Designing driver-vehicle cooperation principles for automated driving systems

Chunshi Guo

Thèse de doctorat

Pour obtenir le grade de Docteur de l'Université de

VALENCIENNES ET DU HAINAUT-CAMBRESIS

Spécialité : Automatique

Présentée et soutenue par GUO, Chunshi

Le 29/05/2017, à Valenciennes

Ecole doctorale :
Sciences Pour l'Ingénieur (SPI)

Equipe de recherche, Laboratoire :
Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielles et Humaines (LAMIH-UMR CNRS 8201)

Conception des principes de coopération conducteur-véhicule pour les systèmes de conduite automatisée

JURY

Président du jury
- Basset, Michel. Professeur. Université de Haute Alsace, MIPS.

Rapporteurs
- de La Fortelle, Arnaud. Professeur. MINES ParisTech, Centre de Robotique.
- Basset, Michel. Professeur. Université de Haute Alsace, MIPS.

Examinateurs
- Kemeny, Andras. Professeur associé. Renault, Arts et Métiers ParisTech.
- Nouvelière, Lydie. Maître de Conférences. Université d'Evry, IBISC.

Directeur de thèse

- Popieul, Jean-Christophe. Professeur. Université de Valenciennes, LAMIH.

Co-directeur de thèse : Sentouh, Chouki. Maître de Conférence. Université de Valenciennes, LAMIH.
Co-directeur de thèse : Haué, Jean-Baptiste. Docteur, Ingénieur ergonome. Renault.

Membres invités
- Boer, Erwin R. Professeur associé. Delft University of Technology.
- Langlois, Sabine. Ingénieur ergonome. Renault.

DESIGNING DRIVER-VEHICLE COOPERATION PRINCIPLES FOR AUTOMATED DRIVING SYSTEMS

Chunshi GUO

Laboratory of Industrial and Human Automation, Mechanics and Computer Science (LAMIH – UMR CNRS 8201)

University of Valenciennes and Hainaut-Cambresis

June 2017

To Shiyao

不积跬步，无以至千里；不积小流，无以成江海

Without accumulating tiny steps, you have no way to go a thousand miles;
Without accumulating little streams, you have no way to form river or sea.

—*Xunzi* (Chinese philosopher, c. 310-c. 235 BC)

ABSTRACT

Given rapid advancement of automated driving (AD) technologies in recent years, major car makers promise the commercialization of AD vehicles within one decade from now. However, how the automation should interact with human drivers remains an open question. The objective of this thesis is to design, develop and evaluate interaction principles for AD systems that can cooperate with a human driver. Considering the complexity of such a human-machine system, this thesis begins with proposing two general cooperation principles and a hierarchical cooperative control architecture to lay a common basis for interaction and system design in the defined use cases.

Since the proposed principles address a dynamic driving environment involving manually driven vehicles, the AD vehicle needs to understand it and to share its situational awareness with the driver for efficient cooperation. This thesis first proposes a representation formalism of the driving scene in the Frenet frame to facilitate the creation of the spatial awareness of the AD system. An adaptive vehicle longitudinal trajectory prediction method is also presented. Based on maneuver detection and jerk estimation, this method yields better prediction accuracy than the method based on constant acceleration assumption.

As case studies, this thesis implements two cooperation principles for two use cases respectively. In the first use case of highway merging management, this thesis proposes a cooperative longitudinal control framework featuring an ad-hoc maneuver planning function and a model predictive control (MPC) based trajectory generation for transient maneuvers. This framework can automatically handle a merging vehicle, and at the mean time it offers the driver a possibility to change the intention of the system. In another use case concerning highway lane positioning and lane changing, a shared steering control problem is formulated in MPC framework. By adapting the weight on the stage cost and implementing dynamic constraints online, the MPC ensures seamless control transfer between the system and the driver while conveying potential hazards through haptic feedback. Both of the designed systems are evaluated through user tests on driving simulator. Finally, human factors issue and user's perception on these new interaction paradigms are discussed.

Key words: human-machine interaction, human-machine cooperation, driver-vehicle cooperation, shared control, haptic shared control, model predictive control, automated driving,

autonomous vehicle, situation assessment, trajectory prediction, highway merging management, vehicle longitudinal control, vehicle lateral control.

RESUME

Face à l'évolution rapide des technologies nécessaires à l'automatisation de la conduite au cours de ces dernières années, les grands constructeurs automobiles promettent la commercialisation de véhicules autonomes à l'horizon 2020. Cependant, la définition des interactions entre les systèmes de conduite automatisée et le conducteur au cours de la tâche de conduite reste une question ouverte. L'objectif de cette thèse est de concevoir, développer et évaluer des principes de coopération entre le conducteur et les systèmes de conduite automatisée. Compte tenu de la complexité d'un tel Système Homme-Machine, la thèse propose, en premier lieu une architecture de contrôle coopératif hiérarchique et deux principes de coopération généraux sur deux niveaux dans l'architecture qui serviront ensuite de base commune pour la conception des systèmes coopératifs développés pour les cas d'usages définis.

Afin d'assurer une coopération efficace avec le conducteur dans un environnement de conduite dynamique, le véhicule autonome a besoin de comprendre la situation et de partager sa compréhension de la situation avec le conducteur. Pour cela, cette thèse propose un formalisme de représentation de la scène de conduite basé sur le repère de Frenet. Ensuite, une méthode de prédiction de trajectoire est également proposée. Sur la base de la détection de manœuvre et de l'estimation du jerk, cette méthode permet d'améliorer la précision de la trajectoire prédite comparée à celle déterminée par la méthode basée sur une hypothèse d'accélération constante.

Dans la partie d'études de cas, deux principes de coopération sont mis en œuvre dans deux cas d'usage. Dans le premier cas de la gestion d'insertion sur autoroute, un système de contrôle longitudinal coopératif est conçu. Il comporte une fonction de planification de manœuvre et de génération de trajectoire basée sur la commande prédictive. En fonction du principe de coopération, ce système peut à la fois gérer automatiquement l'insertion d'un véhicule et donner la possibilité au conducteur de changer la décision du système. Dans le second cas d'usage qui concerne le contrôle de trajectoire et le changement de voie sur autoroute, le problème de partage du contrôle est formulé comme un problème d'optimisation sous contraintes qui est résolu en ligne en utilisant l'approche de la commande prédictive (MPC). Cette approche assure le transfert continu de l'autorité du contrôle entre le système et le conducteur en adaptant les pondérations dans la fonction de coût et en mettant en œuvre des contraintes dynamiques en ligne dans le modèle prédictif, tout en informant le conducteur des dangers potentiels grâce au

retour haptique sur le volant. Les deux systèmes sont évalués à l'aide de tests utilisateur sur simulateur de conduite. En fonction des résultats des tests, cette thèse discute la question des facteurs humains et la perception de l'utilisateur sur les principes de coopération.

Mots-clés : coopération homme-machine, interaction homme-machine, coopération conducteur-véhicule, contrôle partagé, partage haptique du contrôle, commande prédictive, conception centrée sur l'utilisateur, conduite automatisée, véhicule autonome, évaluation de la situation, prédiction de trajectoire, gestion d'insertion sur l'autoroute, contrôle longitudinal des véhicules, contrôle latéral des véhicules.

REMERCIEMENTS

La thèse, activité passionnante s'il en est, n'en reste pas moins une aventure périlleuse et semée d'embûches. Ce travail aurait été bien difficile sans l'aide de plusieurs personnes que je tiens à remercier ici.

Tout d'abord, je souhaite remercier mes encadrants de thèse. Je remercie M. Jean-Christophe Popieul pour m'avoir accueilli au sein du LAMIH et pour sa confiance en me laissant libre dans mes recherches tâtonnantes. Je lui suis reconnaissant pour sa bonne humeur qui faisait de chaque rencontre des moments très agréables. Je remercie sincèrement M. Chouki Sentouh pour les connaissances qu'il m'a apportées sur l'automatique lors de nos nombreuses discussions mais aussi pour m'avoir sensibilisé à l'importance de la publication scientifique. J'exprime ma profonde gratitude à M. Jean-Baptiste Haué pour m'avoir accueilli au sein de l'équipe ergonomie cognitive au Technocentre Renault. Il m'a fait découvrir ce « petit monde merveilleux » de l'ergonomie cognitive et m'a initié à « l'art » de développer des cas d'usage.

Mes remerciements vont également aux membres du jury. Je remercie M. Arnaud de La Fortelle et M. Michel Basset pour avoir accepté d'être rapporteurs de ma thèse et pour leurs remarques constructives qui ont permis d'améliorer ce document. Je remercie également M. Andras Kemeny et Mme. Lydie Nouvelière, les examinateurs du jury pour l'intérêt qu'ils ont porté à mon travail. Merci en particulier à M. Erwin Boer en qualité de membre invité du jury, avec lequel les échanges ont été riches et agréables.

La majorité des travaux présentés dans ce mémoire ont été réalisés dans le cadre du projet « LRA » à l'IRT SystemX. J'adresse mes remerciements à Mme Sabine Langlois, chef de projet pour son support dans ce projet. Mme Langlois m'a honoré de sa confiance, en me laissant piloter une tâche et organiser deux démonstrations dans le cadre du projet. Merci à Boussaad Soualmi, ingénieur de recherche de l'IRT de me remonter le moral dans des moments difficiles, et de m'apporter sa contribution scientifique. Merci à toute l'équipe du « LRA », et particulièrement à Michel, Raïssa, Jamal, Thomas, Willy et Jean-Jacques d'avoir teinté mon quotidien de bonne humeur et d'amitié. Merci à Fenglong, ingénieur de recherche de l'IRT d'être venu me supporter pour ma soutenance.

Merci à mes collègues du service « Expériences utilisateurs » de Renault qui a su me réserver un accueil chaleureux malgré la faible fréquence de mes visites. Je tiens à remercier M. Javier

Ibanez-Guzman, chef d'équipe « véhicule autonome » de Renault pour ses conseils sur la thèse du point de vue robotique. Je voudrais aussi remercier Florent Colombet et Jean-Christophe Collinet du centre technique de simulation pour leur support en termes de simulation de conduite.

Une pensée particulière s'adresse à tous les membres du LAMIH avec qui j'ai passé des moments très agréables surtout au cours de la troisième année de ma thèse. Merci à Jérôme pour partager ses astuces sur SCANeR. Merci à Philippe pour les suggestions sur les beaux coins dans le nord. Je n'oublierai pas mes amis « docteurs et thésards » du LAMIH : Amir (je n'oublierai pas nos discussions lors des repas au RU qui m'ont beaucoup soutenu pendant des moments difficiles), Salvatore (et sa femme Gwendo, bien sûr), Lydia, Ben, Tariq, Yunfei … qui ont ensoleillé mes journées dans le nord.

Je termine par ma famille. Je tiens à exprimer mes profondes gratitudes envers M. Zhang Guo et Mme. Shuqing Li, mes chers parents. Je suis très fier d'être votre fils et vous remercie pour votre amour et votre soutien tout au long de mes études. Un grand merci à mes beaux-parents, M. Shijun Xue et Mme. Qian Liu pour leur compréhension et leur support. Mes derniers remerciements et non les moindres, s'adressent à ma femme, Shiyao. Elle a su, tout au long de cette thèse, me soutenir et m'encourager dans ma voie. Elle est la clef de ma réussite, sans elle à mes côtés, cette réalisation n'aurait pas la même saveur.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS AND ACRONYMS

ABES	Advanced Emergency Braking System
ACC	Adaptive Cruise Control
AD	Automated Driving
ADAS	Advanced Driver Assistance System
CoG	Center of Gravity
CTRA	Constant Turn Rate and Acceleration
EPAS	Electric Power Assisted Steering
ESC	Electronic Stability Control
GLRT	Generalized Likelihood Ratio Test
HFSM	Hierarchical Finite State Machine
HUD	Head-Up Display
LAR	Localization Augmented Reality
LPV	Linear Parameter Varying
MIQP	Mixed-Integer Quadratic Programming
MLE	Maximum Likelihood Estimate
MPC	Model Predictive Control
MRS	Minimum Risk State
NHTSA	National Highway Traffic Safety Administration
RMSE	Root Mean Square Error
SA	Situational Awareness
SAE	Society of Automotive Engineers
SWR	Steering Wheel Reversal
UCD	User-Centered Design

LIST OF APPENDICES

1 INTRODUCTION

1.1 Background and motivation

1.1.1 Automated driving

Technological research and development in driving automation are making continuous advances. As a result, driving automation systems are no more limited to assist the driver in the driving task, some began to take over part of the driving task (e.g., adaptive cruise control (ACC)). Recent demonstrations like DARPA Urban Challenge (Buehler, Iagnemma, and Singh 2009) and Google Car (Markoff 2010) suggest that it seems to be technically feasible to develop motor vehicles that can drive themselves without human intervention in certain situations. Along with these demonstrations, *automated driving* (AD) technology has received increasing attention from the automotive industry, research institutes, governments and becomes a popular topic in the public. Motivations behind continuous research efforts and massive investments on AD technology are the potential benefits offered by this technology—improving road safety; providing critical mobility to the elderly and disabled; increasing road capacity; saving fuel, and lowering emissions.

1.1.1.1 Levels of driving automation

Different terms such as "autonomous", "self-driving", "driverless", "automated" vehicles which are used interchangeably in the media create confusions about just what "automated driving" really means[1]. With the goal of providing common terminology for automated driving,

[1] It should be noted that the term "autonomous" is commonly used by the communities of robotics and artificial intelligence from their disciplinary perspectives.

SAE international has developed a taxonomy of levels of driving automation in SAE J3016 (SAE 2016), as shown in Tab. 1.1.

Table 1.1 Taxonomy of levels of driving automation of SAE

<table>
<tr><th rowspan="2"></th><th rowspan="2">Name</th><th colspan="2">DDT*</th><th rowspan="2">DDT fallback</th><th rowspan="2">ODD**</th></tr>
<tr><th>Sustained lateral and longitudinal vehicle motion control</th><th>Monitoring of driving environment</th></tr>
<tr><td>0</td><td>No Driving Automation</td><td>Driver</td><td rowspan="2">Driver</td><td rowspan="3">Driver</td><td>None</td></tr>
<tr><td>1</td><td>Driver Assistance</td><td>Driver & system</td><td rowspan="4">Limited</td></tr>
<tr><td>2</td><td>Partial Driving Automation</td><td rowspan="4">System</td><td rowspan="3">System</td></tr>
<tr><td>3</td><td>Conditional Driving Automation</td></tr>
<tr><td>4</td><td>High Driving Automation</td><td rowspan="2">System</td></tr>
<tr><td>5</td><td>Full Driving Automation</td><td>Unlimited</td></tr>
</table>

*DDT: dynamic driving task

**ODD: operational design domain

To understand this taxonomy, it is necessary to make clear several notions that differentiate the levels. The first one concerns the task to be automated—*dynamic driving task*. Inspired by the hierarchy of driving tasks of Michon (1985)[2], the dynamic driving task entails "all of the real-time *operational* and *tactical* functions required to operate a vehicle in on-road traffic, excluding the *strategic* functions such as trip scheduling and selection of destinations and waypoints" (SAE 2016, 5). For example, a level 2 driving automation system performs only lateral and longitudinal vehicle motion control part of the dynamic driving task, meanwhiles the human driver needs to monitor the driving environment, while systems with higher levels are capable of performing the entire dynamic driving task. The second important notion is *operational design domain* which defines specific conditions under which a driving automation system is designed to function. For example, a level 2 traffic jam assist system is designed to function in traffic jams on motorways or motorway similar roads at the speed up to 40 km/h.

[2] The hierarchy proposed by Michon will be presented in Chapter 3.

The last one is "DDT fallback", which can be understood as the capability to performing the dynamic driving task until to achieve a so-called minimal risk condition, e.g., a safe stop, after the occurrence of a system's failure or upon operational design domain exit. As shown in Tab 1.1, the division of the role to perform the DDT fallback between the driver and the system is situated between level 3 and level 4.

Based on this taxonomy, SAE uses the term "*automated driving system*" to describe a level 3, 4 or 5 driving automation system[3]. A primary distinction for an AD system is its capability to perform the entire dynamic driving task. Therefore, when an AD system is engaged, the driver no longer has to monitor the environment or the system continuously.

1.1.1.2 State-of-the-art of automated driving

Amid current commercially available driving automation systems, ACC and parking assistance with steering system reach level 1 driving automation. Some series vehicles began to offer level 2 driving automation features. Examples are the Tesla's "Autopilot" (Quain 2016), the Mercedes E-Class's "Drive Pilot" (Mercedes Benz 2015) and the BMW 7 series' "Driving Assistant Plus" (BMW 2017). These systems assume both longitudinal and lateral control on motorways or motorway similar roads. The driver still needs to supervise the system and intervene if needed.

Even though AD systems (level 3, 4 and 5) are not yet ready for commercialization, research efforts to design and develop AD technologies have been made for a long time. The European project PROMETHEUS which ran from 1987 to 1995 came to a successful conclusion with 1000 kilometers of mainly autonomous operation in normal traffic on Paris motorways, as well as a final demonstration from Munich in Germany to Copenhagen in Denmark (Dickmann, Appenrodt, and Brenk 2014). During 1980s and 1990s, the American project PATH focused on automated highway scenarios in the scope of intelligent vehicle-highway systems (IVHS) (Shladover et al. 1991). Demonstrations of PATH promoted the approach of inter-vehicle distance control based on sensors as well as inter-vehicle communications, namely cooperative adaptive cruise control (CACC). The approach of PATH has been followed in recent projects such as the SARTRE project (2012) and the Grand Cooperative Driving Challenge (GCDC) in 2011 (Nunen et al. 2012) which focused on automated vehicle platooning. The DARPA Grand Challenge in 2004 and 2005 (Iagnemma and Buehler 2006), and the DARPA Urban Challenge

[3] In a less strict sense, a level 2 driving automation systems is sometimes referred to as automated driving system in the literature.

in 2007 demonstrated autonomous ground vehicles capable of following a route to arrive at a distant location without hitting obstacles on the way. These two competitions popularized the idea of "autonomous driving" and strongly motivated industry groups and universities to develop technologies toward full driving automation. In Europe, CityMobil (van Dijke and van Schijndel 2012) and its successor CityMobil2 (Alessandrini et al. 2014) implemented automated transport systems in the urban environment. The resultant concept "cybercars" (level 4 driving automation), small-to-medium-sized automated vehicles for individual or collective transport of people or goods, is followed by the attempt of Uber (Pritchard and Krisher 2016) to apply AD technologies for taxi and ride-sharing services.

Rather than target level 4 or 5 automated vehicles that technically do not need a driver, several projects on automated driving took an incremental approach to address the driver's changing roles as the level of automation increases. Examples are the European project HAVEit (F. Flemisch et al. 2010) and the French project ABV (Sentouh et al. 2014). Both projects developped AD systems with multiple levels of automation (from level 0 to level 2) to realize dynamic driving task allocation between the driver and the AD system. The transitions between different levels can be initiated by the driver but also by the system according to the system's operating conditions or the driver's state (assessed by driver monitoring systems). This approach is followed by the European project Adaptive (2016) and the French project CoCoVeA (2013) recently.

