-in (*gap behind*), that is the most correlated factor with collisions. Furthermore, *gap settings* is highly correlated with *gap behind* and *gap in front*, but the *gap in front* is not correlated to the crashes. Finally, the correlation values describe relationship between these factors. Negative value in correlation between the *gap settings* and *crashes* indicates that, when the desired inter-vehicular gaps between platooning vehicles increases, collisions tend to decrease. Likewise, bigger gap behind the ego vehicle when it cut-in caused less collisions. Positive correlation between *cut-in speed* and *crashes* shows that big speed differences during cut-in result in more collisions.

In conclusions, the data shows that collisions are highly correlated with *gap behind*, i.e. how big gap does a platooning vehicle has after the cut-in has occurred. This is followed by *gap settings* and *cut-in speed* respectively.

C. Successful Cut-Ins

Some participants managed to cut-in between the platooning vehicles without a collision, as shown in the summary of

TABLE V: Summary of the Ego Vehicle's Cut-in Behaviours When There is a Cut-in and a Collision.

Gap settings in meters (time headway)	**Collisions** Cut-ins (n)	Collisions (n)	Collisions (percent)	**Cut-in speed[1]**(m/s) min.	max.	avg.	**Cut-in distance measured from the ego vehicle** Distance in front (m) min.	max.	avg.	Distance behind (m) min.	max.	avg.
15 meters (0.6 seconds)	14	13	92.86	23.35	33.64	29.97	-2.56	13.20	6.99	-2.20	13.56	3.78
22.5 meters (0.8 seconds)	52	32	61.54	22.54	34.69	30.20	4.05	19.98	11.61	-1.48	14.45	6.64
30 meters (1 seconds)	61	20	32.79	23.36	36.70	30.20	8.41	28.47	18.56	-2.97	17.09	7.16
42.5 meters (1.4 seconds)	63	4	6.35	25.43	36.24	32.25	32.96	39.75	35.57	-1.25	5.54	2.81

[*] Consider only collisions that occur when the participants cut-in between platooning vehicles.

TABLE VI: Summary of Time-to-collisions and Speed Differences When There is a Cut-in and a Collision.

Gap settings in meters (time headway)	**Time-to-collision (TTC)[1]** Ego vehicle min.	max.	avg.	The vehicle behind ego min.	max.	avg.	**Speed difference at collisions** (m/s) min.	max.	avg.	S.D.
15 meters (0.6 seconds)	n/a	n/a	n/a	0.0020	5.8528	1.1822	-9.98	3.11	-2.88	3.34
22.5 meters (0.8 seconds)	n/a	n/a	n/a	0.0002	5.9918	1.3492	-15.38	1.35	-4.21	4.79
30 meters (1 seconds)	4.2934	5.7381	4.6555	0.0018	5.9991	1.4275	-22.80	1.87	-4.92	5.65
42.5 meters (1.4 seconds)	n/a	n/a	n/a	0.0023	1.1858	0.5171	-7.65	3.07	-2.04	4.42

[1] Consider only TTC less than 6 seconds when the crash did not happen.
[2] Consider only distance when the ego vehicle cut-in between platooning vehicles.
[*] n/a means there were no TTC less than 6 seconds; S.D. = standard deviation.

TABLE VII: Correlations between collisions and other factors.

	Gap settings	Cut-in speed	Gap in front	Gap behind	Crashes
Gap settings	1.0000	-0.1325	**0.5063**	**0.6649**	**-0.5542**
Cut-in speed	-0.1325	1.0000	-0.0496	-0.0994	**0.3810**
Gap in front	**0.5063**	-0.0496	1.0000	**-0.2907**	0.0575
Gap behind	**0.6649**	-0.0994	**-0.2907**	1.0000	**-0.6547**
Crashes	**-0.5542**	**0.3810**	0.0575	**-0.6547**	1.0000

Bold numbers indicate significant correlations at 99% confidence interval.

successful merging runs in the Table VIII. As presented in the table, when there is no collision, the average cut-in speeds are closer to 120 km/h (33.33 m/s) compared to the overall average (see Table IV). Moreover, in majority of successful cut-ins, the participants leave bigger gap behind the ego vehicle. These facts support the conclusions above that speed differences (especially those cut-ins with too low speed) and the distance behind ego vehicle (at the moment of cut-in) are related with collisions. In other words, matching the speed of the platooning vehicles; and leaving more space for them to react, are two important factors for making successful cut-in manoeuvres.

Even though a collision is avoided, there are some dangerous situations indicated by low TTC values (less than 6 seconds), as summarized in Table IX. In these cases, the TTCs are calculated until the end of each experiment runs. As mentioned above in Section IV, the two CACC controllers rely on slightly different information. Moreover, they "weight" each information differently according to their design. Therefore, the TTC values are presented separately by controllers in this case to give some indication about hazardous situations presented to each controllers, and how they managed to handle them. Furthermore, some participants were also exposed to dangerous situations, indicated by low TTC for the ego vehicle. The duration column in the table shows the total duration that the low TTC values happened for the successful cases.

D. Human Drivers' Perspectives

Parts of the questionnaires are analysed, and the results are presented in this section. After each run, the participants were asked to rate the inter-vehicular gap of the platooning vehicle. Note that the participants did not know which gap they are evaluating. The participants rate the gap through the following three questions (on the scale from 1-5):

1) How would you rate the inter-vehicle gap between platooning vehicles in term of safety? (1 = not safe at all; 5 = very safe)
2) Did you feel comfortable while driving between platooning vehicles? (1 = not comfortable; 5 = very comfortable)
3) How easy was it to drive between platooning vehicles? (1 = very hard; 5 = very easy)

The results for the question 1, 2, and 3 are presented in Fig. 5, 6, and 7, respectively.