With the rapid progression of AD technologies, the automotive industry and consulting institutes share an optimistic vision. Automakers consecutively promised the commercialization of "autonomous vehicles" or "driverless cars" by 2020-2025 (Behrmann 2016; Goodwin 2016; Horrell 2017). Due to the ambiguity of the terms "autonomous" and "driverless", the automation level implied by these announcements cannot be stated with certainty. Several consulting bodies forecasted more or less similar timelines for possible AD implementations with 2020 for the introduction of level 3 automation, 2025 for level 4 automation and 2030 as a possible milestone for level 5- full automation driving (KPMG 2015; Dokic, Müller, and Meyer 2015).

1.1.2 Need for driver-vehicle cooperation

Whilst the technical progress of automated driving seems on track, the understanding of the interaction between human drivers and AD systems seems much less clear. The taxonomy of SAE oversimplifies the interaction between the driver and the system by considering it as an "all-or-nothing" function allocation. Within this framework, interaction research in automated

driving has focused on the so-called “takeover scenario” where a human driver is required to take back the control when a level 3 AD systems reach the limits of operational design domain, e.g., due to the change of driving environment or a system failure (Gold et al. 2013; Lorenz, Kerschbaum, and Schumann 2014; Blanco et al. 2015; Walch et al. 2015). Indeed, this scenario could be safety critical due to human factor issues, e.g., driver’s decreased attention level and reduced situational awareness. However, few works explored other interaction paradigms for AD systems out of the framework of the taxonomy of SAE.

The idea of forming a cooperation teamwork between the human operator and the automation has been proposed since the introduction of automated systems in various fields of human activity (Hoc et al. 1994). This idea was later formalized into the framework of human-machine cooperation (Hoc 2001). Following the concept of human-machine cooperation, we intend to study an interaction paradigm in which an operating AD system can share the authority on the driving task with a human driver in a cooperative manner. We refer to this paradigm as driver-vehicle cooperation. Driver-vehicle cooperation could bring about the following potential benefits to the human driver and the AD system.

Potential users of the first-generation AD systems, if commercially available in 2020 as promised, could be those drivers who have already developed their own driving skills and driving styles in manual driving. They may have expectations or judgements on driving behaviors performed by AD systems in reference to their own experiences in manual driving. Especially considering current complex safety concepts for AD systems (Hörwick and Siedersberger 2010), automated vehicles could behave over conservatively in certain situations compared to human drivers. A driver that monitors the automation may have needs to intervene in AD mode, if he is not satisfied with the current driving behavior of his vehicle. Driver-vehicle cooperation enables the AD system to share the authority on the driving task. Instead of deactivating the AD system, the driver can directly modify vehicle’s behavior through the shared control. In this sense, driver-vehicle cooperation aims at not only reducing the interference between the driver and the AD system, but also enhancing the user experience of the driver by offering new interaction ways with the AD system.

From the perspective of the driving performance, driver-vehicle cooperation aims at exploring the synergy between humans and machines to improve the performance of the overall system. It can be foreseen that automated vehicles will share the current road infrastructure with other road users such as conventional vehicles, motorcyclists and pedestrians at the first stage of their

deployment[4]. Human driving activities exhibit strong social patterns, e.g. cues like eye contacts and hand gestures, which are difficult for a machine to interpret (Färber 2016). There are also culture-related conventions which are not prescribed by traffic rules. Compared with a machine, a human driver can detect these patterns easily and make an adequate decision following social rules. Therefore, by allowing the human driver to intervene, the AD system can benefit from the help of the human driver to better handle interactions with other road users, e.g., human driven vehicles, pedestrians and cyclists.

1.2 Context of this thesis

The work in this thesis was supported by the CIFRE (French acronym: Convention Industrielle de Formation par la Recherche) fellowship between Technocentre Renault and LAMIH UMR CNRS 8201 (Laboratory of Industrial and Human Automation control, Mechanical engineering and Computer Science) of University of Valenciennes and Hainaut-Cambresis.

The research described in this thesis was carried out in the academic/industrial research project "Localization - Augmented Reality (LAR)" at the Technological Research Institute (IRT) SystemX ("LRA | IRT SystemX" 2017). The LAR project has two main research objectives:

- Study new interactions and interfaces between the driver and the vehicle, notably based on augmented reality technology for automated vehicles (automobile part, which was considered in this thesis);
- Study location systems procuring a cost/performance/security breakthrough (railway part).

In its automobile part, LAR addressed the usage of level 3 automated driving in highway scenarios. The work of this thesis contributed to *Task 3.2 – Driving supervisor* of LAR. The purpose of this task was to design and simulate a vehicle guidance and control system, namely driving supervisor. The driving supervisor had three major roles in the project: 1) to ensure the control of a level 3 automated vehicle in the defined use cases, 2) to implement the cooperation principles developed in this thesis, and 3) to provide certain information concerning system states and driving environment to the HMIs developed in other tasks.

[4] It takes a long time for new vehicle features to penetrate the vehicle fleet. For example, electronic stability control (ESC) was introduced in the United States in 1995 model year vehicles, but until 2013 ESC was standard or optional on only 42% of registered vehicles in the U.S (Highway Loss Data Institute 2016).

The work of this thesis was supported by other tasks in LAR. Especially, a catalogue of use cases developed in LAR helped us to identify two use cases for cooperation design. Within a multidisciplinary team of human factors and designers, we designed the HMIs through which drivers can cooperate with the designed system according to a cooperation principle in a use case.

1.3 Objective and proposed approach

1.3.1 Objective

The objective of this thesis is to *design*, *develop* and *evaluate* cooperation principles for AD systems. At the interaction level, a cooperation principle describes how an AD system shares the authority on the driving task with a driver. At the functional level, a cooperation principle is used to guide the design of control functions of AD systems in such a way that these systems do not perform the driving task in a closed loop but accept and react appropriately to possible interventions of the human driver. Therefore, we aim not only to propose cooperation principles but also to apply these principles to the design of cooperative control frameworks for AD systems. Finally, the designed frameworks shall be prototyped and evaluated in user tests to generate insights on how users perceive and interact with these cooperative systems.

1.3.2 Research assumptions

The concept of automated driving comes up with a wide variety of possible applications and involves a set of sophisticated functionalities (Ibañez-Guzmán et al. 2012), therefore some assumptions need to be made to delimit the scope of our research. As the thesis work is carried out in the project LAR, most of these assumptions are in line with the requirements of the project.

1) *Level of driving automation*: it is assumed to be SAE level 3[5], i.e., the AD system performs the entire driving task in its operational design domain and the human driver need not monitor the driving environment in AD mode. Except when mentioned explicitly with the level of automation, the term “AD system” hereafter refers to level 3

[5] Since the project proposal was drafted in 2013, the automation level in LAR was aligned with the level 3 in the taxonomy of NHTSA (NHTSA 2013). NHTSA level 3 is equivalent to SAE level 3, hence the level of automation is harmonized to be SAE level 3 in this report.

AD system in this report. A vehicle equipped with an AD system is referred to as an *AD vehicle*.

2) *Perimeter use cases*: Perimeter use cases delimit the perimeter of use of the AD system. In this thesis, the perimeter of use concerns automated driving on a multiple-lane highway at speeds of up to 130 km/h (road speed limit for highways in France). Totally 14 perimeter use cases were identified in LAR. A perimeter use case corresponds to a situation of use of the AD system. It helps to identify potential user needs for HMI contents and potential interferences between the driver and the AD system by taking driver's position in a real driving situation.
3) *Operating conditions*: the AD system is assumed to operate in nominal conditions. Nominal conditions can be roughly understood as that AD system operates in this perimeter of use (perimeter use cases), the information on the external environment which is necessary for the system to operate is available (e.g., detectable lane marks) and there is no internal system failure.
4) *Perception system*: the environment perception is not in the scope of this thesis, and it is assumed that the information of the environment is given by a perception system based on sensor data, a high-accuracy digital map and vehicle-to-vehicle/infrastructure communication. More details on these assumptions are given in Chapter 4.

1.3.3 Research approach

As illustrated by Fig. 1.1, the proposed research approach can be divided into two steps. In the first step, we propose general principles and a functional control architecture for driver-vehicle cooperation design. These general propositions are intended to offer a global view on levels of cooperation and possible extensions of plan and control functions relevant to automated driving toward driver-vehicle cooperation. Furthermore, these general principles and the functional architecture can be used to drive different designs of cooperative systems in different use cases. The work in this step is presented in Chapter 3.

In the second step, we design cooperative control frameworks for AD systems based on the proposed principles and the functional architecture. Considering that interactive systems that utilize advanced technologies can rarely be completely specified in advance, we adopt the *user-centered design* (UCD) approach, an iterative design approach with the active involvement of users (Norman and Draper 1986). As shown in Fig.1.1, the UCD approach adapted in this thesis consists of four steps:

- *Use case definition*: a use case formalizes the context of the use of the system to be designed. It also serves to analyze user needs on cooperation and potential interferences between the driver and the system in the driving task. We define use cases based on the perimeter use cases provided by LAR.
- *Specification*: we implement a cooperation principle in a defined use case, and specify the functions to be developed based on the general functional architecture.
- *Development*: in each use case, we develop a cooperative control framework to achieve the specified functions. In this process, we apply techniques from human supervisory control or shared control. The developed framework is then prototyped in the driving simulation environment.
- *User study*: we conduct a user study on driving simulator to evaluate how users interact with the designed system in the driving task. The evaluation results are used to identify potential problems of the designed systems so that they can be appropriately rectified in the next iterative process.

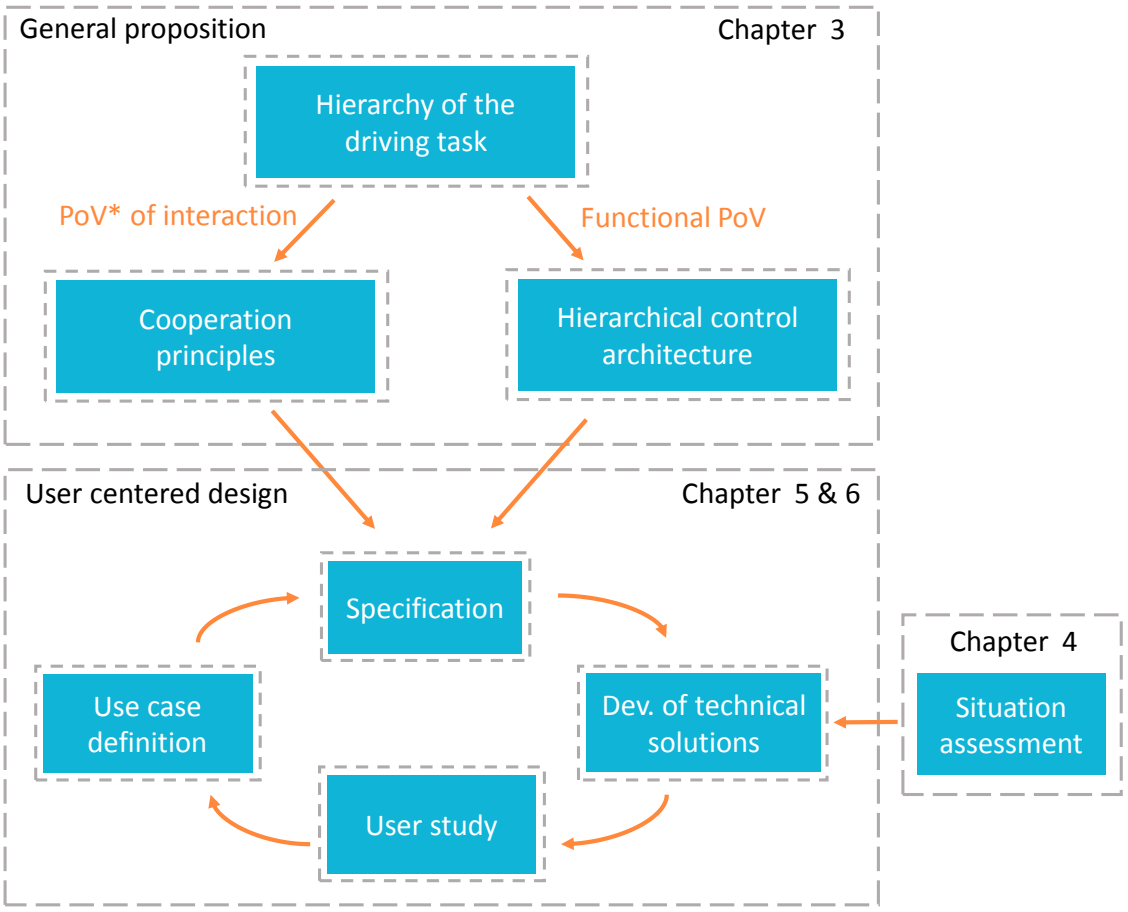


Figure 1.1 Proposed approach to design cooperative control systems for AD systems

Chapter 5 and 6 present two case studies in which we implemented UCD to design cooperative control functions for AD systems. In these use cases, the AD vehicle operates in a dynamic environment involving traffic vehicles. To enable high-level cooperation with the driver on driving maneuvers, e.g., to handle the interaction with those traffic vehicles, the AD system needs to understand the external environment and to anticipate the evolution of the situation. We address this issue by proposing a *situation assessment* function specific to highway driving scenarios (the perimeter of use of the AD system) in Chapter 4. This common function will be then used by the two cooperative control frameworks designed in Chapter 5 and 6.

1.4 Outline of the thesis

The organization of chapters follows the procedure illustrated in Fig. 1.1.

Chapter 2 begins with presenting the concepts and methods that support the design of driver-vehicle cooperation in this thesis. They include the framework of human-machine cooperation, the approach of UCD and the shared control scheme. The presentation of theoretical concepts is then followed by a review of previous works that explored the concept of driver-vehicle cooperation.

Chapter 3 addresses driver-vehicle cooperation as a whole by proposing a functional architecture for cooperative control. Based on this architecture, we discuss how to map planning and control functions used in automated driving to different cooperation levels. We also propose two cooperation principles at two levels in this architecture.

Chapter 4 describes a situation assessment function for highway driving scenarios which provides information on the driving scene to the control functionalities of the cooperative systems in Chapter 5 and 6. This function consists of a representation formalism of the driving scene in the Frenet frame and a long-term trajectory prediction component.

Chapter 5 presents the design of a cooperative longitudinal control framework in a use case of highway merging management. This framework features an ad-hoc maneuver planning function and a model predictive control (MPC) based trajectory generation for transient maneuvers. Following one of the proposed cooperation principle, this framework allows the driver to change the maneuver plan of the system during the interaction with a merging vehicle. The results of a user study of the designed framework are discussed at the end of this chapter.

Chapter 6 deals with another use case on highway lane positioning and lane changing. In this use case, we present our design of a cooperative steering control framework based on the haptic shared control scheme. Through the implemented principle, the driver can take over the steering

control without deactivating the system, benefit from the system's aid during a lane change maneuver and receive haptic alarms when his action could cause the danger. Based on the results of a preliminary user test, we discuss some design issues at the end.

Chapter 7 finally concludes by summarizing the thesis, and by providing perspectives for further work.

1.5 List of publications

The following part gives a list of publications that follow from this research work:

Journal articles

1. **C. Guo**, C. Sentouh, J.-C. Popieul et al.,"Cooperation between Driver and Automated Driving System: Implementation and Evaluation," Transportation Research Part F: Traffic Psychology and Behaviour (2017). (in press).

Conference articles

1. **C. Guo**, C. Sentouh, J.-C. Popieul, and J.-B. Haué, "MPC-based Shared Steering Control for Automated Driving Systems," in 2015 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017. (accepted)
2. **C. Guo**, C. Sentouh, J.-B. Haue, and J.-C. Popieul, "Predictive shared steering control for driver override in automated driving: a simulator study," in 2017 Driving Simulation Conference (DSC), Stuttgart, Germany, 2017 (accepted).
3. **C. Guo**, C. Sentouh, J.-C. Popieul et al., "Cooperation between Driver and Automated Driving System: Implementation and Evaluation," in 2016 Driving Simulation Conference (DSC), Paris, 2016.
4. **C. Guo**, C. Sentouh, B. Soualmi, J.-B. Haue, and J.-C. Popieul, "Adaptive Vehicle Longitudinal Trajectory Prediction for Automated Highway Driving," in 2016 IEEE Intelligent Vehicles Symposium, Gothenburg, Sweden, 2016
5. **C. Guo**, C. Sentouh, J.-C. Popieul, B. Soualmi, and J.-B. Haué, "Shared Control Framework Applied for Vehicle Longitudinal Control in Highway Merging Scenarios," in 2015 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2015, pp. 3098–3103.
6. **C. Guo**, C. Sentouh, B. Soualmi, J.-B. Haue, and J.-C. Popieul, "A Hierarchical Cooperative Control Architecture for Automated Driving Systems," in 22nd ITS World Congress, Bordeaux, France, 2015.

National conference presentation

1. **C. Guo**, C. Sentouh, B. Soualmi, J.-B. Haue, and J.-C. Popieul, "Conception et simulation de fonctions du véhicule autonome – cas d'étude : gestion d'insertion sur l'autoroute," in GTAA Seminar, Lyon, France, 2016.

2 THEORETICAL FRAMEWORK FOR DESIGNING DRIVER-VEHICLE COOPERATION

2.1 Introduction

This chapter gives a brief overview on the concepts and methods that support the designing of driver-vehicle cooperation principles in this thesis. It is divided into three parts.

The first part introduces two important frameworks, the framework of human-machine cooperation proposed by J.-M. Hoc (Section 2.2) and the approach of UCD (Section 2.3), through which this thesis approaches the design of driver-vehicle cooperation. Human-machine cooperation puts human and machine (automation in our case) in a human-machine system and views task distribution within the system as an activity of interference management. It is based on this interpretation of "cooperation" that we propose principles for driver-vehicle cooperation on driving tasks. UCD approach, on the other hand, emphasizes that the purpose of the machine is to serve the user by offering him possibility of actions in his flow of experience. Section 2.3 gives a short introduction of UCD approach. We highlight the concept of use case and the role of user test, and how we use them in our design work.

The second part (Section 2.4) presents two control schemes—shared control and human supervisory control—that can serve as technical solutions to develop cooperative systems. The last part focuses on previous works that addressed shared authority issue between the driver and the automation from a cooperation perspective.

2.2 Human-machine cooperation

In the field of automation, the question of "human-machine cooperation" is raised in design of automation in complex control systems which need to be supervised and managed by human operators. Instead of taking a machine-centered point of view, i.e., automate whatever can be automated, leaving the rest to human (Bainbridge 1983, 775), *human-centered automation* puts the "system" surrounding human and machine, and considers that a task is performed by the overall "human-machine system". In this way, human-centered automation does not isolate the design of the automation, but accords specific attention to interactions between the automation and the human operator. The concept of human-machine cooperation is developed based on human-centered automation, hence this section begins by introducing some important concepts of human-centered automation.

2.2.1 Task allocation and level of automation

At the core of the human-centered automation design is the *task allocation* between the automation and the human operator, i.e., which tasks should be automated and which tasks should remain to be done manually. One of the well-known approach to deal with task allocation is to define levels of automation (Sheridan and Verplank 1978). A level of automation designates a degree to which a task can be automated. Various levels of automation together form a taxonomy of level of automation. Different taxonomies of level of automation were proposed in the literature (Sheridan and Verplank 1978; Endsley and Kaber 1999; R. Parasuraman, Sheridan, and Wickens 2000). They share the same "pattern" in general. First in a vertical direction, those taxonomies cover from a bottom level of manual control to a top level of full automation while including several intermediate levels. Second in a traversal direction, the tasks are modelled in a "perception-action" circle borrowed from the control theory. For instance, Parasuraman (2000) grouped the tasks in four categories: information acquisition, information analysis, decision and action selection as well as action implementation. One example of the taxonomy of level of automation is shown in table 2.1. As presented in Chapter 1, the taxonomy defined by SAE for driving automation follows the same pattern.

Table 2.1 Taxonomy of level of automation defined by Endsley (1999)

	Roles			
Level of automation	Monitoring	Generating	Selecting	Implementing
1) Manual control	Human	Human	Human	Human
2) Action support	Human/Computer	Human	Human	Human/Computer
3) Batch processing	Human/Computer	Human	Human	Human
4) Shared control	Human/Computer	Human/Computer	Human	Human/Computer
5) Decision support	Human/Computer	Human/Computer	Human	Computer
6) Blended DM*	Human/Computer	Human/Computer	Human/Computer	Computer
7) Rigid system	Human/Computer	Computer	Human	Computer
8) Automated DM	Human/Computer	Human/Computer	Computer	Computer
9) Supervisory control	Human/Computer	Computer	Computer	Computer
10) Full automation	Computer	Computer	Computer	Computer

*DM : decision making

2.2.2 Adaptive automation

If a taxonomy of level of automation reveals possible ways for task allocation, the real difficulty lies in choosing a target level. The machine-centered point of view drives the designer to seek a level based on the technical capabilities[6]. Human-centered automation design, in contrast, highlights the human operator's performance in the evaluation criterion for the choice. If the potential benefits of automation are often intended, it is those "unintended" negative effects on human performance that pose serious challenges to design an effective system (Billings 1997, 183).

A main concern of negative effects of automation on human's performance is the so-called "out-of-the-loop" performance problem (Endsley and Kiris 1995). Experiments and human factor analysis showed that human's performance is degraded when the human operator is removed from a control loop. This performance degradation is manifested by: attention level decrement, complacency (over-trust in automation), loss of situational awareness and skill decay in the long term.

[6] It seems that the development of AD systems in the automobile industry follows this technical-first perspective, i.e., a progress of level of driving automation in the taxonomy of SAE according to the technical advancement.

Adaptive automation, also referred to as *dynamic function allocation*, consists of one of the attempts of human-centered automation aiming to overcome the out-of-the-loop performance problem (Millot and Kamoun 1988; Inagaki 2003; Kaber and Endsley 2004). The principle of adaptive automation is continually adjusting the level of automation in order to keep the operator in the loop. For example, if an operator's workload becomes high and consequently his performance drops, the level of automation needs to be enhanced, i.e. the automated system takes over the tasks augmenting human workload. On the contrary, if the human operator is detected to be drowsy at a high level of automation, some tasks can be distributed to him to maintain his attention. How to implement adaptive automation remains a complicated task. A main contribution of adaptive automation is the creation a mechanism of dynamic task allocation, which is inherited by human-machine cooperation.

2.2.3 Definition of "human-machine cooperation"

Human-machine cooperation offers a new perspective to consider the relation between the human and the automated system. The concept of the human-machine cooperation emerged in the context of automating complex tasks in dynamic situations (industrial process control, air traffic control, highly automated aircraft piloting, etc.). In face of the complexity of the environment, designers developed automated systems mimicking human-like cognitive abilities such as situation assessment and decision-making. Hollnagel and Woods (1999) defined such human-machine systems as a type of "joint cognitive system". A human operator tends to show a cooperative attitude towards the automation that behaves like a human according to Nass, Fogg, and Moon (1996). Thus, human-machine cooperation was studied based on the analogy between human-human and human-machine relations.

For example, Schmidt (1991) summarized different forms of interaction to help understand the cooperative work of human beings: construction and maintenance of a reciprocal awareness, orientation of attention of others, and negotiation between the actors. Consequently, these forms of interaction change the tasks distributed between each actor and therefore their roles in cooperative work. Regarding these statements, particularly the fact that one actor adapts his activities in a cooperative work so as to facilitate the tasks of other actors, Hoc (1996, 2001) gave the following definition of cooperation:

> *Two agents are in a cooperative situation if they meet two minimal conditions.*
>
> *1) Each one strives towards goals and can interfere with the other on goals, resources, procedures, etc.*

2) Each one tries to manage the interference to facilitate the individual activities and/or the common task when it exists.

The first condition highlights the crucial role of "interference" in cooperation. Hoc borrowed here the physics meaning of "interference" as a metaphor. Two waves can reinforce each other if they are in phase or, on the contrary, weaken each other if they are not. The interference occurring during the cooperation is due to the interdependence between goals of the actors. Positive interference favors the achievement and maintenance of the goals of actors. Negative interference may lead to conflicts that affect the performance, augment the workload, and can even undermine the operation safety. To manage the interference means to reinforce the positive interference while reducing the negative one. The second condition reveals another characteristic of cooperation, i.e., each one adapts its activities to facilitate those of others. This implicates a minimal symmetry between human and automation in the framework of human-machine cooperation. If a human operator can be assisted by the automation, the inverse case is also true. Under such a symmetry, human-machine cooperation aims to exploit the synergy between human and automation.

Nevertheless, human-machine cooperation cannot be fully symmetric because responsibilities cannot be shared. The final authority needs to be explicitly predefined to account for potential conflicts. Moreover, "who" is responsible for the overall performance of such a human-machine system also needs to be considered. To answer these questions are not trivial in that it necessitates comprehensive examinations of technical, economical and legal aspects.

Starting from this definition, Hoc addressed human-machine cooperation in a cognitive approach. One fundamental property of a cognitive system is the use of internal representations of the external environment. An efficient cooperation, from a cognitive point of view, relies on a shared representation of the task environment maintained by the actors in cooperation. Within human-machine system, this shared representation, called *common frame of reference* (COFOR) by Hoc, offers a *common ground* or *common reference* for human and automation to share goals, plans and intentions to perform cooperative activities.

In summary, a cognitive perspective of human-machine cooperation leads to the following principles for driver-vehicle cooperation design in this thesis:

1) Human-machine cooperation is an activity of interference management. *An efficient human-machine cooperation should reinforce the synergy of human and automation (positive interference), and at the same time mitigate conflicts of their activities (negative interference).*

2) *Who holds the final responsibility should be clearly defined in human-machine cooperation*.
3) *An efficient human-machine cooperation should generate and maintain shared representations on external situation and goals between human and automation.*

2.3 Support from user-centered design approach

Stemming from human-centered automation, human-machine cooperation integrates human and automation into a human-machine system and considers the system performing a task as a whole. By formulating the dynamic task allocation between human and automation as the cooperative activity, human-machine cooperation aims at enhancing the performance of the overall system.

UCD, on the other hand, views human using technical system as a resource of his action. With a user's perspective, UCD helps find user's needs and the usability of a system. Its benefits for driver-vehicle cooperation design consist in making the cooperative system easy to understand and use by human drivers. For this reason, it is worthy making a tour of UCD to see how it can support the design of an automated system that interacts with a user.

2.3.1 User-centered design approach

UCD concerns how to design effective technical systems that are intended for human use. The concept of UCD emerged in the context where technical systems designed solely through a technological-driven approach posed problems for users and sometimes could lead to serious consequences like accidents (Parasuraman and Riley 1997). The concept of UCD firstly proposed by Norman and Draper (1986) highlighted that the design of interactive systems should be guided by user needs rather than technological possibilities. Since then, UCD draws on multiple sources of knowledge from cognitive and social psychology, human factors and ergonomics, and computer sciences to support creating systems that are based on user's characteristics and possible situations of use. Nowadays, the design process based on UCD has been normalized in ISO 9210-210 (ISO 2010). In practice, a design process based on UCD has the following features:

- Understanding and formalizing the context of use (including users, tasks and situations of use);
- Involving the user in the design process actively;

- Conducting user-centered evaluations of the design solutions and modifying the design based on user feedback, thus forming iterative design cycles rather than linear, rigid design process.

Within the scope of UCD, the concepts of use case and user test are used in this thesis. They are presented in the following sections.

2.3.2 Use case

Use cases are widely employed in both object-oriented software engineering and HMI design. However, use case has not a precise definition mainly due to its high adaptability to various applications (Constantine and Lockwood 2001). In this thesis, use cases are related with situations of use of the designed system. The adoption of this point of view is motivated by the contribution from cognitive ergonomics. As a branch of cognitive ergonomics, the theory of "structural coupling" assumes that the activity of the user, based on his past experiences and the present environment, gives the meaning of the technical systems that he uses (Maturana and Varela 1992). His engagement in the situation filters the information that he perceives and directs his use of the system. In this sense, the expected function of the technical system depends not only on the needs and characteristics of the user but also on the situation in which the system is used. From this perspective of cognitive ergonomics, the fundamental idea behind designing technical systems is to design future situations of use that will enrich the user's experience and to which the user can adapt easily.

To find future situations in which the driver would cooperate with the AD system, we start by analyzing existing driving situations. We formulate an existing driving situation as a use case in which we analyze how a human driver interacts with the driving environment (road infrastructure, road conditions, traffic …) in manual driving. Then we identify potential user needs emerging in this concrete driving situation. Finally, these needs shall be addressed when we design cooperative functions for the AD system. Fig. 2.1 shows an example on how we exploit a use case in our design work. This use case deals with highway merging management which will be presented in detail in Chapter 5. We begin by analyzing driver's needs when he faces a merging vehicle. The following hypotheses can be made based on empirical observations:

1. He may want to preserve his comfort. Thus, he may expect the AD system to maintain a constant speed to pass the merging vehicle.
2. He may want to show his courtesy or respect to social conventions. He may expect the AD system to make a lane change or reduce its speed to yield.

3. In ambiguous situations, he may expect the AD system to take an initiative by manifesting its intention.

From this analysis, we can also suppose that a driver may need to know if the system is monitoring this merging vehicle and what is the intention of the system. Based on these assumptions on user needs, we can sketch a future situation of use for driving-vehicle cooperation in which the system shows its intention towards the merging vehicle and the driver can indicate his intention in return. Note that there are formal methods in the discipline of cognitive ergonomics to perform empirical observations such as the self-confrontation interview (Theureau and Jeffroy 1994) and explicitation interview techniques (Vermersch 1994)[7].

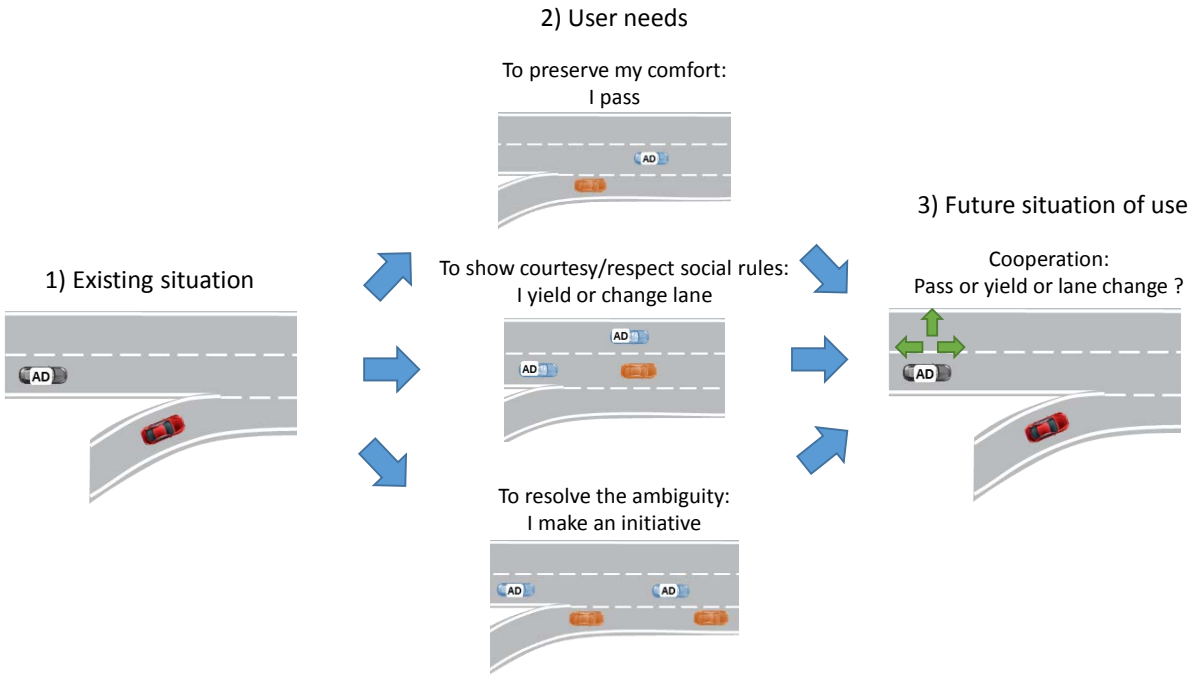


Figure 2.1 Example of a use case driven approach

From a technical standpoint, use cases serve to instantiate general cooperation principles, i.e., to specify system functions and HMI contents. Moreover, since this thesis was carried out in a project with a team of multidisciplinary competences, use cases served as common framework for scenario modelling, functional and HMI design.

[7] French-speaking ergonomists have long been familiar with these techniques in the activity analysis.

2.3.3 User test

User test is an essential part of UCD. It allows the designer to evaluate how users interact with the designed system in the intended situation of use. Users, through their adaptability and creativity, bring transformations of the interaction intended by the designers. They may use systems in ways that could never be conceived of by their designers. Particularly in our case, we envision a future situation of use for driver-vehicle cooperation which does not exist yet. User test could provide evidence for the design choice and assumptions. Lastly, performing user tests at an earlier stage of development helps avoid potentially costly mistakes at the final stage before the deliverance of the final product.

Utility and usability are two important dimensions to evaluate a designed system that interacts with a human. Utility, also referred to as usefulness, responds to the question "whether the functionality of the system in principle can do what is needed" (Nielsen 1994, 25). Utility can be quantified in terms of the frequency and the way of the use of the system. It can also be measured by subjective evaluations, such as subjective scales on the usefulness.

Usability has already been a concept standardized by ISO, which defined usability as "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use" (ISO 2010). In order to make the usability of a system quantifiable, Nielsen (1994) defined five usability attributes:

- *Learnability*: the degree of being easy to learn,
- *Efficiency:* the amount of effort required to complete the task,
- *Memorability*: the degree of being easy to remember,
- *Errors*: those the user make during the use of the system,
- *Satisfaction*: the degree to which the user was happy to use the system.

During a usability test, the usability is assessed via user's performance metrics to complete tasks based on the system to be evaluated. Common performance metrics are rate of task success, time on task, user errors, user workloads, etc.

2.4 Extensions of control theory towards human-machine cooperation

While the two precedent sections discussed two theoretical frameworks which provide important design guidelines for our design of driver-vehicle cooperation, this section focuses on how to develop a cooperative control system from a control perspective. We identify two control schemes which consider human's control in the control-loop of the system. *Shared control* scheme applies to the case where human control action is continuous and enters directly

in the control loop, where in *human supervisory control* the human operators provides symbolic commands, e.g., button presses, that can influence the control of the automation.

2.4.1 Shared control

Shared control scheme applies conventional control theory to incorporate human's manual control in the control loop. It aims to determine the final command for a *plant* (usually a process or an actuator to be manipulated) when both human and automation control the plant simultaneously. Depending on the type of control interface, shared control approaches in the literature can be classified into two categories (Abbink and Mulder 2010): *blended shared control* and *haptic shared control*.

2.4.1.1 Blended shared control

Blended shared control scheme is applied to those control interfaces having no direct mechanic connection with the controlled plant. This kind of interface is also referred to as *by-wire systems*, e.g., steering-by-wire system and electronic throttle. Since the human's control is in form of electronic signal, it can be augmented or reduced in the automated system, thus making the blending formalism highly flexible.

Various blending formalisms exist in the literature. Among others, blending by weighting is the most popular approach. This blending formalism can be expressed by

$$u_{final} = (1 - \alpha)u_h + \alpha u_a, \tag{2.1}$$

where u_{final}, u_h and u_a designate final control input, controls of the human and the automation respectively. The weighting factor α, called *blending policy* (Dragan and Srinivasa 2013), determines the shared control authority. Urdiales et al. (2010) designed a blending policy based on the "efficiency" of the human's input in an application for wheelchair's navigation guidance. The "efficiency" is characterized by three factors of human input: "smoothness" (angle between the current direction of the robot and the provided motion vector), "directiveness" (angle formed by the robot heading and the direction towards the goal), and "safety" (distance to the closest obstacle). Anderson et al. (2013) designed a steering controller to support the driver for hazard avoidance. The blending policy in the shared control is a piecewise linear function of the front tire slip angle, which is a metric of vehicle's dynamics stability. Sentouh et al (2013) developped a cooperative steering assist controller which blends human driver's control based on a metric characterizing the conflicts between the driver and the system.

Other blending formalisms can be found in the literature as well. Cerone, Milanese, and Regruto (2009) developed a feedforward controller that predicts the lateral error of vehicle's position to

the lane center under the influence of driver's steering torques. The predicted lateral error compensates the real lateral position error, hence reducing the control of a feedback lane-keeping controller. In this way, the driver can take over the control without switching off the controller. Chipalkatty and Egerstedt (2010) and Erlien, Fujita, and Gerde (2016) formulated a shared control problem in the framework of MPC. A MPC controller seeks to identically match the human's commands if the plant's states fall within safety constraints. If the controller predicts that the human's commands will steer the plant's states out of safety zone, it adapts its control to guarantee the safety. A logic of blending a part of control to track human's commands with another part to ensures the safety is formulated implicitly in the problem representation of MPC.

The blended shared control scheme benefits from the design flexibility offered by the by-wire system. First, the human's input in form of electrical signal can be adjusted directly at the blending level. Second, the desired plant behavior and the force feedback to the human user can be designed independently because there is no mechanical link between input and output. However, the human operator may not be aware of the automation's activity, because the blending occurs in the controller but not on the physical control interface. To overcome this limit, feedback channels, e.g., haptic feedback on control interface, visual feedback, need to be added to make automation's control easy to understand.

2.4.1.2 Haptic shared control

One definition of haptic shared control is given by Abbink and Mulder (2010). Haptic shared control "… allows both the human and the [automation] to exert forces on a control interface, of which its output (its position) remains the direct input to the controlled system." The core of this definition is that human and automation are in physical interaction (by force). This characteristic distinguishes haptic shared control from blended shared control.