All the answers (if any) are presented in the figures without any exclusion. Maximum number of answers is 74, thus it can be observed that not everyone answers all the questions. Even when there are answers, some of them may not be valid, because it is difficult to rate the inter-vehicular gap if the participant went ahead or behind the platoon (which is especially the case when the inter-vehicular distance is 15 meters). As shown in Table IV, most participants either go in front or behind the platoon when the gap is 15 meters. Some participants who decided to go ahead of the platoon evaluate the 15 meters gap as safe. Perhaps because it did not cause the conflict, or that they did not have the opportunity to see the gap.

Furthermore, some participants answer the second and third questions, even though the person did not cut-in between the platooning vehicles. We could not have controlled this because the participants answer the questionnaire immediately after each run on their own during the experiments, since it would be difficult to recall the sequence of the gap afterwards. And, although the participants encounter the same gap size 2 times each, they may experience it differently, because we did not control how they encounter the platoon (the conditions for

TABLE VIII: Summary of the Ego Vehicle's Cut-in Behaviours When There is a Cut-in Without a Collision.

Gap settings in meters (time headway)	**Cut-in behaviours (n)**			**Cut-in speed[1] (m/s)**			**Cut-in distance measured from the ego vehicle**					
	In front	Between	Behind	min.	max.	avg.	Distance in front (m)			Distance behind (m)		
							min.	max.	avg.	min.	max.	avg.
15 meters (0.6 seconds)	40	1	19	33.21	33.21	33.21	4.54	4.54	4.54	6.46	6.46	6.46
22.5 meters (0.8 seconds)	15	20	6	27.74	34.19	32.05	4.87	11.81	8.24	6.19	13.13	9.96
30 meters (1 seconds)	0	41	13	28.96	35.80	32.47	3.82	21.27	10.98	4.56	21.68	14.82
42.5 meters (1.4 seconds)	0	59	10	26.82	37.26	32.19	4.52	30.82	16.78	7.18	33.48	21.46

[1] Consider only when the participants cut-in between platooning vehicles.

TABLE IX: Summary of Time-to-collisions When There is a Cut-in Without a Collision.

Gap settings in meters (time headway)	**Time-to-collision (TTC)[1]**											
	Ego vehicle				The vehicle behind ego							
	min.	max.	avg.	dur.	The *Rajamani* controller				The *Ploeg* controller			
					min.	max.	avg.	dur.	min.	max.	avg.	dur.
15 meters (0.6 seconds)	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0.00	n/a	n/a	n/a	0.00
22.5 meters (0.8 seconds)	2.3967	5.9844	3.9367	1.65	3.1300	5.9995	4.2998	3.71	1.7430	5.9898	3.6216	3.27
30 meters (1 seconds)	2.9125	5.9984	4.3530	1.30	2.7552	5.9969	4.2426	4.91	2.6095	5.9976	4.7045	3.35
42.5 meters (1.4 seconds)	2.6962	5.9965	4.5198	1.91	4.0360	5.9976	4.7661	5.33	3.4464	5.9981	5.0025	5.21

[1] Consider only TTC less than 6 seconds when the crash did not happen.
[*] n/a means there were no TTC less than 6 seconds; dur. = total duration in seconds when the TTCs are less than 6 seconds.

starting the platoon is fixed, but the participants may control the ego vehicle differently on each run).

For the first question regarding safety of the gap between platooning vehicles, the participants tend to rate larger gap as safer, except for the smallest gap due to the reasons discussed above. Figure 5 summarizes all the available answers for this question.

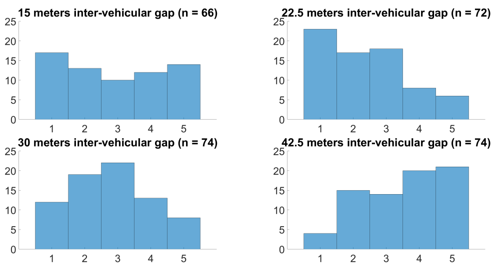


Fig. 5: Histogram of the answers to the first question of the questionnaire (*How would you rate the inter-vehicle gap between platooning vehicles in term of safety? (1 = not safe at all; 5 = very safe)*).

Opinions regarding whether the participants feel comfortable driving between platooning vehicles are summarized in Fig. 6. The participants clearly feels more comfortable in the largest gap setting for the platooning vehicles (42.5 meters). However, it is not clear for other gap settings, except the 22.5 meters gap, which shows that the participants tend to feel uncomfortable.

Lastly, Fig. 7 shows the answers when the participants were asked about difficulties on driving between the platooning vehicles. Most of the participants find it very easy to drive between the platooning vehicles at the largest gap setting (42.5 meters). They tend to feel that it is very difficult to drive at the 22.5 meters gap setting, but the conclusions cannot be drawn from the other two gaps.

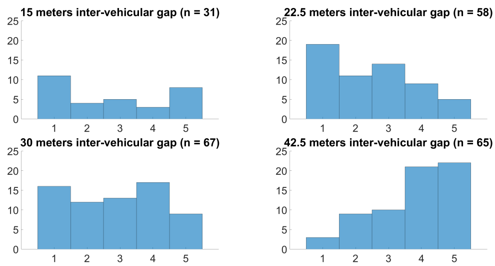


Fig. 6: Histogram of the answers to the second question of the questionnaire (*Do you feel comfortable while driving between platooning vehicles? (1 = not comfortable; 5 = very comfortable)*).