Two design perspectives of haptic shared control can be identified in the literature. With the first perspective, the computed force is used as a force feedback for a human operator who is responsible for controlling the plant. In this sense, the force feedback aims either to provide guidance or to convey the information on the environment to the human operator. From this perspective, the force feedback is usually computed in the framework of *impedance control* (Hogan 1985). Impedance can be perceived as an ability of a manipulator (e.g., human's arm) to resist the external disturbance. It can be modeled as

$$F_{imp} = K^d(\mathbf{x}^d - \mathbf{x}) + B^d(\dot{\mathbf{x}}^d - \dot{\mathbf{x}}), \tag{2.2}$$

where F_{imp} is impedance, $\mathbf{x}^d$ and $\dot{\mathbf{x}}^d$ are desired motion (position and velocity), $\mathbf{x}$ and $\dot{\mathbf{x}}$ are actual motion, and K^d and B^d are stiffness and viscosity, constituting the desired impedance characteristics. Abbink (2006) and Mulder et al (2011) focused on understanding human's neuromuscular responses to forces. In this manner, they designed haptic controllers that could "mirror" the impedance adaptability of human muscle. Takada, Boer, and Sawaragi (2013) proposed a framework of shared haptic steering control from a user's perspective—integrating the designed system into driver's existing cognitive-action-cycle. In a use case of passing through narrow paths, they implemented this framework that facilitated driver's epistemic probing activity while communicating the information on lateral deviation and distance to obstacles to the driver by force feedback.

The second design perspective is more oriented to human-centered automation. A more sophisticated controller is designed to perform a target task, thereby a major design issue is the shared authority between the automatic controller and the human operator. In haptic shared control framework, each one can gain/release the control authority by augmenting/reducing the force exerted on the interface. Abbink, Mulder, and Boer (2012) provided four guidelines to address the shared authority issue. They argued that human should be able to experience (haptically adaptive automation) and to initiate (haptically adaptable automation) the smooth shift of authority. Saleh et al. (2013) and Soualmi et al. (2014) used human driver model in the control synthesis. Consequently, the controller was capable of adapting control output to the driver's actual control. Mörtl et al. (2012) recognized the redundancy of control inputs of multiple actors (human and robots). They decomposed the redundant input forces into an external force part and an internal force part. The internal forces arise from control mismatch between the actors. Therefore, they can be considered as a *haptic negotiation channel* to communicate disagreement and motion intentions. From this perspective, they estimated human's internal force and adapted robots' efforts to reduce human's effort. Nguyen, Sentouh, and Popieul (2016) modulated the steering torque of an automatic steering controller according to the driver's involvement level and drowsiness state. Thus, the controller increased its control when the driver was in underload (prone to be drowsy) and overload conditions.

A main advantage of haptic shared control is that human can be informed of automation's activity in a direct and intuitive manner (through haptics). Meanwhile, the shift of authority is seamless and smooth. Haptic shared control methods proposed in the literature often rely on specific paths and therefore need a specific goal or way points. When the human operator has a different goal than that of the system, conflicts (negative interference) may arise, thus workloads of the human operator increase (Boink et al. 2014). The methods cited above

mitigated conflicts in a reactive manner, i.e., the controller reduced its control input when conflicts arise. These methods did not consider the intention of the human operator which could be different from that of the automation, nor did they consider the risks due to human's action in a dynamic environment. From a human-machine cooperation perspective, haptic shared control framework needs to allow the human operator and the automation to share their goal/intentions and situational awareness behind their control actions.

2.4.2 Human supervisory control

In human supervisory control scheme, the human operator is given a role of supervisor. He provides automation specific goals or waypoints prior to or during the operation, while automation is responsible for controlling the plant to achieve the designated goals or waypoints[8]. In contrast with shared control scheme where human input is continuous, human intermittently intervenes in the control process to give a new goal. In this way, the main task of human is to monitor automated tasks instead of directly participating in control activities.

Human supervisory control has widely been applied for telerobotics, process control (Sheridan 1992). It began to draw attention in the automotive community along with the progress of driving automation. A representative example is the "active lane change assist" application which has already been commercially available. Through this application, the driver can initiate a lane-change maneuver performed by the driving automation system by switching the turn signal (Handelsblatt Global 2015; Quain 2016). Concerning research projects, Geyer et al. (2011) proposed and implemented a "Conduct-By-Wire" principle. In this principle, the driver assigns maneuver commands via a so-called maneuver interface that enumerates all possible maneuvers in the applied use cases (see Fig. 2.2). The automated system then performs the selected maneuver automatically. Albert et al. (2015) evaluated four interaction concepts for AD vehicles in highway lane change scenarios: "manual lane change", "trajectory control" (the driver instantly triggers maneuvers, same with "active lane change assist"), "maneuver planning" (the driver plans a stack of upcoming maneuvers), and "automatic lane change". The user test results suggested the driver's preference to transfer as many as tasks to the AD system.

Human supervisory control scheme offers the human operator simple methods to interact with the automation when he is not in direct control of the system. In this sense, it is more suitable

[8] The scope of human supervisory control discussed here is limited at a higher level than operational level. The "control takeover" scenario and the teleoperation application (e.g., object manipulation through teleoperator) included in the definition of Sheridan (1992) are not considered here.

for AD applications where the driver can do secondary tasks. However, as discussed in Section 2.2.2, a main issue of human supervisory control is human's out-of-the-loop performance problem. Current research directions to remedy this problem consist of: 1) find effective strategies to reengage the driver with the control task, 2) determine the amount of time a normal driver needs to regain control and 3) design appropriate interfaces through which a driver can quickly rebuild the situational awareness (Blanco et al. 2015). Even if human supervisory control scheme offers the convenience for the driver to express directly his intention through symbolic command, it cannot address potential user needs to subtly adjust vehicle trajectory as a driver does by turning steering wheel or pressing pedals in manual driving.

Figure 2.2 Tactile touch display of a maneuver interface in the "Conduct-by-wire" project (Franz et al. 2012)

2.5 Previous works on driver-vehicle cooperation

This section reviews some previous works on driver-vehicle cooperation. Those works already cited in the scope of shared control and human supervisory control are not repeated here. The studies reviewed in this section are more oriented to cognitive frameworks or functional architectures. For each study, we focus on how it deals with the shared authority between the driver and the system in the driving task.

2.5.1 H-metaphor (Flemisch et al. 2003)

Flemisch et al. proposed a H(orse)-metaphor in which the relation between the driver and the automation is compared to the relation between a rider and a horse. When riding a horse, the rider can loosen the reins to give the horse more authority or grip the reins tighter to enforce his will. The rider and the horse infer each other's intention through the reins. The reins are a metaphor for a haptic control interface between the driver and the automation. The loose-rein control corresponds to a highly automated mode where the automation controls the vehicle in

a large part, while the driver is still in the control-loop to accompany the automation. The tighten-rein control stands for an assistance mode where the driver assumes most control authority. The design philosophy in H-metaphor is followed by haptic shared control scheme. Both argue that the human driver should remain in the control loop such as to benefit from increased performance and reduced workload.

2.5.2 Levels and modes of cooperation proposed by Hoc, Young, and Blosseville (2009)

Within the framework of human-machine cooperation, Hoc, Young, and Blosseville decomposed the cooperative activity between the driver and the automation into three levels: meta-level, planning level and action level. Cooperation in meta-level addresses the use of models of the agents (driver and automation) in the design process. The use of driver model in the design of automation aims at making the automated system understand driver's behaviors, thus adapting its control to that of the driver. Developing user's model of system's operation is expected to ensure a proper use of the system by future users. The cooperative activities at the planning level aim to maintain a common frame of reference on which the driver and the automation share their plans and goals. At the lowest level, the cooperation activity is directly related to action and corresponds to a local, concrete and short-term interference management.

A main contribution of levels of cooperation is the decomposition of the cooperative activity (authority management) which could be too complex to be studied as a single entity. Moreover, this decomposition was based on cognitive process, i.e., action level corresponding to sub-symbolic processing, plan level to symbolic processing and meta-level to a kind of long-term cognitive effect (experience). At each level, the authors designed multiple cooperation modes similar to levels of automation. But these modes were also proposed with the perspective to enhance the driver's cognitive performance rather than simply allocate functions. These modes include perception mode (to enhance driver's perception), mutual control mode (like shared control), function delegation mode (equivalent to level 1 automation of SAE) and full automation mode (level 2). Finally, the authors advocated the role of cooperation in action in the interference management, because the information is transferred through sub-symbolic processing, which is less cognitive costly compared to symbolic processing level.

This cognitive framework has its strength in modelling and analysing the interaction between the driver and the automation, e.g., it enables a new classification method for driver assistance systems (Navarro, Mars, and Young 2011). However, it is difficult to derive technical solutions for driver-vehicle cooperation from this kind of cognitive framework. A more technical-

oriented functional architecture is still needed. What's more, transitions between different cooperation modes are not discussed in this framework. Finally, this framework views the role of the driving automation as to provide temporal intervention in the driving task rather than take over the driving task on a sustained basis, thus limiting the application scope of this framework.

Some cooperation modes in this framework were implemented in the project PARTAGE which will be presented in the following section.

2.5.3 Project PARTAGE (Hoc 2012)

PARTAGE (French word meaning "share") is a French project (2009-2012) on the cooperation between driver and *advanced driving assistance system* (ADAS). The use case treated in PARTAGE was the lane departure avoidance. PARTAGE implemented three cooperation modes proposed in the last section. The first mode was based on the shared control scheme in which the driver and the ADAS continuously share the control authority. A main contribution of PARTAGE to this cooperation mode was the development of a cybernetic driver model that integrates human driver's anticipatory capacity and neuro-muscular dynamics of human arm (Sentouh et al. 2009). It was also proved that the shared control law integrating this driver model improved the overall performance of the human-machine system. The second mode, namely "motor priming mode", rendered haptic vibrations with discrete levels of amplitudes on the steering wheel. The vibrations were triggered based on the risk metrics of lane departure, aiming to prompt the driver to correct the trajectory. The third cooperation mode was the "corrective mode" in which the system intervenes to correct vehicle trajectories, in case of the lane departure or the loss of stability.

In addition to the contributions in terms of the control aspect, PARTAGE also addressed the human factors aspect of driver-vehicle cooperation. These efforts included proposing indicators for cooperation performance, analysing the common reference for risk assessment between the automation and the driver, and evaluating the acceptability of the different cooperation concepts.

2.5.4 Project HAVEit (Hoeger et al. 2008)

The EU funded research project HAVEit (Highly automated vehicles for intelligent transport, 2008-2011) developed and demonstrated several concepts of highly automated driving. The main use case treated in HAVEit was highway automated driving. In HAVEit, the AD system is referred to as "co-system", and was developed within a joint system with the driver. The co-system in HAVEit had multiple discrete levels of automation, as shown in Fig. 2.3. The H-

metaphor inspired HAVEit to maintain the driver "meaningfully involved in the driving task". From the point of view of human-centered automation, HAVEit approach underscored the dynamic task repartition such that the driver can be relieved in overload and underload conditions. Moreover, a driver state assessment module was developed to evaluate if the driver is capable to take over the control in case of system's failure or limits.

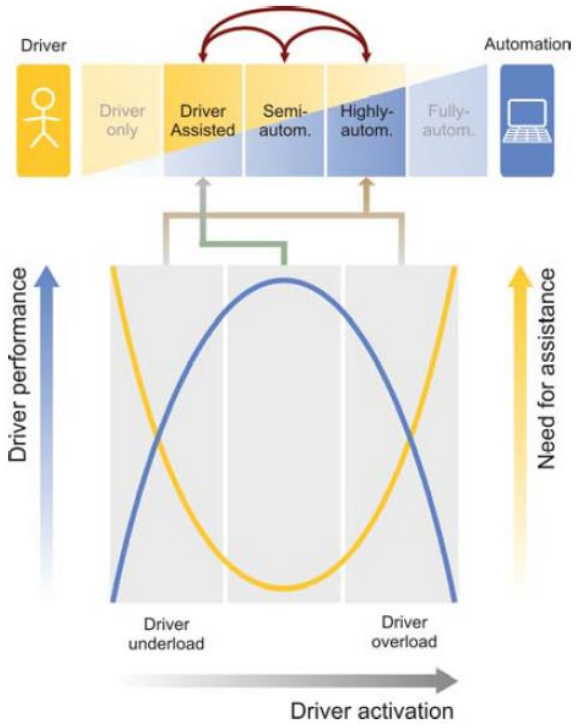


Figure 2.3 The automation spectrum in HAVEit and the dynamic task allocation (figure extracted from (Flemisch et al. 2010))

Glaser et al. (2010) designed a maneuver-based interaction concept for the co-system in HAVEit. A key component in this interaction concept is a maneuver grid that enumerates possible maneuvers in the context of highway driving. By visualizing this maneuver grid, the co-system can recommend maneuvers to the driver. The co-system can also evaluate the driver's maneuver intention with the help of this grid. The driver will be warned by HMI if his intention corresponds to a maneuver with high risk in this grid. At the control level, force feedback steering wheel and pedals enriched the HMI of the co-system. The pedals can be used for speed adaptation up to a certain pedal position when the automation is active. Following the principle that the driver has the final authority on driving, transition to manual driving mode will be triggered when the driver presses a pedal over this threshold. Meanwhile a vibration will be rendered on the pedal to inform the driver. In lateral dimension, the co-system will issue a lane change request in the HMI display if it detects a possible lane change maneuver. The

driver can validate it by turning on the corresponding indicator or turning in the corresponding direction.

The discrete levels of automation defined in HAVEit have influenced the taxonomies of levels of automation such as BASt (Gasser et al. 2009) and SAE definitions. Moreover, the impacts of legal safety on AD system design were discussed in details (Vanholme et al. 2013). An essential contribution of HAVEit to the safety of automated driving is the notion of "minimum-risk state (MRS)"[9] which was adopted in the taxonomy of SAE later.

2.5.5 Project ABV (Glaser 2013; Sentouh et al. 2014)

French project ABV (Acronym in French "Automatisation Basse Vitesse" meaning "Low speed automation", 2009-2012) focused on the use case of driving automation in a congested highway traffic (speed inferior to 50 km/h).

The driving automation system designed in ABV had three discrete levels of automation: manual driving mode (level 0), automatic longitudinal control (level 1), and ABV mode equivalent to the partial automation (level 2) in the taxonomy of SAE. The transitions between different levels of automation can either be initiated by the driver or the automation. Since the driver needs to supervise the system (level 2), a driver monitoring system was used to ensure that the driver was not drowsy and was aware of the situation. ABV contributed to AD system with multiple levels of automation by introducing the concept of "wake-up procedure". The wake-up procedure is a sub-state in each driving mode. If the driver monitoring system detects that the driver is distracted, the system will engage the wake-up procedure. The warning signal (visual and sound signals) will be issued through HMI. Particularly, if the AD system is in automatic longitudinal mode or ABV mode, the AD system will also reduce the vehicle speed, and enter a MRS if the driver does not react appropriately after a certain time.

Another contribution of ABV consisted in exploring shared control scheme in the ABV mode to manage the shared authority issue. Contrary to the conventional "override" mechanism through which the system suspends its control when the driver intervenes, the automated system shares lateral control with the driver in ABV mode if the latter steers.

[9] MRS refers to a state the driver or the AD system intends to reach in case the system is no longer capable of performing a certain level of automation. Whether the system can automatically achieve a MRS discriminates a level 4 system from a level 3 system.

Fig. 2.4 shows the shared control architecture of the AD system in ABV. Two levels of cooperation can be identified in this architecture. The first one is called low-level cooperation (LLC in Fig. 2.4). The low-level cooperation concerns the cooperation in action implemented by haptic shared control. High-level cooperation (HLC in Fig. 2.4) corresponds to planning cooperation level where the path for the controller to follow is determined by the state of the driver and the driving environment state.

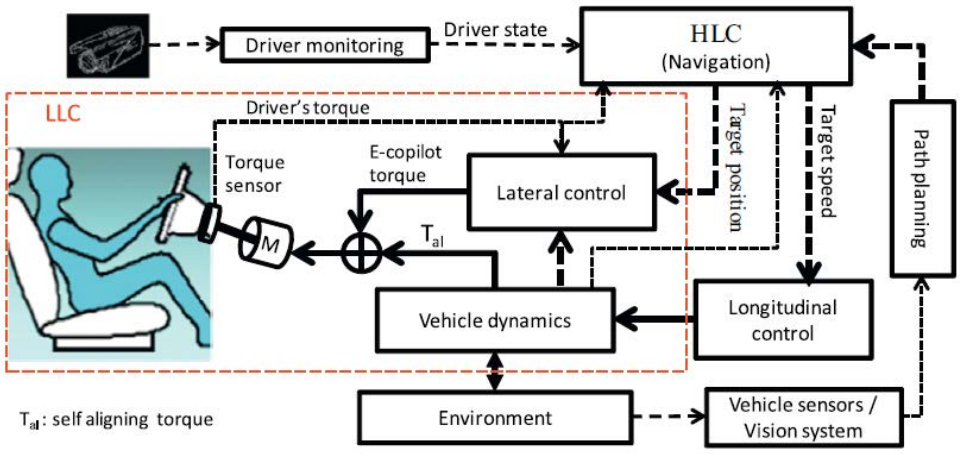


Figure 2.4 Shared control architecture implemented in ABV (figure extracted from (Sentouh et al. 2014))

2.5.6 Project INTERACTIVE (Alessandretti et al. 2014)

INTERACTIVE (2010-2013) focused on developing ADAS that provide active interventions for accident avoidance from early holistic preventive safety, to automatic collision avoidance, and to collision mitigation. Lio et al. (2015) developed a concept of artificial "co-driver". The "co-driver" designates an automated system that is able both to drive similarly like human driver and to infer human intentions. The co-driver has a layered architecture, so the complex driving activities can be decomposed into each layer. In order to imitate human driving behaviors, an internal loop consisting of inverse models and forward emulators was integrated in the architecture, as shown in Fig. 2.5. The forward loop (the inverse model) aims at reproducing human action plans. In this loop, optimal motion planning technique was employed to generate different motion hypotheses. Then the back loop, constituted by forward emulators is responsible for matching the motion hypotheses to human actions. Since the effects of human action on system future states are predicted in the emulation, "forward" was used in an extrapolation sense. The concept of co-driver contributed to solutions for maintaining a common of reference between the driver and the automation at the planning level (see Section 2.5.2).

In INTERACTIVE, the co-driver is instantiated by a warning system. The co-driver based its warning strategies on the estimation of driver intentions. Driver intention estimation was in turn realized by the internal loop of inverse models and forward emulators.

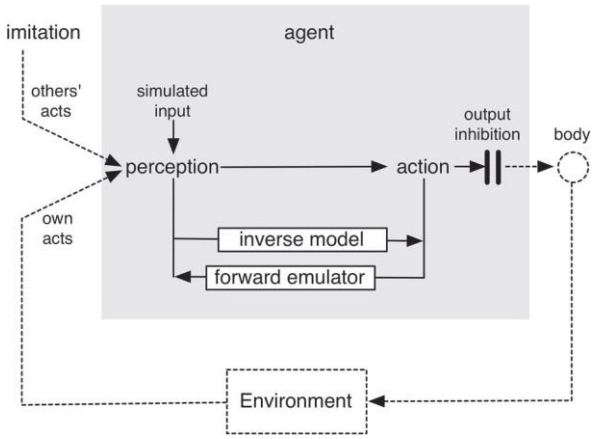


Figure 2.5 Architecture of the "co-driver" (figure extracted from (Lio et al. 2015))

2.6 Conclusion

This chapter has presented some theoretical concepts and methods in the literature which form a general theoretical framework in which we study and design driver-vehicle cooperation in this thesis.

Firstly, we presented the concept of human-machine cooperation with its origin and the definition of J.-M. Hoc. Human-machine cooperation was proposed to address task distribution within the human-machine system. This concept offers important design principles for us to manage the shared authority between the driver and the AD system in this thesis. These principles include viewing cooperation as an activity of interference management, the final authority definition and the role of shared situational awareness and shared goals to enhance the overall performance of the human-machine system.