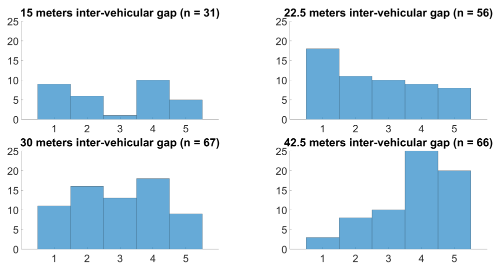


Fig. 7: Histogram of the answers to the third question of the questionnaire (*How easy was it to drive between platooning vehicles? (1 = very hard; 5 = very easy)*).

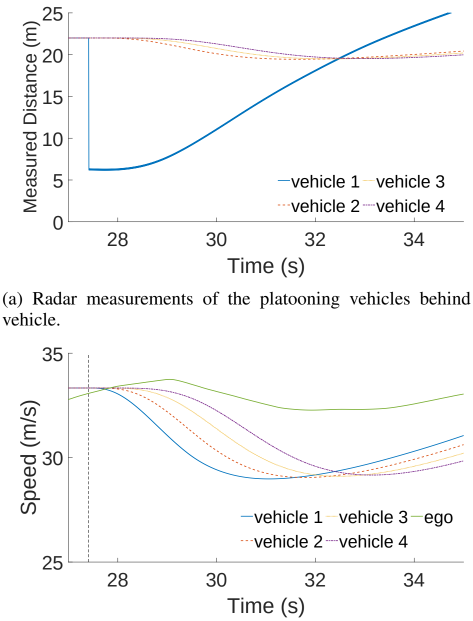
E. Platooning Vehicles' Perspectives

From the CACC controller's perspectives, the ego vehicle can be seen as a disturbance when it cut-in or cut-out, although only a few participants cut-out (change to the left lane after merging into the platoon). As mentioned above that the platooning vehicles only have a forward looking radar, that detects an object in front in their own lane. Therefore, the ego vehicle is only detected when it has already started to cut-in between platooning vehicles, which is often too late to react. Among all 69 collisions when cut-in, 62 of them (89.85%) have at least a positive gap behind the ego vehicle when the cut-in occurs. Therefore, most collisions are rear-end collisions, i.e. a platooning vehicle collided into the ego vehicle.

As discussed above, distances in front of a platooning vehicle after the ego vehicle cut-in is highly correlated with collisions. Moreover, even though the collision did not happen, it can cause dangerous situations, which affect the platoon's safety, stability, and efficiency. However, short distance alone is not the only factor, as the correlation table shows that the speed difference is also an important factor.

Nevertheless, the platoon is disturbed even when the collision did not happen. To illustrate this from the platoon's perspective, Fig. 8 shows a situation, that has short distance behind the ego vehicle but did not lead to a collision. The figure shows radar measurements of the platooning vehicles behind the ego vehicle, and the speed of the ego vehicle and the platooning vehicles behind it. After the cut-in the distance between vehicle 1, which is platooning vehicle behind the ego vehicle, to the ego vehicle is about 6 meters. The speed profile of all vehicles behind the cut-in vehicle shows that, the cut-in manoeuvre caused a disturbance for the platooning vehicles behind the ego vehicle. Even though the cut-in speed of the ego vehicle was 33.08 m/s, which is very close to the platooning vehicle's speed (33.33 m/s), disturbances can be observed on all platooning vehicles behind the ego vehicle.

No analysis after collisions can be done with this data, because a vehicle in *plexe-sumo* that is involved in a collision will be relocated when the collision occurs. Although the simulation did not stop after the collision, it is difficult to keep track of the exact position of all vehicles, as we decided to collect the data only from the driving simulator. The driving simulation software only receives the initial position of the platooning vehicles to display in the simulation. After that, only speeds of the platooning vehicles are received from *plexe-sumo*.



(a) Radar measurements of the platooning vehicles behind the ego vehicle.

(b) Speed of the ego vehicle and the platooning vehicles behind it. The cut-in time is indicated by black dashed line.

Fig. 8: A Disturbance caused by a cut-in manoeuvre.

VI. Safety Evaluation

This section compares the two CACC controllers used in this experiment from safety perspectives using safety indicators based on TTC. TTC are presented in the previous section for the ego vehicle and the platooning vehicle behind it (after the cut-in occurred). Additionally, TTC of every platooning vehicles in the scenario will be considered in this section. Note that, if there is a collision, the TTCs are calculated only until the time of the collision. Upper bound for TTC is 6 seconds, because the TTCs over 6 seconds are commonly regard as safe, according to [22] as aforementioned. Moreover, another study [32] shows that the speed of the follower is independent to that of the leader when the headway is more than 6 seconds.

Regarding the lower bound of TTCs, many have suggested different critical lower bounds, which depend on the situation under evaluation. For instance, [33] regarded 1.8s as a suitable lower limit for their study, although the observed average TTC_{min} found were 3.5s with an assistance system, and 2.4s without the system. Another study [23] mentioned that the 3 seconds threshold is considered adequate for making decision whether the situations are acceptable or dangerous.

Therefore, this section will investigate the TTC ranging from 1 to 3 seconds as previously done in [23], [24].

For the safety evaluation, we follow the measures presented in [23], namely Time Exposed Time-to-collision (TET) and Time Integrated Time-to-collision (TIT). If we denote the threshold value as TTC^*, the total simulation time (or until the collision) as T, and the time step (i.e. 0.005s) as Δt, then TET and TIT of the vehicle i can be defined as follows.

$$TET_i = \sum_{t=1}^{T} \delta_i(t) \cdot \Delta t,$$

$$\text{where} \quad \delta_i(t) = \begin{cases} 1 & \forall 0 < TTC_i(t) \leq TTC^* \\ 0 & otherwise \end{cases}$$

$$TIT_i = \sum_{t=1}^{T} (TTC^* - TTC_i(t)) \cdot \Delta t, \quad \forall 0 < TTC_i(t) \leq TTC^*$$

For each participant and experimental run, we use the definition above to calculate TET and TIT for all platooning vehicles, until the collision point or the end of simulation (whichever comes first). Then we aggregate the indicators by the experimental run numbers (see Table I). The results are presented in the Table X. The total time for all the experimental runs is about 3 hours 50 minutes (≈13753.17 seconds), please note that this time include the time after the collisions.