Then we introduced the UCD approach which provides us a user's perspective to design driver-vehicle cooperation. This perspective highlights the role of the automation as a resource for facilitating human's interaction with the environment. The concepts of use cases and user test were implemented in our design work in Chapter 5 and 6.

In the next section, we presented shared control and human supervisory control schemes. By analysing some related works in the literature, we showed that both had their interests and limits to be applied for driver-vehicle cooperation. This remark motivated us to develop a hierarchical control framework which combine these two schemes and in which the shortcoming of each can be compensated by the other scheme. This effort will be presented in Chapter 3.

This chapter was closed with a review of previous works on driver-vehicle cooperation. In the next chapter, we will present how we derived cooperation principles within human-machine cooperation framework and mapped shared control and human supervisory control in a hierarchical control architecture for AD systems.

3 GENERAL ARCHITECTURE AND PRINCIPLES FOR DRIVER-VEHICLE COOPERATION

3.1 Introduction

It is well known that human cognitive processes are complex. AD systems, on the other hand, tend to be complex too. The complexity of AD systems is mainly due to the need to achieve different tasks simultaneously in a challenging dynamic environment. Thus, to design cooperation in this complex human-machine system is not trivial.

Hierarchical model is widely used to represent complex systems. Simon (1996) in his influential book "The sciences of the artificial" gave a synthesis of hierarchical models applied for social, biological, physical and symbolic systems. A system in this hierarchical representation is composed of interrelated subsystems, each of the latter being in turn hierarchical in structure until some lowest level of elementary subsystem are reached. The first strength of a hierarchical model is that it provides descriptions of a system at different levels of details. Moreover, Simon assumed that some hierarchical systems are *nearly decomposable*. This near decomposability is manifested by that linkages between subsystems at a level are stronger than linkages between subsystems across levels. Under this assumption, one can study behaviors of subsystems at a level while ignoring their interactions with subsystems at other levels. On the other hand, while studying interactions between the processes at different levels in a whole system, one can ignore the details of subsystems at a level.

Hierarchical models have been extensively used to describe human operator's behaviors in human-centered automation. For example, Rasmussen's model of human behavior (Rasmussen 1983) which decomposed human behavior into knowledge-based, rule-based and skill-based levels has widely been used as a frame of reference to design automated functions that support human's activity. As presented in Chapter 2, Hoc (2001) has decomposed the cooperative activity into three levels based on human's cognitive process. On the machine side, layered architectures, one type of hierarchical architecture, which originated from the discipline of robotics, are popular system architectures for autonomous vehicles. Given the strength and the popularity of hierarchical representation, we aim to decompose driver-vehicle cooperation into different levels to reduce the complexity of interaction design.

This decomposition is based on the common hierarchy between Michon's hierarchical model of the driving task and several planning and control systems used in autonomous vehicles in the state-of-the-art. Section 3.2 demonstrates this common hierarchy by presenting Michon's model and planning and control functions of driving automation systems. In Section 3.3, we analyze the potential interference between the system and the driver in the driving task at each level and propose a possible form of cooperation for each level. Based on these analyses, we derive a new cooperative control architecture by integrating new cooperative functions into a planning and control architecture of AD system. Finally, we propose two principles for driver-vehicle cooperation at the tactical level and the operational level respectively in Section 3.4. Each principle describes how the driver and the system share authority in the decomposed driving task.

3.2 A common hierarchy to describe driver behavior and functions for driving automation

Michon's hierarchical model of the driving task (1985) assumed human driver behavior as a hierarchically ordered structure of different behavior levels. In this section, we attempt to demonstrate that the functions of several automated systems (including AD systems) that automate partial or the entire driving task can be organized into the hierarchy of Michon's model. This reveals a kind of mapping between the assumed driver behavior levels and system functional levels of AD system. The mapping offers a possibility to design driver-vehicle cooperation at each common level. Along with this demonstration, we introduce a layered functional architecture of AD systems in which we present principal planning and control functions. This architecture serves as a basis for the hierarchical cooperative control architecture presented in the next section.

3.2.1 Point of departure: Michon's hierarchical model of the driving task

In Michon's model, the driving task was decomposed into the following three levels: *strategical, tactical and operational levels*. These three levels can be distinguished by the driving behavior and the time frame involved at each level. The general trip planning, including trip goal setting, route plan and modal choice takes place at the strategical level. General trip plan can be made in advance of a trip. Specific strategical decisions can generally be done many minutes before the execution during the trip. At the tactical level, the driver performs maneuver control to ensure a safe navigation in local driving environment during a trip. Decisions on maneuvers are considered to take place in seconds, and strongly depend on driver's situation awareness on the surrounding environment. The operational level consists of immediate vehicle control and the decisions on control actions require only milliseconds.

Moreover, interactions exist between the cognitive processes at different levels. Essentially, the processes at a higher level provide goals or impose constraints on the processes at a lower level. Conversely, the processes at the lower level ascend feedbacks on local task execution towards the higher level. The processes at the higher level can then adapt the goals to fit the outcomes from the lower level. For example, a driver who leaves for summer vacation may expect to drive for leisure on the route. His strategy could be "no rush", which could engender tactical decisions such as prioritizing cruising, avoiding overtaking maneuvers (strategical level). However, excessively slow traffic on the road may motivate the driver to accelerate, to pass vehicles (tactical level). Eventually, he may either plan another route to or adopt a more aggressive driving strategy (strategical level).

3.2.2 Control function of ACC

Before discussing the hierarchy of functions for automated driving based on the tri-hierarchy of Michon, it is of interest to examine driver assistance systems as an intermediated step. ACC is a typical driver assistance system that automates longitudinal vehicle motion control. At the tactical level, a standard ACC can realize two maneuvers in the longitudinal dimension: *cruising*—a maneuver that keeps a constant speed and *vehicle following*—a maneuver that maintains a safe time headway with a lead vehicle. The main task at the tactical level hence concerns which maneuver to be engaged. A basic mechanism is a "switch" strategy based on the target (lead vehicle in the path) detection information (Winner 2012). Bageshwar, Garrard, and Rajamani (2004) argued the existence of a so-called *transitional maneuver*. This transitional maneuver is responsible for achieving a smooth transition from a cruising maneuver to reach a steady vehicle following maneuver.

At the operational level, an ACC performs vehicle longitudinal control to realize cruising or following maneuver. From a control perspective, a maneuver needs to be translated to a reference for a controller. For cruising maneuver, the reference is a set speed; however, for vehicle following, different references have been proposed, including static references like the popular constant time headway policy (Ioannou and Chien 1993), and dynamic references accounting for multiple performance factors (Martinez and Canudas-de-Wit 2007). Different controllers can be used for speed control and distance control respectively (Rajamani 2006). An alternative consists of a single controller combined with a transient trajectory generation module (Kim 2012).

Based on the above discussion, we can see that control functions of ACC can be mapped into the hierarchy of driver behavior in a straightforward manner. The same is true for a driving automation system that performs lateral control, e.g., a lane keeping assistance system with active lane change assist function.

3.2.3 Planning and control functions of AD systems

The complex nature of an AD system has been analyzed and characterized in several works. Following the concept of a *rational agent* (Russell and Norvig 2009), an AD system can be decomposed into three subsystems at the first level: *perception*, *planning* (also referred to as decision-making) and *control* subsystems (Taş et al. 2016). In our design work, we focus on functions in planning and control subsystems.

The reports on the autonomous vehicles in DAPRA Urban Challenge serve as an ideal database of reference of system architectures (Buehler, Iagnemma, and Singh 2009). Several autonomous vehicles adopted similar tri-layered architectures for planning module with the decomposition consistent with that in Michon's model. Tab. 3.1 summarizes these vehicles with their planning functions mapped into the three levels of the driving task. As more recent examples, the PRORETA 3 project (Bauer et al. 2012) and PROUD project demonstrators (Broggi et al. 2015) and Audi's A7 concept car (Ulbrich and Maurer 2015) all adopted similar layered architectures.

Table 3.1 List of the vehicles in DARPA Urban Challenge with tri-layered planning architecture

Vehicles	Strategical level	Tactical level	Operational level
Boss (Urmson et al. 2009)	Mission planning	Behavior executive (FSM*)	Motion planning
Junior (Montemerlo et al. 2009)	Global path planner	FSM	Road navigator
Odin (Reinholtz et al. 2009)	Route planner	Driving behaviors (DAMN**)	Motion planner
Skynet (Miller et al. 2009)	Route planning	Tactical planner (State-based reasoning)	Path generation
AnnieWAY (Kammel et al. 2009)	Mission planning	Maneuver planning (FSM)	Collision avoidance
Knight Rider (Patz et al. 2009)	Mission planning	Core AI (FSM)	Path planner

*FSM: finite state machine; **DAMN: Distributed Architecture for Mobile Navigation

The control subsystem which controls vehicle actuators belongs to the operational level by its nature, thus we incorporate it into the lowest layer in the tri-layered architecture shared by the above-cited autonomous vehicles. This reorganized functional architecture with a unified name convention is illustrated in Fig. 3.1. Note that the interface with the *perception and situation assessment* subsystem is also partitioned into three abstraction levels to highlight the main information on the external environment that the function at each layer requires.

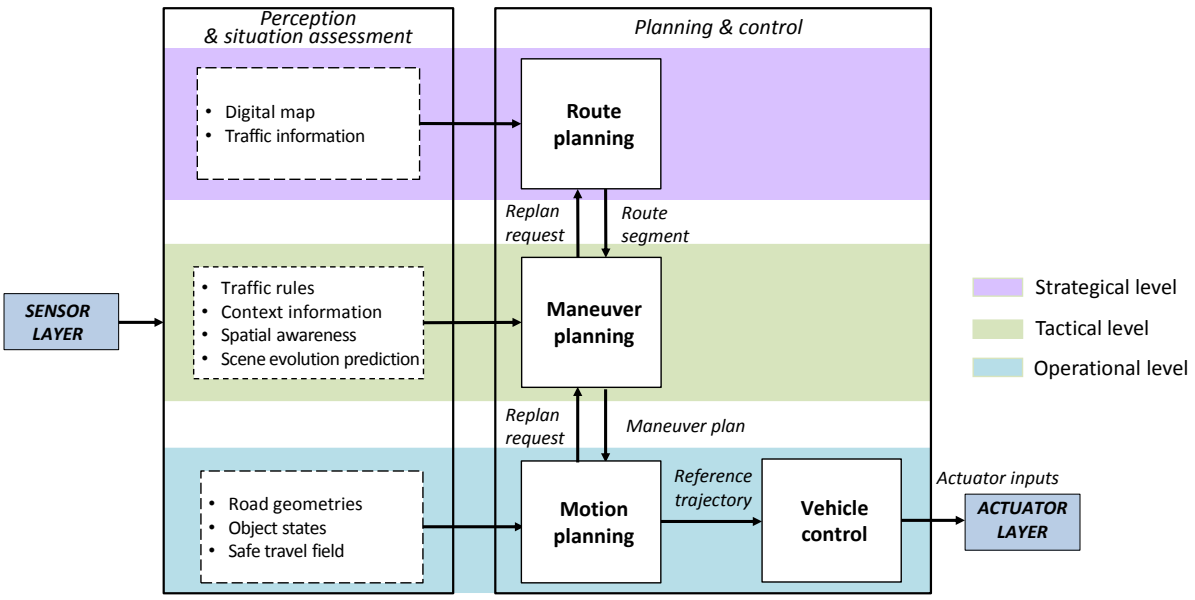


Figure 3.1 A typical functional architecture for AD systems adapted from the architectures cited in Tab. 3.1 (solid-border blocks for functions, while dash-border blocks for interfaces)

Like in-car navigation systems, the function *route planning* at the strategical level plans a route to the destination according to certain criteria such as time-saving, energy economy and available time for using the AD mode. The planned route is split and the actual route segment is descended to maneuver planning at the tactical level.

The function *maneuver planning* at the tactical level is responsible for generating maneuvers not only to follow the planned route from the strategical level but also to handle local situations, e.g., to overtake a slow vehicle or to yield at an interaction. The methods for maneuver planning used by the DARPA autonomous vehicles are indicated in parentheses in the column "Tactical level" of Tab. 3.1. They all follow the behavior-based paradigm originating in the field of robotics (Matarić and Michaud 2008). Behavior-based paradigm highlights the concept of "*situated*"[10], i.e., behaviors emerge from the interactions between the cognitive agent and the environment. In this sense, which maneuver to perform by the AD system strongly depends on the current driving situation. Consequently, the situation awareness, i.e., the understanding of the external environment plays a crucial role for behavior-based systems.

The approaches for maneuver planning in Tab. 3.1 can be broken down into two steps—*alternative generation* and *maneuver selection*. Alternative generation is responsible for generating one or multiple maneuver alternatives to cope with the current situation. In practice, basic maneuvers are designed *a priori* considering the use cases to be addressed and the goal to be attained by each behavior (referred to as *behavior synthesis,* Matarić and Michaud 2008). These basic maneuvers can be organized according to different driving situations such as intersection, parking lot and lane driving. The second step aims to generate a maneuver among multiple alternatives. This decision-making process is formulated as the *action-selection mechanism* (Pirjanian 1999). There are basically two classes of approaches for action selection—*arbitration* which selects one from multiple options and *command fusion* which combines different alternatives. State transitions in a finite state machine are a kind of arbitration, whereas DAMN (Distributed Architecture for Mobile Navigation) voting (Reinholtz et al. 2009) and fuzzy sets (Naranjo et al. 2008) are typical command fusion techniques which have been applied in automated driving.

The behavior-based paradigm has its inherent advantage for driver-vehicle cooperation, in that it emphasizes the role of the context. It is hence possible for both the driver and the automation

[10] A similar hypothesis of "situated action" was proposed by Lucy Suchman (1987) to explain human acitivity and cognitive process. This hypothesis has influenced interactive system design and practices involved in cognitive ergonomics.

to share maneuver plans within a same context. This is consistent with the design guideline of maintaining a *common frame of reference* that we drew from the concept of human-machine cooperation in Chapter 2. However, behavior-based maneuver planning approaches have the limits in managing unforeseen situations and ensuring maneuver feasibility for the lower level (Chen et al. 2009).

The driving task at the operational level is performed by two functions: *motion planning* and *vehicle control*. Motion planning is responsible for planning a trajectory or a path as a realization of a maneuver plan (usually as end vehicle poses or states) generated from the tactical level. A recent survey by Paden et al. (2016) reviewed motion planning algorithms for automated driving. Vehicle control function aims at controlling relevant vehicle actuators to track the reference trajectory while stabilizing the vehicle. Considering the complexity of vehicle dynamics, vehicle control is often separated into longitudinal and lateral dimensions. Various techniques based on the control theory are applied, covering from the simple PID control to advanced control techniques such as optimal, adaptive, predictive, fuzzy control.

3.3 Proposition of a hierarchical cooperative control architecture (Guo et al. 2015)

The above analysis demonstrates that planning and control functions for automated driving can be decomposed into the tri-level hierarchy of Michon's model. Within this common hierarchy, we discuss how the human driver could cooperate with the system's function at each level. Then we extend the planning and control architecture presented above to a new architecture with new functions added to realize the suggested cooperation at each level.

3.3.1 Decomposition of driver-vehicle cooperation

At the strategical level, the driving task is related to route planning. Possible cooperation forms can be inspired from the use of current in-car navigation systems. For example, the driver can select a preferred route before or during a trip. The system can also suggest new route alternatives to the driver during the trip, e.g., based on the front traffic information via vehicle to infrastructure communication. Driver-vehicle cooperation in route planning can be regarded as a form of arbitration on multiple route alternatives. In the tri-level hierarchy, route arbitration can also be influenced by the cooperation at lower levels. For instance, when the driver intervenes at the operational or tactical level to make the vehicle move towards a direction not planned by the system at a highway split, the system shall replan a route or ask the driver whether he wishes to change to anther destination. In addition, there are other "strategical"

issues in driver-vehicle cooperation at the highest level, e.g., transitions of system's operating mode (manual driving mode, automated driving mode, etc.).

At the tactical level, the driver and the system may have different maneuver plans to handle a situation. This interference in maneuver plans could be due to different value criteria of the driver and the system in their maneuver decision-making processes. In this hierarchy, two means of interaction can be exploited to manage this interference. The first one consists of offering the driver a mean to directly indicate his maneuver plan at the tactical level. With the second mean, the driver can directly control the vehicle at the operational level and the system infers and adapts to driver's maneuver intention at the tactical level. No matter which mean, how to select a maneuver plan between that of the driver and that of the system can be formulated as a *maneuver arbitration* problem.

At the operational level, the driver may interfere with the system regarding to vehicle's relative positions to the road (e.g., lateral deviation to lane center) or to other vehicles (e.g., gap to a lead vehicle). Therefore, he may have needs to act on the vehicle control interface (pedals and steering wheel) to subtly control vehicle's trajectory. Moreover, in some urgent situations, it may be more natural for a driver to directly react by turning steering wheel or pressing pedals. When the driver exerts control simultaneously with the AD system, how to determine the final control input can be formulated as a *shared control* problem (refer to Section 2.4.1).

3.3.2 A functional hierarchical architecture for cooperative control

At each level of the planning and control bloc in the system architecture (Fig. 3.1), we add a new function or adapt the original function to realize the cooperation forms proposed above. In this way, we derive a new architecture for planning and control function, which is referred to as *hierarchical cooperative control architecture* for AD systems, as shown in Fig. 3.2. The driver's commands or control enter in the architecture via a proper HMI at each level. At the strategical level, the route arbitration module arbitrates a route based on possible route alternatives and the driver's route choice and passes it to the tactical level. The tactical level cooperation takes place in the maneuver arbitration module. The arbitrated maneuver will be executed at the operational level. At the operational level, the cooperation takes the form of shared control. The control of the system and that of the driver together influence the vehicle's behavior through the actuator layer. Without the driver's intervention, the system's own decision and control at each level are passed directly to its subordinate level, i.e., the system operates in the same way as in a classical planning and control architecture shown in Fig. 3.1.

This architecture offers a generic description of system functions and function interactions involved in driver-vehicle cooperation. The process of cooperation, i.e., how the AD system and the driver share the authority on each task is described by a cooperation principle. The next section will present the proposed cooperation principles.

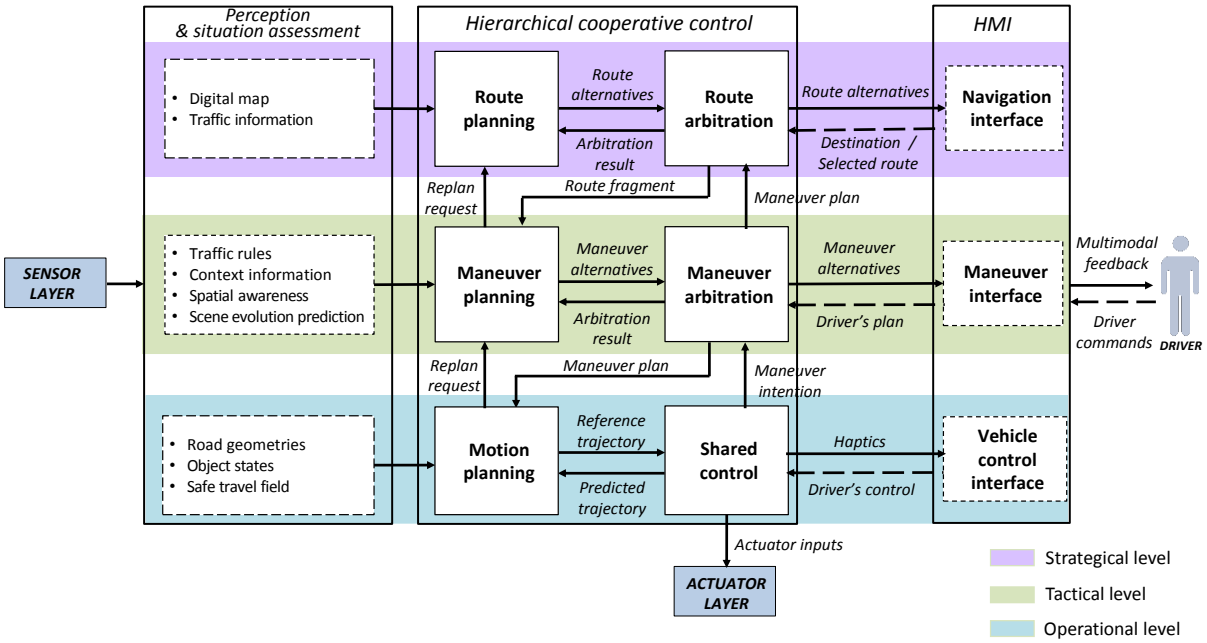


Figure 3.2 Architecture of the hierarchical cooperative control within an AD system (solid-border blocks for functions, while dash-border blocks for interfaces)

3.4 Proposition of cooperation principles

The cooperation in the route planning is not in the scope of this thesis. We propose cooperation principles for tactical and operational levels respectively. Each principle describes in which way system's functions share the authority with the driver to manage the interference at this level.

3.4.1 Principle for maneuver cooperation

The central design issue at the tactical level is to manage the interference between the driver and the system on a maneuver plan. As discussed above, one possible cooperation method is to offer the driver a mean to indicate his maneuver plan, and the cooperation takes form of arbitration. Considering that the driver does not need to continuously monitor the environment and the system in the AD mode, the cooperation can be initiated by the system, i.e., the system invites the driver to participate into the cooperation. From a user's perspective, showing system's intended maneuver could be beneficial for the driver in terms of increasing his situation awareness. Based on these considerations, we intend to explore the "situated" nature

of the behavior-based paradigm used in system's maneuver planning. When the AD system encounters a new situation, it generates feasible alternatives. Instead of arbitrating between these alternatives in a full autonomous manner, the system shows its intended maneuver as well as other alternatives (if there are) to the driver. The driver can select other plausible alternatives. Then the system adopts the driver's selection as its maneuver plan. This principle for maneuver cooperation can be briefly stated as follows:

The system shows its intended maneuver plan and plausible alternatives, while the driver can select an alternative if he is not satisfied with system's plan.

To implement this principle first requires the system to understand the current situation. Moreover, the system needs to predict situation's evolution to present his intention and alternatives in advance, the driver can hence have sufficient time to give his choice. Both requirements highlight the role of "situation awareness" of the system which will be a focus in Chapter 4.

3.4.2 Principle for control cooperation

In the hierarchical cooperative control architecture, driver-vehicle cooperation at the operational level is formulated as a shared control problem. As introduced in Section 2.4.1, there are two classes of shared control schemes—blended shared control and haptic shared control. Haptic shared control has an appealing advantage of incorporating both control allocation and activity communication into a unified framework (Abbink, Mulder, and Boer 2012). Moreover, haptic shared control can be implemented through a conventional actuator infrastructure, contrary to blended shared control which usually requires a costly by-wire system. For these two reasons, we adopted haptic shared control scheme in the cooperation design.

Traditionally, shared control addresses the case where the human operator always remains in the control-loop and continuously interacts with the automation (Abbink, Mulder, and Boer 2012). For AD applications where the driver for the most of time is not in the control-loop in the AD mode, we consider shared control as a mode similar as an override mode, i.e., the system adapts its control once the driver's action is detected in the loop. The advantage of overriding is that the driver has the maximum of control freedom. Shared control, on the other hand, has the potential of reducing the driver's workload and enhancing the driver's performance especially in difficult tasks. We intend to propose a cooperation principle that draws advantages from both modes. This principle for control cooperation is stated as follows:

The driver can quickly and easily regain the control authority when he deems necessary or desirable; the system supports the driver if it detects the driver's intention and renders the resistance if his action could undermine the safety.

The first part of this principle aims to offer an override mode to the driver to meet potential user needs, e.g., to temporally correct vehicle trajectories or to adjust the vehicle's speed with ease. The second part takes advantage of shared control scheme. Instead of suspending its control like in a classical override mode, the system actively shares the control authority in two aspects. In the first aspect, the system can assist the driver by predicting his maneuver intention. In this sense, the cooperation at the operational level triggers the cooperative activity at the tactical level. Thus, the driver could use traditional vehicle control interface as a maneuver command interface. Another aspect of shared control in this principle is that the system can exert the control to resist the driver if his action could cause a hazard on the driving safety. This resistance is essentially used to warn the driver of potential hazards, e.g., collision risk. This warning could be useful, especially considering that the driver's situation awareness could be decreased in the AD mode, as suggested in several human factors studies (Blanco et al. 2015). Moreover, haptic feedback can be directly processed in the sensorimotor loop of a human, thus it is more efficient in terms of communication than symbolic representation which usually requires the driver's further interpretation (Hoc, Young, and Blosseville 2009). The similar concepts inspired from the flight envelope control of the aviation domain (Billings 1997) have been implemented in the shared control for driving assistance systems in the work of Itoh and Inagaki (2014) and that of Erlien et al. (2016). Whilst the former work discussed a "hard protection" mode (the driver's control can be overridden by the system via by-wire system) and a "soft protection" mode (haptic resistance) in a lane-changing collision avoidance scenario, the latter designed a by-wire active steering system that can override the driver's control if the system predicts that the vehicle trajectory will violate a safety envelope of vehicle states.

3.4.3 A brief discussion on the final authority issue

The principle for maneuver cooperation deals with the shared authority on maneuver decision-making. In this principle, the system still holds the full control authority. This implies that the alternative selected by the driver could be abandoned by the system at the control level in highly dynamic situations.

At the operational level, the driver and the system shares the control authority. In contrast to maneuver cooperation, the driver's control immediately influences the dynamics of the vehicle. In the proposed principle, the system can resist the driver to prevent potential danger. The driver

and the system may hence enter in conflict situations and a fundamental question needs to be answered: who has the final authority?

The problem of the final authority perhaps is one of the most difficult in the human-automation interaction design. This question has already been raised in the field of aviation long before. Airbus and Boeing adopt different design philosophies regarding to this issue[11]. In the field of automobile driving, the environment is more dynamic than the flight environment and not all the human drivers are as skilful as well-trained pilots faced with the tasks they need to perform. In emergency situations, vehicle active safety systems such as ESC (electronic stability control) and AEBS (advanced emergency braking system) intervene automatically to avoid accidents (loss of stability or front collision). The driver cannot override these systems. However, the nature of these intervention systems is quite different with that of AD systems, because the former intervene only in close-accident scenarios (assuming that these situations are beyond the control of the driver, therefore it is not against the requirements of Vienna's convention), while the latter continuously performs automatic control, or using the term proposed by SAE, *on a sustained basis* (SAE 2016). Under the current legal regulations, the driver has the final authority and can always override low-level driving automation systems which performs sustained automatic control, e.g., ACC. A level 3 AD system still relies on the driver as a fall-back in case that it cannot handle the current situation. This implies that the driver still needs to guard the system under certain conditions. In this sense, it seems that the driver should have the final authority. However, the questions like how a level 3 AD system reacts to close-accident situations or whether those intervention systems (AEBS) shall be a functionality of the AD system or they shall operate separately have not yet conclusive answers. If it is technically not yet ready to answer the final authority issue, human factors deliberation is also needed. Interested readers are referred to the works of Young, Stanton, and Harris (2007) and Flemisch et al. (2012) which discussed the final authority issue from the perspective of cognitive ergonomics.

[11] The autopilot system in several series aircraft of Airbus incorporates a safe envelope as "hard" constraints which the driver cannot override. This means that the pilot can acquire the control authority limited within this envelope when the system is active, but the final authority is retained by the system. The pilot can exceed this envelope only by turning off portions of flight control. Boeing, on the other hand, gives the pilot the final authority in flight control. Boeing's system renders haptic feedback on the yoke to inform the pilot of the envelope limits, but the driver can always override the system by exerting more forces. (Billings 1997)

Given the complexity of this question, we make the following assumptions: the principle for control cooperation between the AD system and the driver is proposed for normal driving conditions, i.e., the resistance rendered by the system aims to alarm the driver and to prevent potential danger. Therefore, the driver can always override the system. In case of the immediate collision danger, we assume that a collision avoidance system like the system in (Brännström, Coelingh, and Sjöberg 2013) will intervene to avoid or mitigate the collision.

3.5 Conclusion

In this chapter, we identified a common hierarchy between a classical model of the driving task and a typical layered architecture of AD system. Based on this hierarchy, we proposed a form of cooperation at each level and elaborated a new hierarchical cooperative control architecture. At the tactical level, we proposed a principle that exploits the behavior-based paradigm to enable the cooperation on maneuver plans, while another principle at the operational level describes how the driver and the system share control authority in the haptic shared control scheme. The general architecture and cooperation principles are implemented in two use cases presented in Chapter 5 and 6 respectively.

As we highlighted in Section 3.4.1, the "situation awareness" of AD system, i.e., the understanding of the current situation and the anticipation of future events, plays a pivotal role for the maneuver cooperation. The problem on the creation of the "situation awareness" will be addressed in the next Chapter.

4 VEHICLE SITUATION AWARENESS FOR DRIVER-VEHICLE COOPERATION

4.1 Introduction

AD vehicles operate in a *dynamic*, *partially observable*, and sometimes *hostile* environment. Understanding such a difficult environment is crucial for an AD vehicle to generate safe and robust driving behaviors. As argued in Chapter 2 and 3, a shared situation representation aids the driver and the system to understand the goals and plans of each other during the cooperation.

The problem of human's awareness and understanding of what is happening in the vicinity is formulated as the problem of *situation awareness* (SA) in the field of human factors. Endsley defined SA as follows:

> *Situation awareness is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future. (Endsley 1988)*

In this definition, she distinguished three levels of SA: Level 1 SA—perceiving critical features from in the environment, Level 2 SA—understanding the meaning of those features and Level 3 SA—anticipating what events will happen in the near future. While the SA at each time instant can be considered as a "state of knowledge", the process to create the situational awareness is called *situation assessment*.

The similar concept of situation assessment has also been applied in the sensor data processing in intelligent vehicles, e.g., Polychronopoulos et al. defined "situation and threat assessment" as:

> *STA [Situation and Threat Assessment] establishes a view of activities, manoeuvres, locations and properties of moving and stationary obstacles and from it estimates what is happening or going to happen and the severity which events will occur. (Polychronopoulos et al. 2004)*

The analogy between the definition of Endsley and that of Polychronopoulos is obvious. Papp (2012, 61–80) directly borrowed the definition of Endsley to describe the equivalent functionality of intelligent vehicles. In an AD system, the perception module is responsible for creating Level 1 SA through extracting the features through raw sensor data and tracking objects; the creation of Level 2 SA corresponds to the creation of an internal representation as the system understands the external world: the creation of Level 3 SA is equivalent to the prediction of futures states of the dynamic objects in this representation.

In this Chapter, we deal with Level 2 and 3 SA of AD systems which are needed by the planning and control functions involved in driver-vehicle cooperation. Considering the complexity of situation understanding and trajectory prediction, we simplify these two problems based on the use cases treated in this thesis. The problem of creating SA for the AD system in this chapter is hence formulated as developing a situation assessment function including the following two modules:

1) *Scene representation* (Section 4.3): it is responsible for integrating independently perceived features into an associative context. It facilitates the system to interpret sensory data and to further analyse the situation. It corresponds to Level 2 SA.
2) *Trajectory prediction* (Section 4.4): the prediction of the scene evolution (Level 3 SA) is simplified as the prediction of the trajectory for a target vehicle. The predicted trajectory is used by the maneuver planning function in Chapter 5.

Fig. 4.1 shows the internal structure of situation assessment and its relations with the cooperative control frameworks developed in Chapter 5 and 6.

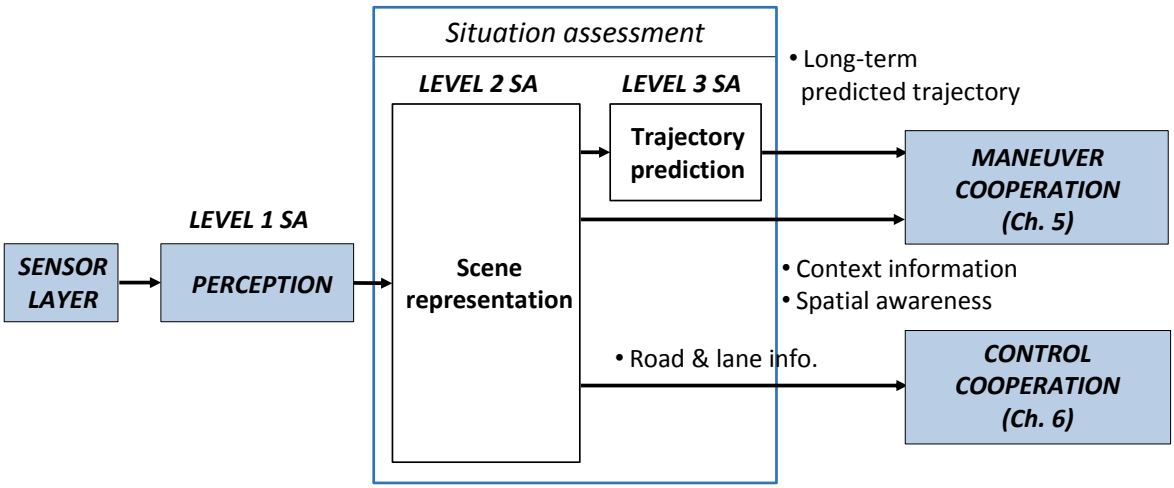


Figure 4.1 The framework of situation assessment

4.2 Assumptions on the perception function

The environment perception (Level 1 SA) is not in the scope of this thesis, therefore the following assumptions are made on system's perception function.

It is assumed that the AD vehicle contains a sensor system that has a coverage with 150m in front and 50m behind. In the lateral dimension, we assume that all the lanes in the carriageway of the AD vehicle can be covered. This detection zone illustrated in Fig. 4.2 can be achieved by combining different types of sensors like radar, Lidar, and camera.

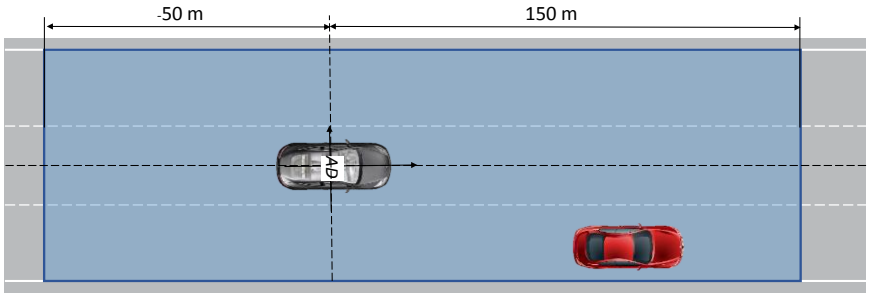


Figure 4.2 Assumed sensor coverage (constrained by highway bordure)

It is further assumed that a sensor data fusion layer is available. The output of this fusion layer is a list of the detected traffic vehicles. The state vector $\mathbf{x}^{(i)}$ of a traffic vehicle i is defined as

$$\mathbf{x}^{(i)} = [x \quad y \quad \psi \quad v_x \quad a_x \quad \dot{\psi}\,]^{\mathrm{T}}, \tag{4.1}$$

where (x, y, ψ) is the vehicle pose in a fixed earth frame Λ^E , v_x and a_x are longitudinal velocity and acceleration in the vehicle-fixed frame Λ^V, and $\dot{\psi}$ yaw rate (see Fig. 4.3). The uncertainty of the traffic vehicle states is considered in the trajectory prediction.

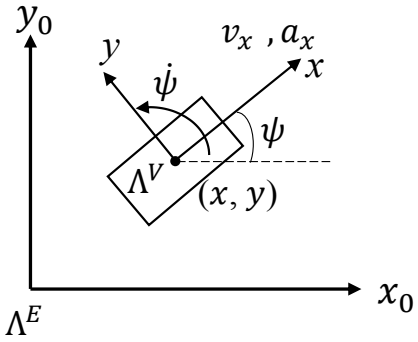


Figure 4.3 Simplified bicycle model for vehicle tracking

Finally, it is assumed that a localization system and vehicle motion sensors (e.g., odometer, velocity encoder, and inertial measurement unit (IMU)) provide the state vector of the ego vehicle[12] $\mathbf{x}^{ego}$ which is defined as

$$\mathbf{x}^{ego} = \begin{bmatrix} x & y & \psi & v_x & v_y & a_x & a_y & \dot{\psi} \end{bmatrix}^{\mathrm{T}}, \tag{4.2}$$

with v_y and a_y for lateral velocity and acceleration in the vehicle-fixed frame of the ego vehicle.

4.3 Highway driving scene representation

4.3.1 Principles of scene representation

A low-level scene representation focuses on the locations of detected objects and the geometric structure of the environment. It primarily consists of metric data. A typical example is the map-based representation which can be considered as the result of the mapping from the measurement space (sensor measurements) to the state space of the objects in the environment (Thrun, Burgard, and Fox 2005, 123–24). There are principally two ways of indexing the objects in a map, known as *feature-based* and *location-based*. In a feature-based map, each element of the map specifies the properties and location of one object, whereas the index of each element in a location-based map corresponds to a location and the value is the property of that specific coordinate. Fig. 4.4 shows two examples of these two kinds of maps.

[12] In this report, we use the term "ego vehicle" when we need to take an ego-centric perspective, e.g., when we mention the states and parameters of the vehicle which is equipped with the system of our interest.