TABLE X: TET and TIT Indicators for all Platooning Vehicles Separate by the Experimental Run Numbers (C indicates number of collision(s))

Run	TTC^* = 1s		TTC^* = 2s		TTC^* = 3s		C
	TET	TIT	TET	TIT	TET	TIT	
#1	2.77	1.6148	3.12	4.6579	3.34	7.7684	7
#2	21.49	7.4711	27.53	24.7834	31.40	46.6002	17
#3	4.93	3.1125	5.67	8.4581	6.80	14.2763	8
#4	2.13	1.0483	2.34	3.3607	2.34	5.6907	4
sum	31.32	13.2466	38.66	39.2601	43.88	74.3356	36
#5	5.21	2.6136	9.37	10.4356	10.18	20.3642	7
#6	8.55	5.3061	9.79	14.1978	11.25	24.8300	16
#7	10.30	4.6207	16.43	18.0912	19.45	35.9853	12
#8	0.25	0.2128	0.25	0.4578	0.25	0.7028	1
sum	24.31	12.7532	35.84	43.1824	41.13	80.8823	36

According to the Table X, the upper part (separated by the dashed line) is the runs that used the *Rajamani* controller, and the lower part used the *Ploeg* controller. It can be observed that collisions occurred equally to the two controllers, 36 times each. Thus, we sum the TET and TIT for each controller. The results show that the *Rajamani* controller are more exposed to the dangerous situations that the *Ploeg* controller, suggested by TET. According to TIT, the results show that while the *Ploeg* controller is less exposed to the danger, differences of TTCs from the critical threshold are bigger during those time. However, the authors of [23], who proposed these indicators, conclude that TET is *"especially useful in a comparative analysis of scenarios"* and easier to interpret the meaning compared to TIT, thus TET is preferred in comparative studies. Hence, if we evaluate according to the TET, the *Ploeg* controller is better than the *Rajamani* in terms of road traffic safety. Furthermore, the duration column (dur.) in the Table IX is actually TET values for the TTCs less than 6 seconds, when there is a cut-in without a collision. The TET in that table also supports our conclusion here regarding the controller.

Finally, if we consider the average TTCs observed in the Table IX and Table VI. The two tables suggest that, the average TTCs are about 1.1 seconds and 4.4 seconds, when there is a collision, and no collision respectively. We can also conclude that critical TTC value of 2 to 3 seconds would also be suitable for safety evaluation of cut-in situations, although the minimum TTC when there is no collision is 1.7430 seconds.

VII. Discussions

From this study, we acknowledge some limitations in the simulation framework and the scenario set up. Thus they will be discussed in this section, together with the results.

A. The Cut-In Scenario

According to the guideline from the Swedish Transport Administration (Trafikverket) [34], recommended length for the on-ramp section is at least 250 meters (for a rural road with speed limit at least 110 km/h). However, the road section for the experiments was selected from an existing road database, and this recommendation was not taken into account when selecting the section. Hence, the length that was used is only 190 meters (see Fig. 2). Since the distance is shorter than usual, it might be more demanding for the participants to perform the "merging onto highway" task in such a limited space. Nevertheless, none of the participants mentioned the difficulty caused by this shorter length.

Even though the driving simulator has over 180 degree field-of-view in front, it does not display the view behind the vehicle, thus excludes blind spots from side- and rear-view mirrors. This seems to be the biggest problem for the participants. The majority of them said that, they usually look behind to check the blind spot, in addition to using the side mirrors. However, this is not available in the driving simulator, because the passenger car cabin only has front half of the car and there is no projector behind. This makes lane changing/merging tasks more difficult for most participants, as many of them did not see the vehicle in the blind spot, then collide with it while changing lane. Therefore, to have a view behind the ego vehicle is an important thing to consider when creating realistic cut-in scenarios.

B. Data and Results

We have a few hypotheses about the reasons for frequent cut-in ahead of the platoon at 15 meters gap settings. First, it may not be difficult to go in front of the whole platoon in the 15 meters settings, this is indicated by observing the cut-in speed for these cases, which are not as high compared to the other gap settings (see Table IV). The other reason could be because 15 meters is perceived as a too short distance to cut-in, so the participants decided to either accelerate and go ahead of the whole platoon, or wait and go behind it. The last reason could be the scenario design. As mentioned above, the platooning vehicles only appear on the road after the ego vehicle has reached a certain speed and position on the road. Starting position offset might be too far behind for the 15 meters case. The offsets in starting point is needed, due to differences in gap settings cause the different lengths of the platoon. Also, when a platoon is short, there is a lower chance that the participants will encounter the platoon as we wish.

Looking further on to the questionnaire data during the 15 meters gap runs. If we only consider participants who had a collision or went behind the platoon, the safety rating for this group is lower compared to the overall results (see Fig. 9). Thus, we can draw the conclusion that if the participants have seen the gap, the 15 meters gap is usually considered as not safe.

Moreover, the fact that the participants did not know the size of the gap they are evaluating could contribute to some unclear answers in the questionnaires. We suspect that it is difficult for the participants to detect small differences in distance, for instance the difference between 15 and 22.5 meters might be hard to notice while driving at high speed. Therefore, the evaluation might be highly affected by the sequence of the gap presented to the participants.