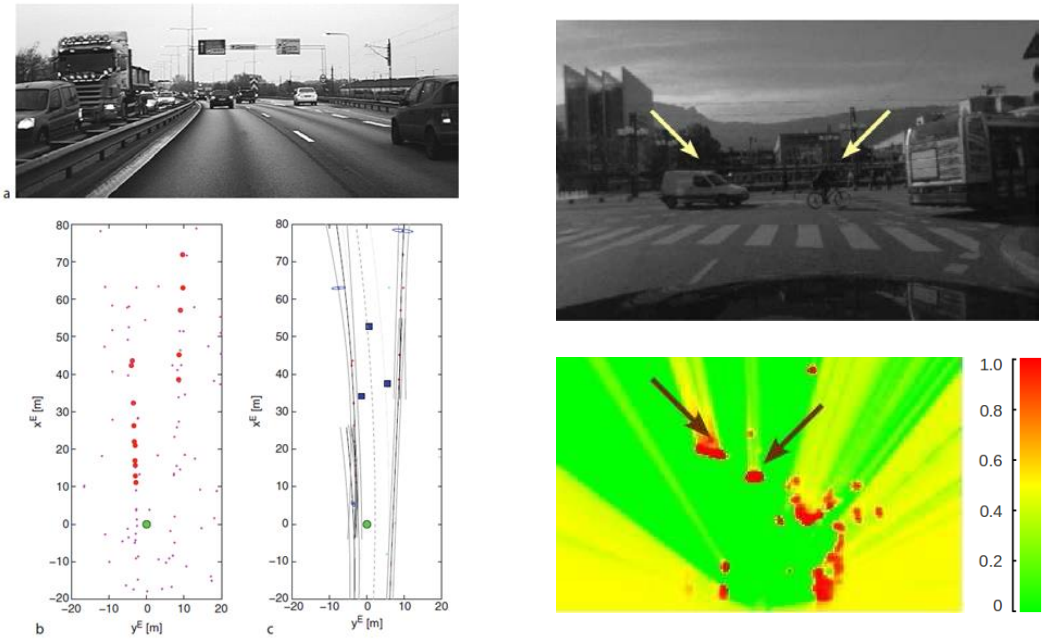


Figure 4.4 Two examples of map-based representation for intelligent vehicles. *Left*: a feature-based map specifying different features (the ego vehicle (green point), tracked traffic vehicles (blue rectangles) and road edge (solid lines)), extracted from (Lundquist, Schön, and Gustafsson 2012, 365–96); *Right*: a location-based map (occupancy-grid map) in which each coordinate is associated with the probability of occupancy, extracted from (Laugier et al. 2011).

A high-level scene representation handles the modelling of the driving context so that the system can make high-level decisions (in a sense of reasoning). It contains much richer information than a low-level scene representation. For example, Ulbrich et al. (2015) defined the components of a scene as illustrated in Fig. 4.5. The *dynamic elements* entail the states and attributes of dynamic objects in a scene (e.g., traffic vehicles and pedestrians). The *scenery* subsumes all geo-spatially stationary aspects of the scene, essentially the information on road network, e.g., topological information, lanes and lane markings. Digital maps constitute a solution for the representation of scenery. They store *a priori* information on the environments and help the AD system acquire the contextual information on a real driving situation (Ibañez-Guzmán et al. 2012, 1281). Fig. 4.6 shows an example to illustrate the importance of the knowledge on road network for understanding the situation. If the AD vehicle assesses the situation merely relying on the states of the red vehicle, e.g., relative positions and velocity, scene (a) and (b) are the same. However, if the AD system knows that the lane of the red vehicle will merge, it can be aware of the higher criticality of scene (a). A *self-representation* primarily contains the states and attributes of the ego vehicle. Finally, the scene representation is completed by a *semantic representation* layer which provides semantic descriptions for the entities on the scene, e.g., object classes and relationships among entities. Semantic representation is a formalism of knowledge representation which enables the system to infer

new knowledge (Sowa 1999). A common method to represent semantic information is ontology (Nardi and Brachman 2003, 16–20).

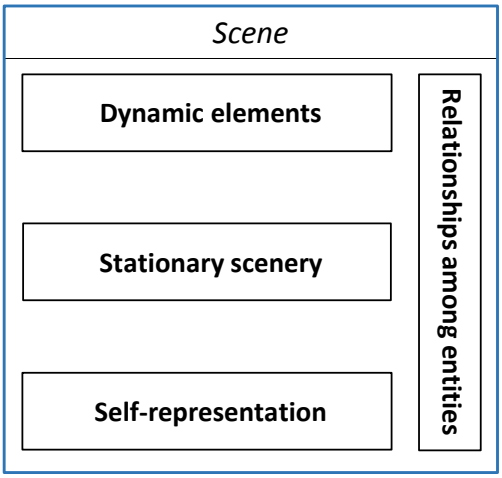
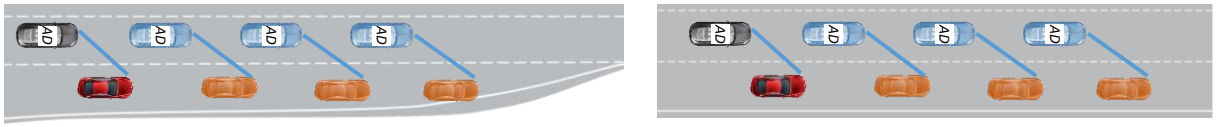


Figure 4.5 Components of a scene defined by Ulbrich et al. (2015)

Examples of formalized models used for scene representation consist of the "world model" in the 4D/RCS architecture for unmanned vehicle systems (Albus et al. 2002) and the "Local Dynamic Map" in the project SAFESPOT (Papp, Brown, and Bartels 2008) which was constructed through object oriented modelling approach.



(a) (b)

Figure 4.6 Illustration of two highway driving scenes sharing the same actual configuration

Constructing a formal model for scene representation is a very challenging task, because it requires a carefully designed data structure and complex processes to maintain and update the heterogenous data from different sources. Here we aim to design a simplified model which can provide sufficient information for planning and control functions to handle situations in our defined use cases. In this model, we address the following components of scene representation:

- *Dynamic entities:* we represent vehicle states in the Frenet frame to decouple longitudinal and lateral motions;
- *Scenery*: we use a digital map to provide necessary road and lane information for coordinate transformation, planning and control functions;
- *Semantic representation*: we propose a qualitative representation to describe the spatial relations of traffic vehicles to the ego vehicle.

4.3.2 Map use in scene representation

High-precision digital maps with high-precision localization solutions have already been exploited for vehicle guidance and control in automated vehicles (Ardelt, Coester, and Kaempchen 2012; Ziegler et al. 2014). Given that the designed cooperative systems in this thesis are implemented in driving simulation environment, we simulate a digital map based on RoadXML, a road network format for driving simulation use (Chaplier et al. 2012). This map mainly provides the following information:

- Road geometries for coordinate transformation and for vehicle lateral control (Chapter 6),
- Lane information for the lane-assignment of the detected vehicles,
- Road connectivity information (highway entry) for maneuver planning (Chapter 5)

RoadXML contains much richer information than that is needed for our use. In order to speed up data processing, we propose a simplified format for highway road network. Fig. 4.7 shows the hierarchy of the proposed map format. In this format, a highway network is composed of *tracks*. Each track corresponds to a mainline roadway or a ramp (with different identifiers). A track has two attributes—*road geometry* and *lane segment*. The road geometry of a track is defined by *curves* and *shape points*. The curves of a track are a series of straight lines, circle arcs, and clothoids which obey the same curvilinear equations, with null, constant, and linearly variable curvatures respectively. As clothoids cannot be evaluated in closed form, they are pre-sampled and the sampled points are stored in *shape points* of a track. The shape points of a clothoid will be used in the coordinate transformation in Section 4.3.3. A lane segment contains the information on lane numbers, lane widths, and the types of lane marks.

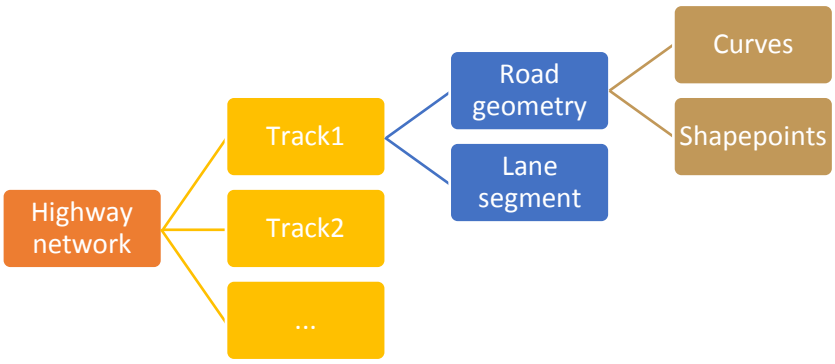


Figure 4.7 Hierarchy of the proposed map format for highway road network

The connectivity of the tracks follows the typology of a highway road network, i.e., ramps join a highway mainline via *highway entrance terminals* (also called *highway entry*) or *exit terminals* (AASHTO 2010). Fig. 4.8 illustrates a highway entry pattern. The track1 and track 2

are linked by the last node of track2 denoted by E1 which signifies the first point to enter the mainline. The ID of this node is associated with the curve in track1 which the track2 joins. By querying the next mainline curve that is connected with a ramp (associated with a ramp's end node), the AD system is able to calculate the distance to the next entry or exit.

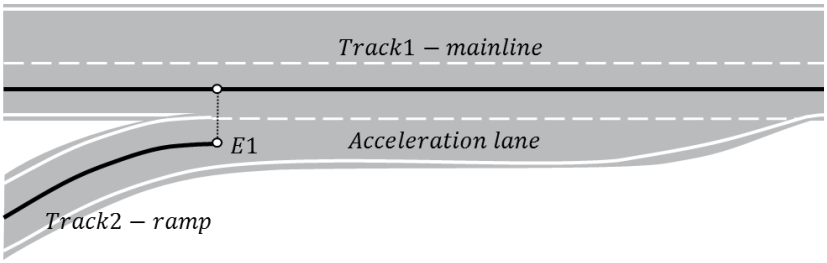


Figure 4.8 A highway entry section to illustrate the connectivity of a mainline track and a ramp track

This format is inherited from RoadXML, i.e., each attribute in this format has its equivalence in RoadXML. Therefore, it is possible to utilize a XML parser to extract features from a map of RoadXML. One result is shown in Fig. 4.9.

We implemented a "point-to-curve" map-matching algorithm in the literature to localize the AD vehicle in the map[13]. Once the AD vehicle is matched with a curve, a local map within the same ranges defined by the coverage of sensors is extracted. The state transformations for the detected traffic vehicles are then performed within this local map.

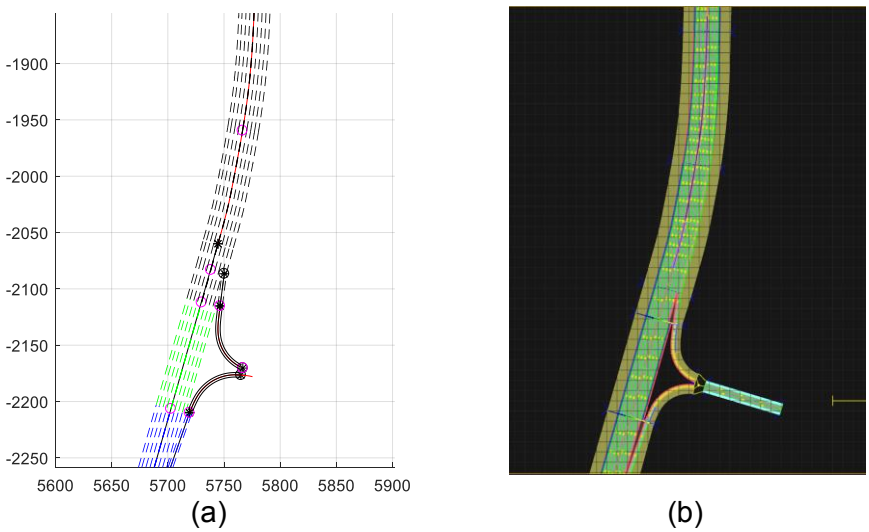


Figure 4.9 Visualization of a same map in the proposed map format (a) and RoadXML (b)

[13] This algorithm is described and evaluated by White, Bernstein, and Kornhauser (2000). It corresponds to the Algorithm 3 in their paper. The main principle of this algorithm is to match the AD vehicle's pose to the closest curve by searching the entire network on the first run of the algorithm. For the following runs, the algorithm only queries those arcs which connect the current curve.

4.3.3 State transformation into the Frenet frame

The *Frenet frame* (or Frenet-Serret Frame) is a moving reference frame which is originally used for curve analysis (Willmore 2012, 14). In $\mathbb{R}^2$, the basis of a Frenet frame is composed of a unit tangent vector $\mathbf{t}(s)$ and a unit normal vector $\mathbf{n}(s)$ of a point of a curve $\mathbf{r}(s)$ parameterized by its arc length s, as shown in Fig. 10. The curvature $k(s)$ of the $\mathbf{r}(s)$ is defined by

$$k(s) = \frac{d\psi(s)}{ds}, \tag{4.3}$$

where $\psi(s)$ is the angle between the tangent and the *x*-axis. $k(s)$ is a key factor in the Frenet frame, because it characterizes the relation between $\mathbf{t}(s)$ and $\mathbf{n}(s)$ through the famous formulas of Frenet

$$\begin{aligned} \frac{d\mathbf{t}}{ds} &= k\mathbf{n} \\ \frac{d\mathbf{n}}{ds} &= -k\mathbf{t} \end{aligned}. \tag{4.4}$$

The coordinates of a point in a Frenet frame are given by (s, d) with *s* for arc length and d the offset in the direction of $\mathbf{n}(s)$. We use $\mathbf{x}_{2D}$ to denote a 2D vector for the coordinates of a point in the Cartesian frame. The relation between the Cartesian coordinates (x, y) and the Frenet coordinates (s, d) is described by

$$\mathbf{x}_{2D}(s, d) = \mathbf{r}(s) + d\mathbf{n}(s). \tag{4.5}$$

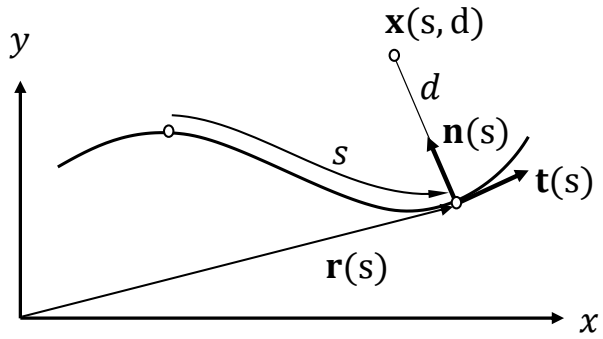


Figure 4.10 The Frenet frame on a curve

In the application for vehicle trajectory control, a Frenet frame is attached to a road curve, e.g., a lane centreline. In this case, the Frenet frame is called *road (fixed) coordinate system*. For example, Λ^{RD} in Fig. 4.11 is a road coordinate system. We define the state vector of a detected object in Λ^{RD} as

$$\mathbf{s} = [s \quad \dot{s} \quad \ddot{s} \quad d \quad \dot{d} \quad \ddot{d}]^{\mathrm{T}}. \tag{4.6}$$

The transformation from **x** to **s** is given in Appendix 1. By representing vehicle states in the Frenet frame, the movement of a vehicle is decoupled into a tangent component $[s, \dot{s}, \ddot{s}]$ and a normal component $[d, \dot{d}, \ddot{d}]$. This decoupling facilitates vehicle motion planning and control, which is a main reason for the popularity of the Frenet frame in vehicle planning and control applications.

The origin of the Frenet frame $O(s^{ego})$ takes the projected ego vehicle's position on the road curve. In this way, $s^{(i)}$ of an object *i* is directly a relative longitudinal position regarding to the ego vehicle. Note that the position of $O(s^{ego})$ is obtained during the map matching with an updating frequency of 20Hz.

4.3.4 Qualitative mapping

Qualitative mapping aims to generate a concise qualitative representation of the traffic scene. Spatial representation uses *discrete quantity spaces* (they can be understood as discrete classes, e.g., *front*, *back*, *left*, *right* and *close*, *far*) to describe orientation and distance information of an object in a 2D space (Clementini, Felice, and Hernández 1997). We propose a spatial representation to describe the relative positions of traffic vehicles with respect to the ego vehicle. The final representation form is similar with the matrix-form for the organization of the detected vehicles proposed by Bartels et al. (2009).

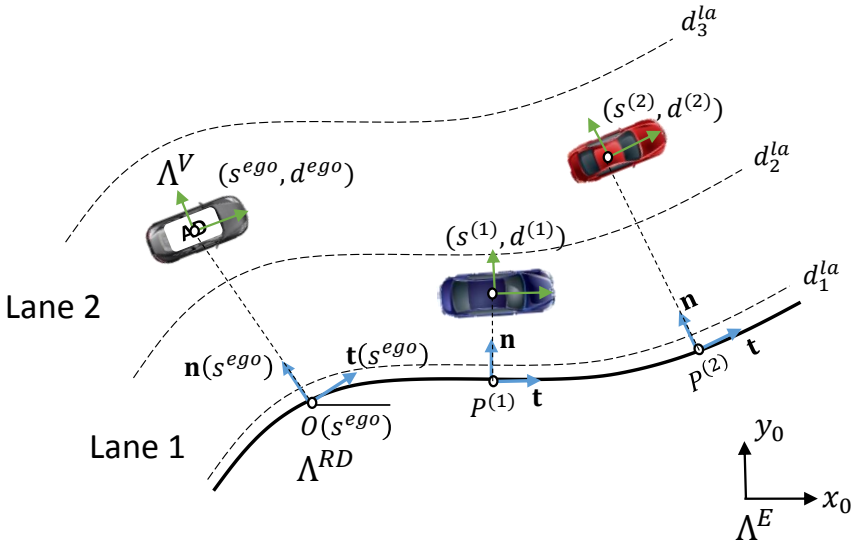


Figure 4.11 Coordinates of on-road vehicles in a road coordinate system

Fig. 4.12 illustrates the proposed spatial representation. This representation adopts an egocentric point of view and uses Λ^{RD} as the frame of reference. It contains 16 position classes which form a 4×4 matrix. Lateral relations (in the direction of *d*-axis) are partitioned by lanes. Therefore, the qualitative mapping in the lateral dimension is equivalent to lane assignment.

Since the *d* coordinates of lane boundaries can be extracted from the attribute "lane segment" in the digital map, the lane assignment process is trivial. Note that we set an on-ramp lane class to map the merging vehicles so that the AD vehicle can be aware of the highway merging context.

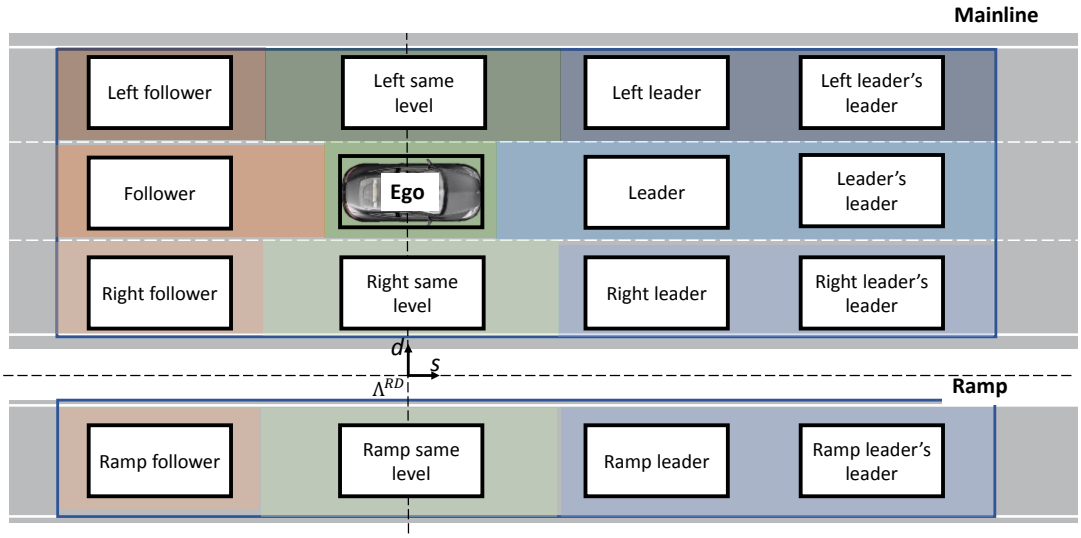


Figure 4.12 Highway qualitative spatial representation

Longitudinal relations are organized into four classes {*leader's leader, leader, same level, follower*}. Qualitative mapping of vehicles in the longitudinal dimension is based ordering their *s* coordinates. A *same level* vehicle is discriminated from a *leader* and *follower* if its s coordinate falls in the interval [-5, 5] m.