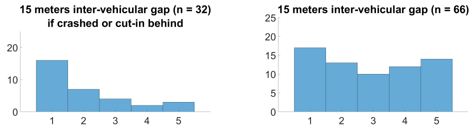


Fig. 9: Histogram of the answers to the first question of the questionnaire (How would you rate the inter-vehicle gap between platooning vehicles in term of safety? (1 = not safe at all; 5 = very safe)). This shows a comparison between the group that crashed or went behind, and the overall answers.

C. Avoiding Collisions

This subsection investigates how difficult it would be to avoid the collisions that was observed in this study. One of the alternatives to avoid the collision is to equip an additional long-range radar to the platooning vehicles, looking at their right side (45-60 degree with respect to their heading) in order to detect the manually driven vehicle. Figure 10 illustrates the radar, collision point, and detection point in our assumptions. We calculate the situation backwards from all collisions that occurred in the experiments, with the following assumptions below:

1) The radar can detect up to 150 meters.
2) The platooning vehicle detects the ego vehicle approximately 4.5 seconds before it will reach the original collision point (if the speed of 120 km/h is maintained, the collision will happen).
3) Maximum deceleration for avoiding collisions is -2.5 m/s^2.
4) A first order low-pass filter, which imitates actuation lag, as implemented in Plexe is used (see Fig. 11).
5) The speed difference at the collision is known.
6) The ego vehicle maintains constant speed from the detection point until the collision point.
7) If the platooning vehicle did not react, its speed at the collision is 120 km/h.
8) Time step is 0.01 seconds in the calculation (Δt).

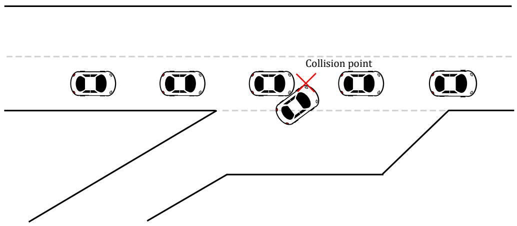
Start from the speed difference at collisions, we calculate how long time it will take to match the ego vehicle's speed, if constant deceleration of -2.5 m/s^2 is applied. This process is explained in the equation below, where t is the time needed to reduce speed and v_{diff} is the speed difference at the collision.

$$t = \frac{v_{diff}}{-2.5}$$

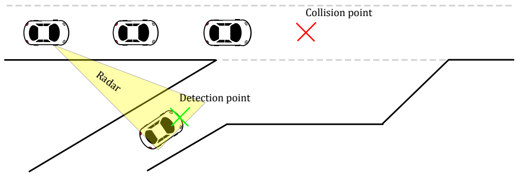
According to the assumptions, the platooning vehicles can detect the manually driven vehicle 4.5 seconds before collisions. Therefore, if t is less than 4.5 seconds, it will brake only if necessary, then keep a constant speed. The equation below, where a is acceleration, describe this process:

$$a = \begin{cases} -2.5 & \text{until time} \quad t \\ 0 & \text{from} \quad t \quad \text{until 4.5 seconds} \end{cases}$$

Otherwise, if t is more than 4.5 seconds, it applies -2.5 m/s^2 for 4.5 seconds. Speed and distance travelled are calculated at every time step. As mentioned in the assumption 3, even though the constant deceleration is assumed, the actual acceleration values applied in the calculation consider actuation lag as depicted in Fig. 11.



(a) Collision point where the collision was observed in the experiments.



(b) Detection point where the ego vehicle is assumed to be detected. This point is assumed to be 4.5 seconds before the original collision happened.

Fig. 10: Illustration of the assumptions about additional sensor, collision point, and detection point.

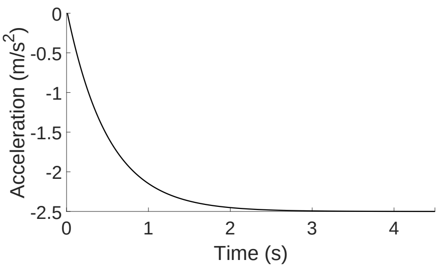


Fig. 11: Acceleration profile while braking

TABLE XI: Expected Parameters of the Platooning Vehicles if the Collision is Avoided by Braking.

	min.	max.	avg.	S.D.
New speed	23.36	31.92	28.26	3.05
New distance behind ego	4.29	34.67	18.13	8.27

Therefore, we conclude that, if we can detect the manually driven vehicle before it cuts-in, the collisions can be avoided by applying -2.5 m/s^2 braking to the platooning vehicles. Nevertheless, this requires at least an extra sensor in order to detect the manually driven vehicle, since we assume that the manually driven vehicle has no wireless communication capability.

Furthermore, it is also possible that the cut-in manoeuvre triggers *advanced emergency braking systems* (AEBS), which can be dangerous for vehicles behind. According to the European regulation [35], AEBS have at least 4 m/s^2 deceleration, and it should not start before a TTC equal to or less than 3 seconds. However, we observed several cases that has TTC less than 3 seconds as shown in the Section VI, hence AEBS could be activated. Consequently, the emergency braking manoeuvre can be dangerous for other road users behind the platoon. Although it may help avoiding collisions for platooning vehicles, the emergency braking causes even more disturbance to the traffic flow.

VIII. Future Work

There are answers in free text, transcript from the short discussion with participants, and another questionnaire, that were excluded from the analysis of this paper, because they are more related to the social acceptance issues, which is not the main goal of this paper. Nevertheless, the questionnaire data can compliment the analysis presented in this paper, as a future research direction in human factors area. For example, most participants mentioned that they expect the platooning vehicles to react as people would normally do, such as changing to the left lane or slow down to make way for merging vehicle(s). Analysing this further could answer questions regarding social acceptance. Moreover, there is more to learn about the participants' cut-in behaviour, since the data regarding throttle level, brake activation, steering wheel angle, and usage of turning indicators are recorded for the ego vehicle.

Besides acceptance issues, a more thorough analysis on the effects of cut-in on string stability and fuel efficiency can be done using the data collected in this work. However, acceleration of the platooning vehicles are not recorded in the dataset. Only data regarding position, road, and speed are available from the platooning vehicles.

In order to avoid collisions, besides equipping additional sensor as assumed in the Section VII-C, there are other alternatives for detecting the manually driven vehicle and avoiding collisions. For example, using a road side unit to monitor the merging lane, and send this information to the platoon via infrastructure-to-vehicle communication. Alternatively, the manually driven vehicle can be equipped with a simple communication device that broadcast GPS coordinates and speed to the V2V communication network, as also suggested in [6] to make CACC vehicles aware of the manually driven vehicle. Another solution is that the platoon uses the GPS coordinates and road map data to automatically reduce the speed and increase the gap at each merging point on a highway, to facilitate vehicles that need to merge onto the highway. These alternatives can be simulated, and then compared with the results presented in this paper to see whether these possible solutions would improve the situations.

Furthermore, reaction time of the driver in each CACC vehicles could also contributes to avoiding collisions. The simulation framework used in this work can also be used to include human participants as a driver of a CACC vehicle. Learning about the reaction time of CACC drivers, similar to the work done on ACC by *Annika F.L. Larsson, et al.* [36], could be another valuable future work.

The coordinate system for positions of the vehicles in the driving simulator follows the track coordinate system defined in the OpenDRIVE[5] standard format, which describe the position in (s,t,h) coordinates. The s defines longitudinal position along the reference line of the road, t describes lateral position (positive to the left), and h indicates the elevation. In the logged data, s and t of all vehicles are recorded at each time step. Therefore, we can obtain cut-in trajectories of the ego vehicle from plotting s and t, as illustrated in Fig. 12. This data can be used to create a model of cut-in behaviours, which can become a car-following model in the traffic simulator for instance. Thus, enabling future simulation studies of highway cut-in scenarios without using the driving simulator, which will reduce the cost (to get access to a driving simulator) and time (for recruiting participants). Also, having a car-following model that includes the cut-in behaviour is also listed as next research steps in [16].

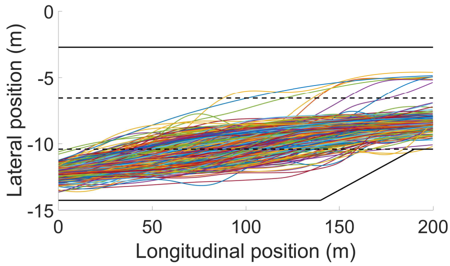


Fig. 12: All cut-in trajectories during this study.

Lastly, the size of platoon in this study is fixed at five vehicles, which corresponds to 80 meters long at the shortest gap setting. Future experiments with different platoon sizes, to find optimal platoon length might be required. Because grouping vehicles in a long platoon could improve traffic flow on the main highway, but that could prevent other vehicles from merging, thus creating a bottle neck at the on-ramp instead. This issue is also as reported in [7] that *"the system has a negative effect on traffic safety in the merging process. Close-CACC platoons prevent other vehicles from cutting in"*.

IX. Conclusions

This paper presents a study of a highway cut-in scenario, where a manually driven vehicle encounters a platoon of five connected and automated vehicles at the merging point of the highway. The study uses an existing C-ITS simulation framework, which contains driving, network, and traffic simulators. A driving simulator is used to control the manually driven vehicle, driven by 37 participants recruited through VTI's driver database. For the vehicles in the platoon, two existing CACC controllers are investigate with four different desired

[5]http://www.opendrive.org/

inter-vehicular gap settings: *i*) 15 meters; *ii*) 22.5 meters; *iii*) 30 meters; and *iv*) 42.5 meters.

The observed behaviours show that the participants cut-in between the platoon less often, when the inter-vehicular gap in the platooning vehicles is small. This frequency of cut-in increases as the gap size increases. When cut-in, the overall collision rate is about 23.31% of all the experiment runs. If we consider collision rate when cut-in for each gap, the smallest gap (15 meters) has the highest rate at 92.86%, and the largest gap (42.5 meters) has the lowest at 6.35%. Setting the platoon's speed at 120 km/h (speed limit of the road) might be too high, considering the fact that the participants perform cut-ins at 108-115 km/h, on average.

As mentioned in Section V, cut-in collisions are highly correlated with difference in cut-in speed and distance between vehicles. When there is no collision, the average cut-in speeds are about the same speed as platooning vehicles, and there is a bigger gap behind the ego vehicle after cut-in compared to the other cases. Therefore, *two important factors for making successful cut-in manoeuvres are to match the speed of the platooning vehicles; and leaving more space for them to react*. In addition, the collisions observed in this study can be avoided by applying brake at -2.5 m/s^2 to the platooning vehicles, assuming that the vehicle coming from the on-ramp can be detected, as shown in the Section VII-C.

Furthermore, TTC seems to be a good safety indicator for this situation, because it considers both distance between vehicles and speed difference. When there is no collision, both CACC controllers handle the situation well, according to the total duration of the low TTCs and the average TTC presented in Table IX. The results from the questionnaire shows that there is a clear difference between the answers for 22.5 meters and 42.5 meters gap, where the participant feel safer and more comfortable in the latter. However, obvious conclusions cannot be drawn from their opinions on the other two gaps.

Two CACC controllers were involved in exactly same amount of collisions as summarized in Table X, thus we cannot conclude which one is clearly safer then the other. Nevertheless, a comparison of their behaviour with respect to safety presented in Section VI shows that, the *Rajamani* controller is exposed to dangerous situations longer, suggested by the higher TET values. However, higher TIT values for the *Ploeg* controller suggest that, TTC values during the exposed time are lower, which indicate that the situations are more severe.

Moreover, this paper suggests in Section VII, how the set up in this study can be improved for future research. Also, the importance of involving human driver in testing and evaluation is highlighted in this experiment.

Most importantly, this paper shows that *it is necessary for CACC controllers to have a strategy for handling cut-in situations at merging points of highways, with extra sensors and/or procedures such as equipping radar that looks to the side, increasing inter-vehicular gap at merging points, etc.* Otherwise, collisions and dangerous situations as presented in this paper could occur. Some strategies have been suggested in the literature, which are mostly theoretical. One important future research direction would be to investigate how the solutions will be implemented, and how much do they improve the safety of cut-in situations.

Acknowledgement

This study is partly funded by the Swedish government agency for innovation systems (VINNOVA) through the NGEA step 2 project (ref 2015-04881), and Vehicle and Traffic Safert Centre at Chalmers (SAFER). Furthermore, the authors would like to acknowledge Bruno Augusto, Gunilla Sörensen, Niklas Strand, Alexander Eriksson, Mattias Söderholm Mullaart, and Henrik Nilsson for their support.

References

[1] D. N. Godbole, F. H. Eskafi, and P. P. Varaiya, "Automated highway systems," *IFAC Proceedings Volumes*, vol. 29, no. 1, pp. 5506 – 5511, 1996, 13th World Congress of IFAC, 1996, San Francisco USA, 30 June - 5 July. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1474667017585584

[2] J. K. Hedrick, D. McMahon, V. Narendran, and D. Swaroop, "Longitudinal vehicle controller design for ivhs systems," in *1991 American Control Conference*, June 1991, pp. 3107–3112.

[3] H.-S. Tan, R. Rajamani, and W.-B. Zhang, "Demonstration of an automated highway platoon system," in *Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No.98CH36207)*, vol. 3, Jun 1998, pp. 1823–1827 vol.3.

[4] S. E. Shladover, C. Nowakowski, X.-Y. Lu, and R. Ferlis, "Cooperative Adaptive Cruise Control: Definitions and Operating Concepts," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2489, pp. 145–152, 2015.

[5] C. Bergenhem, H. Pettersson, E. Coelingh, C. Englund, S. Shladover, S. Tsugawa, and M. Adolfsson, "Overview of platooning systems," in *Proceedings of the 19th ITS World Congress :*, 2012, pp. 1–7. [Online]. Available: http://publications.lib.chalmers.se/publication/174621

[6] S. Shladover, D. Su, and X.-Y. Lu, "Impacts of Cooperative Adaptive Cruise Control on Freeway Traffic Flow," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2324, pp. 63–70, 2012. [Online]. Available: https://doi.org/10.3141/2324-08

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

[8] W. J. Schakel, B. van Arem, and B. D. Netten, "Effects of Cooperative Adaptive Cruise Control on traffic flow stability," in *13th International IEEE Conference on Intelligent Transportation Systems*, Sept 2010, pp. 759–764.

[9] M. P. Lammert, A. Duran, J. Diez, K. Burton, and A. Nicholson, "Effect of Platooning on Fuel Consumption of Class 8 Vehicles Over a Range of Speeds, Following Distances, and Mass," *SAE Int. J. Commer. Veh.*, vol. 7, pp. 626–639, 09 2014. [Online]. Available: https://doi.org/10.4271/2014-01-2438

[10] V. Turri, B. Besselink, and K. H. Johansson, "Cooperative Look-Ahead Control for Fuel-Efficient and Safe Heavy-Duty Vehicle Platooning," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 1, pp. 12–28, Jan 2017.

[11] A. A. Alam, A. Gattami, and K. H. Johansson, "An experimental study on the fuel reduction potential of heavy duty vehicle platooning," in *13th International IEEE Conference on Intelligent Transportation Systems*, Sept 2010, pp. 306–311.

[12] A. Alam, "Fuel-Efficient Distributed Control for Heavy Duty Vehicle Platooning," pp. ix, 97, 2011, qC 20111012.

[13] J. Axelsson, "Safety in Vehicle Platooning: A Systematic Literature Review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1033–1045, May 2017.

[14] M. Aramrattana, T. Larsson, J. Jansson, and A. Nåbo, "A simulation framework for cooperative intelligent transport systems testing and evaluation," *Transportation Research Part F: Traffic Psychology and Behaviour*, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1369847816306635

[15] 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*, vol. 15, no. 1, pp. 296–305, Feb 2014.

[16] V. Milanés and S. E. Shladover, "Handling Cut-In Vehicles in Strings of Cooperative Adaptive Cruise Control Vehicles," *Journal of Intelligent Transportation Systems*, vol. 20, no. 2, pp. 178–191, 2016.

[17] H. Liu, X. D. Kan, S. E. Shladover, X.-Y. Lu, and R. E. Ferlis, "Impact of cooperative adaptive cruise control on multilane freeway merge capacity," *Journal of Intelligent Transportation Systems*, vol. 22, no. 3, pp. 263–275, 2018. [Online]. Available: https://doi.org/10.1080/15472450.2018.1438275

[18] M. Aramrattana, T. Larsson, C. Englund, J. Jansson, and A. Nabo, "Simulation of cut-in by manually driven vehicles in platooning scenarios," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Oct 2017, pp. 1–6.

[19] M. Aramrattana, C. Englund, J. Jansson, T. Larsson, and A. Nåbo, "Safety analysis of cooperative adaptive cruise control in vehicle cut-in situations," in *Proceedings of 2017 4th International Symposium on Future Active Safety Technology towards Zero-Traffic-Accidents (FAST-zero)*. Society of Automotive Engineers of Japan, 2017.

[20] Q. Xu and R. Sengupta, "Simulation, analysis, and comparison of ACC and CACC in highway merging control," in *IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No.03TH8683)*, June 2003, pp. 237–242.

[21] J. C. Hayward, "Near miss determination through use of a scale of danger (traffic records 384)," *Highway Research Board, Washington, DC*, 1972.

[22] K. Vogel, "A comparison of headway and time to collision as safety indicators," *Accident Analysis & Prevention*, vol. 35, no. 3, pp. 427 – 433, 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0001457502000222

[23] M. M. Minderhoud and P. H. Bovy, "Extended time-to-collision measures for road traffic safety assessment," *Accident Analysis & Prevention*, vol. 33, no. 1, pp. 89 – 97, 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0001457500000191

[24] M. S. Rahman and M. Abdel-Aty, "Longitudinal safety evaluation of connected vehicles' platooning on expressways," *Accident Analysis & Prevention*, vol. 117, pp. 381 – 391, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0001457517304505

[25] J. Jansson, J. Sandin, B. Augusto, M. Fischer, B. Blissing, and L. Källgren, "Design and performance of the VTI Sim IV," in *Driving Simulation Conference*, 2014.

[26] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, January 2011.

[27] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO - Simulation of Urban MObility," *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, pp. 128–138, December 2012.

[28] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler, and R. L. Cigno, "Plexe: A platooning extension for veins," in *2014 IEEE Vehicular Networking Conference (VNC)*, Dec 2014, pp. 53–60.

[29] A. Wegener, M. Piórkowski, M. Raya, H. Hellbrück, S. Fischer, and J.-P. Hubaux, "Traci: An interface for coupling road traffic and network simulators," in *Proceedings of the 11th Communications and Networking Simulation Symposium*, ser. CNS '08. New York, NY, USA: ACM, 2008, pp. 155–163. [Online]. Available: http://doi.acm.org/10.1145/1400713.1400740

[30] R. Rajamani, *Vehicle dynamics and control*, 2nd ed. Springer US, 2012.

[31] 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 *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Oct 2011, pp. 260–265.

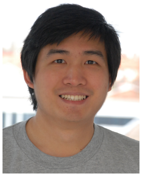
[32] K. Vogel, "What characterizes a "free vehicle" in an urban area?" *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 5, no. 1, pp. 15 – 29, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1369847802000037

[33] J. Hogema and W. Janssen, "Effects of intelligent cruise control on driving behaviour: a simulator study," *TNO-report TM-96-C012*, 1996.

[34] Trafikverket, *Krav för vägars och gators utformning*, 2015, ISBN: 978-91-7467-744-7, In Swedish.

[35] 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," 2012.

[36] A. F. Larsson, K. Kircher, and J. A. Hultgren, "Learning from experience: Familiarity with ACC and responding to a cut-in situation in automated driving," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 27, pp. 229 – 237, 2014, vehicle Automation and Driver Behaviour. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1369847814000746



Maytheewat Aramrattana is a PhD student at Halmstad University, Sweden. He works with the Swedish National Road and Transport Research Institute (VTI) as part of his PhD project, VICTIg. His BEng in Electrical Engineering is from Kasetsart University in Thailand (2010), and his MSc in Computer Science and Engineering (Embedded and Intelligent Systems programme) is from Halmstad University. His research interests are in simulation, modelling, testing, and evaluation of cooperative intelligent transport systems.

Dr. Cristofer Englund received his PhD in electrical engineering from Chalmers University of Technology, Sweden in 2007. Currently he holds a research managerial position within cooperative systems at RISE Viktoria, Gothenburg, Sweden. He is also the research director of the Traffic Systems competence area at SAFER, Vehicle and Traffic Safety Centre at Chalmers University of Technology. In addition, he holds an adjunct senior lecturer position at Halmstad University within information technology. His research interests include automated driving, machine learning, and data mining.

Tony Larsson is a Professor at Halmstad University, Sweden. After a Master in Mechanical Engineering 1974 from University of Linköping, Sweden, he worked at Ericsson AB for 28 years in the areas of design and testing of distributed real-time systems. During this time he was appointed technical expert in system design methods and got a PhD in Computer Science at Linköping University, Sweden 1989. After this he worked shortly for FMV with system architecture for network based defence and in 2003 he became Professor of Embedded Systems at Halmstad University. His current interests include software architecture supporting cooperative embedded systems aiding intelligent transport, vehicle and mobile sensor network applications relying on wireless communication.

Dr. Jonas Jansson is head of the department *Traffic and Road User*. The department focus on research in the human factors, traffic safety, vehicle technology and driving simulation areas. Dr. JanssonâĂŹs background is in driver support systems and vehicle dynamics. He was previously with Volvo Car Corporation between 1999 and 2007, where he worked as function owner for Volvos automatic emergency braking function. His thesis, in the area of automatic control, focus on pre-crash braking systems. Current areas of interest are methodology for advanced driving simulation, technology for automated driving and electric road systems.

Arne Nåbo is a Research Director at VTI (Swedish National Road and Transport Research Institute), for the research unit Driving Simulation and Visualization. He specializes in Driver support systems, driving simulator applications and futurological studies. After finishing his Licentiate of Technology he worked at Saab Automobile AB between 1992 and 2011 with human factors and active safety in product development and advanced engineering. He has expert knowledge within HMI for Active Safety, driver behavior and experimental design of simulator studies, and his current interest includes vehicle automation and road transport electrification.