By mapping tracked vehicles into the cases of this matrix, the AD system can focus on different vehicles in decision-making or control process. For instance, if the system intends to perform a lane change maneuver, it can directly query *same level* and *follower* cases in the target adjacent lane to check the feasibility of a lane change. When the AD vehicle is passing a highway entry, it is of interest for the AD system to query the cases on ramp to get the information on those "*on-ramp* vehicles" which may merge in front.

4.4 Adaptive vehicle trajectory prediction (Guo et al. 2016)

4.4.1 Definition of the trajectory prediction

Predicting how the current situation evolves in the future allows the AD vehicle to reason and act in advance. It is beneficial for the driving safety (e.g., collision prediction) and the riding comfort (e.g., soft breaking in advance compared with the last moment hard breaking). In a dynamic environment, the prediction task is to compute future states of dynamic objects based

on their current states. The predicted future states are called *trajectory*. In the deterministic framework, a predicted trajectory can be expressed by

$$\mathbf{X} = \{\mathbf{x}(t) | t \in \mathcal{T}\}, \tag{4.7}$$

where $\mathcal{T}$ is the prediction horizon

$$\mathcal{T} = [t_0, t_H], \tag{4.8}$$

with t_0 for the *epoch time* at which the state is defined[14] and t_H for the end time. In the probabilistic framework, since exact knowledge of object's state is not available, $\mathbf{x}(t)$ is replaced by its posterior distribution $P(\mathbf{x}(t)|\mathbf{x}(t_0))$. Trajectory prediction can also be made in the discrete framework. In this case, $\mathcal{T}$ is discretized by N time instants and the state at a time instant $k \in [0, N]$ is predicted.

4.4.2 Related works

Trajectory prediction relies on a *motion model* that represents the motion of a dynamic object. Based on the assumption on model's structure and parameters, trajectory prediction approaches can be grouped in two categories: *parametric* and *nonparametric* approaches (Lefèvre et al. 2014).

4.4.2.1 Parametric approaches

Parametric approaches use *parametric* motion models, i.e., the models depend on their internal parameters. Parametric motion models are usually defined by a set of linear or nonlinear first-order differential state equations in the form of

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}, \mathbf{u}, \mathbf{w}, t), \quad t \in \mathcal{T}, \tag{4.9}$$

where $\mathbf{u}$ is control input, $\mathbf{w}$ is process noise characterising the uncertainty on the model. Once the model is determined, any state in $\mathcal{T}$ can be obtained by analytical or numerical integration. Table 4.1 lists different typical parametric motion models used for vehicle trajectory prediction.

[14] In practice, the current time is often adopted as the epoch time t_0.

Table 4.1 Typical motion models used for vehicle trajectory prediction

Category	Model	Description	Use & critics
Kinematic	*Curvilinear motion model* (Schubert, Richter, and Wanielik 2008)	The vehicle is simplified as a mass point, hence its trajectories can be represented by curves, e.g., segment, circular arc, spiral. Examples are constant velocity, constant acceleration, constant turn rate and acceleration (CTRA) models. A survey of this kind of models was done by Schubert et al. (2008) .	Curvilinear motion models are simple, however, the accuracy holds only for short temporal horizons ($\mathcal{T} < 1s$) . Therefore, they are popular for the filtering (one-step prediction) in vehicle tracking applications.
	Kinematic bicycle model (Luca, Oriolo, and Samson 1998; Kuwata et al. 2008)	Kinematic bicycle model simplifies the vehicle as a two-wheel bicycle. The geometric relationship between the front wheel steering angle δ_f is assumed to follow the *Ackermann steering law*: $\delta_f = \frac{l}{R} = \frac{l\dot{\psi}}{v_x}$ (Mitschke and Wallentowitz 2004, 566).	This model arises from the path tracking problem for wheeled robots. It can be used for the so-called *closed-loop prediction*, i.e., formulating the prediction as a path tracking control problem. Thus, a steering controller needs to be integrated in the prediction process. The reference path can be the road centerline. Nevertheless, the closed-loop model is nonlinear and its stability is not always guaranteed.
Dynamic	*Dynamic bicycle model* (Huang and Tan 2006; Brannstrom, Coelingh, and Sjoberg 2010)	Dynamic models take into account the tire forces that affect the motion of a vehicle. For vehicle kinematics, dynamic bicycle model considers the lateral slip at the center of gravity caused by lateral tire forces.	Considering external forces on the vehicle, dynamic bicycle model captures vehicle dynamics much better than the two models above. However, the internal vehicle parameters (e.g., mass, tire stiffness, road friction) are not observable by exteroceptive sensors. Thus, it is often used to predict trajectory of the ego vehicle.

In the probabilistic framework, it is necessary to consider the effect of the uncertainty over the prediction horizon. This problem is called *uncertainty propagation*. The following three methods for uncertainty propagation can be found in the literature:

- *Covariance updating*: in the Kalman filter, uncertainty is modelled by the covariance of the predicted state under the Gaussian assumption. It can be computed by running recursively the prediction step in the Kalman filter during the entire prediction horizon:

$$P(k+1) = F(k)P(k)F(k)' + Q(k) \quad k = 0, \dots, N-1, \tag{4.10}$$

  where $P(k)$, $F(k)$, $Q(k)$ denote state prediction covariance, state transition matrix and process noise covariance (Bar-Shalom, Li, and Kirubarajan 2001).
- *Monte-Carlo simulation*: it is a sampling based approach. The main idea is to create initial state samples randomly according to specified probabilistic distributions. The corresponding trajectories are then generated by numerically simulating the model. Finally, the trajectories satisfying the predefined criterion will be counted and the probability can be derived. In principle, Monte-Carlo simulation technique can only approximate the intended probability and its precision depends on the sample size.
- *Stochastic reachable set representation*: this method arises from the domain of reachability analysis (Althoff 2010). Instead of generating sampled trajectories in a finite number of simulation runs, this method intends to approximate all possible trajectories by stochastic reachable sets. Reachable sets, in a geometric sense, can be represented by polytopes enclosing all the possible states.

4.4.2.2 Nonparametric approach

Nonparametric methods treat the driver (intention and controls) and the vehicle (dynamics) as a whole system and do not fix the model structure and parameters in advance. Instead, they rely on motion databases to build their prediction systems.

Lefèvre, Vasquez and Laugier (2014) summarized two groups of nonparametric approach. The first group is *prototype trajectories* based. The general idea is to represent maneuvers by so-called prototype trajectories that need to be learned from database. Two formalisms for prototype trajectories exist: *cluster* and *Gaussian process*. Once the prototype trajectories obtained offline, the prediction can be performed by matching a part of historical trajectories of the target to them online. The second group is *maneuver intention estimation* based. To enhance the accuracy of trajectory matching, this group of methods firstly estimates the current maneuver being executed then matches the historical trajectory to the prototype trajectories corresponding to the estimated maneuver. The maneuver estimation problem is often

formulated as a *probabilistic inference* problem, in which the maneuver transition mechanism is modelled by *hidden Markov model* or *dynamic Bayesian network*. The transition probabilities between the different maneuvers need to be learned from motion databases.

Nonparametric approach saves efforts in modelling and assumption making. But it heavily relies on the training data specific to driving scenarios. Second, the accuracy of probabilistic inference based on hidden Markov model or dynamic Bayesian network depends on the number of states (maneuver) and the number of features (e.g., velocity, position, context information) incorporated in the model. However, the computation complexity is exponential in these numbers, thus imposing a challenge for real-time applications.

4.4.3 Problem of long-term trajectory prediction

A main challenge for trajectory prediction is to extend the prediction horizon while guarantee a prediction accuracy. This problematic is addressed in the framework of long-term trajectory prediction. "Long-term" here stands for a horizon around 3-5 seconds. This horizon allows an AD vehicle with the speed of 130km/h to forecast a potential collision at 180m far in front.

To improve the prediction accuracy, it is of importance to examine sources of prediction error. According to Huang and Tan (2006), two types of error cause prediction deviation:

- *Type-A prediction error*: the initial-condition error, i.e., the error in state estimates; the inaccuracy of the motion model; measurement noises;
- *Type-B prediction error*: the error due to assumptions on control input.

While Type-A error is often handled in the phase of state estimation, Type-B error is the target error to be reduced in the framework of long-term prediction. Trajectory prediction methods in the literature often make constant input assumptions, e.g., the constant velocity, constant acceleration and CTRA models in the Table 4.1. Since the driver-vehicle system can be characterized as a low frequency system[15], constant input assumptions hold for short prediction horizons (less than 1s). Nevertheless, the vehicle's motion pattern is not uniform under certain circumstances, e.g., when the driver is carrying out a maneuver like lane changing or in highly dynamic situations like stop & go. Thus, the prediction accuracy of these models is quite poor in long-term horizon.

[15] According to Mitschke and Wallentowitz (2004, 657,673), the crossover frequency $\frac{w_c}{2*\pi}$ is about 0.3 – 0.5 Hz for the lateral dynamics and about 0.06 – 0.12 Hz for the longitudinal ones under the nominal operating conditions.

One method to reduce the Type-B error is to adapt the motion model's structure or parameters online. Pandey and Dalbah (2014) used a multiple model approach in which each model characterizes the dynamics in a sub-stage during a turning maneuver. The model transition logic, however, is determined by heuristics. Houenou et al. (2013) adapted the parameters of a quintic polynomial model to a recognized lateral maneuver (lane change or lane keeping). The parameters are determined by minimizing a predefined cost function. Nevertheless, the weights on different costs remain arbitrary.

Another method uses driver models to generate control inputs instead of making constant input assumption. Lefèvre (2014) employed two driver models originally developed for the traffic simulation applications to model driver's vehicle-following maneuver. Fu et al. (2014) used a proportional controller for cruising maneuver and a classical car-following model for following maneuver. In the lateral dimension, the human driver's lane keeping maneuver is modelled by a state feedback controller plus a feedforward term compensating for road curvatures in the work of Kim and Yi (2014). The prediction accuracy can be enhanced by integrating driver's model, because this method captures the human's adaptability to the environment (e.g., lead vehicle, road curvatures). A difficulty to implement a driver model lies on the determination of model's parameters.

4.4.4 Contribution to the long-term longitudinal trajectory prediction

4.4.4.1 Motivation

Longer prediction horizon could be beneficial for the AD system in terms of more foresighted reactions. The predicted trajectories of traffic vehicles can be used by the maneuver planning function at the tactical level, e.g., they can serve to assess the risk of each maneuver alternative. To address the problematic of long-term trajectory prediction, we adopt the adaptive prediction framework proposed by Houenou et al. (2013). This framework blends short-term predicted trajectory based on CTRA model and long-term trajectory based on maneuver recognition. Here our interest is the long-term prediction part. In this part, Houenou et al. decoupled the longitudinal and lateral predictions in the Frenet frame. In each dimension, a quintic polynomial motion model is used. In the lateral dimension, the proposed apporach firstly recognizes the current maneuver of the target vehicle. Then the parameters of the motion model are adapted online to match the recognized maneuver.

While Houenou et al. applied this adaptive approach for lateral trajectory prediction, they kept the constant acceleration assumption for the longitudinal prediction part. Nevertheless, this assumption is no more valid in dynamic situations in which vehicles maneuver with strongly

varying acceleration, like force-in merging and stop & go in traffic jams. Given that these situations often occur in dense traffic on metropolitan beltways, an adaptive longitudinal trajectory prediction approach contributes to the robustness of the whole trajectory prediction for automated highway driving application. Therefore, we aim to propose a longitudinal trajectory prediction method that can adapt itself to dynamic driving maneuvers. Such a method, as a counterpart of the lateral trajectory prediction method proposed by Houenou et al. enriches the long-term trajectory prediction in Frenet frame.

4.4.4.2 Framework overview

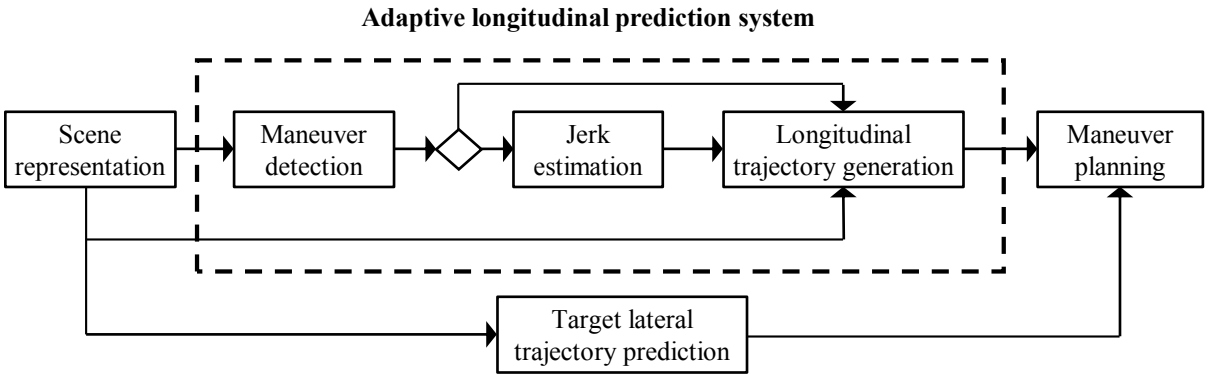


Figure 4.13 Architecture of the adaptive longitudinal trajectory prediction

Fig. 4.13 illustrates the architecture of the adaptive longitudinal trajectory prediction framework. The scene representation module provides the initial states of a target vehicle. The longitudinal maneuver recognition is formulated as a binary maneuver detection problem. If no maneuver detection is declared, the constant acceleration model is used to generate longitudinal trajectory. If a maneuver is detected, the longitudinal jerk of the maneuvering vehicle is estimated. The estimated jerk and context information are integrated in a quintic polynomial model used for trajectory generation. The predicted longitudinal trajectory with the lateral trajectory will be used to evaluate maneuver alternatives of the AD vehicle in the maneuver planning function.

4.4.4.3 Maneuver detection

The method for maneuver detection takes place in the domain of maneuvering target tracking (Li and Jilkov 2002). The fundamental question for the maneuver detection is: “Is the target maneuvering?” Answering this question is a decision problem, which can be formulated as a hypothesis testing problem

H0: The target is not maneuvering*; H1:* The target is maneuvering.

When applied for longitudinal maneuver detection in the context of highway driving, *H0* assumes the constant acceleration or slight acceleration and *H1* for strong acceleration variation.

Under *H0*, longitudinal dynamics of a nonmaneuvering vehicle are represented by the constant acceleration model

$$\mathbf{x}(k+1) = F\mathbf{x}(k) + \mathbf{w}(k),\ k = 0,1,2 \dots, \tag{4.11}$$

where $\mathbf{x}(k) = [\ \dot{s}(k) \quad \ddot{s}(k)\ ]^T$ [16] and $F = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}$ with T denoting time step size. $\mathbf{w}(k)$ is the discrete time process white noise whose covariance matrix is

$$Q = \begin{bmatrix} \frac{1}{3}T^3 & \frac{1}{2}T^2 \\ \frac{1}{2}T^2 & 1 \end{bmatrix} \tilde{q} \tag{4.12}$$

where a small power spectral density $\tilde{q}$ characterizes small jerks for a nonmaneuvering target. The measurement model is

$$\mathbf{z}(k) = \mathbf{h}(\mathbf{x}(k)) + \mathbf{v}(k)\ , \tag{4.13}$$

where $\mathbf{z}(k) = [v_x \quad a_x]^T$ is the measurement vector and $\mathbf{v}(k)$ is the measurement noise assumed to be white. The function $\mathbf{h}(\mathbf{x}(k))$ is nonlinear corresponding to the state transformation from the Frenet frame to the Cartesian frame (refer to Appendix 1), which is expressed as

$$\mathbf{h}(\mathbf{x}) = \left[\frac{\dot{s}}{cos\Delta\psi} \quad \frac{\ddot{s}}{cos\Delta\psi} + \frac{\tan\Delta\psi\dot{\psi}\dot{s}}{\cos\Delta\psi} - \frac{(2\kappa\tan\Delta\psi + c_m)\dot{s}^2}{\cos\Delta\psi}\right]^{\mathrm{T}}, \tag{4.14}$$

where $\Delta\psi$ is the deviation of the target's yaw angle from that of the road center, κ is the road curvature and c_m is curvature changing rate. (4.11) and (4.13) form an extended Kalman filter $\mathcal{F}$ based on the constant acceleration assumption. Note that at each measurement update step, (4.14) is linearized at the predicted state $\bar{\mathbf{x}}$ following the first order Taylor expansion, i.e.,

$$\mathbf{h}(\mathbf{x}) = \mathbf{h}(\bar{\mathbf{x}}) + H(\bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}}), \tag{4.15}$$

where $H(\bar{\mathbf{x}}) = \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\bar{\mathbf{x}}}$ is the Jacobian of $\mathbf{h}(\bar{\mathbf{x}})$.

[16] The reason for not considering the curvilinear abscissa *s(k)* as a system state is that the origin of the Frenet frame moves with the center of gravity of the AD vehicle. Thus *s(k)* is decided by both the dynamics of the AD vehicle and the target vehicle .

A maneuver that represents a change of motion pattern manifests itself as a large measurement innovation $\mathbf{v}(k) = \mathbf{z}(k) - \mathbf{h}(\bar{\mathbf{x}})$. The detection of such a jump in the innovation process can be done via statistical analysis. One procedure is based on the *normalized innovation squared*:

$$\epsilon_v = \mathbf{v}(k)^T S(k)^{-1} \mathbf{v}(k), \tag{4.16}$$

where $S(k)$ is covariance of $\mathbf{v}(k)$. For more details on computing $\mathbf{v}(k)$ and $S(k)$, interested readers can refer to manuals on Kalman filter (Bar-Shalom, Li, and Kirubarajan 2001).

Since the innovation sequence $\mathbf{v}(k)$ is *zero mean and white* under linear-Gaussian assumption, ϵ_v is χ^2 distributed with the degrees of freedom equal to the dimension of the measurement. In order to reduce the sample variability, the *fading-memory sum* of ϵ_v can be used in practice, which is defined by

$$\epsilon_v^\rho(k) = \rho\, \epsilon_v^\rho(k-1) + \epsilon_v(k), \tag{4.17}$$

where $0 < \rho < 1$. The effective window length s_ρ of the fading-memory sum can be obtained by

$$s_\rho = 1 + \rho + \rho^2 + \cdots = \frac{1}{1-\rho}. \tag{4.18}$$

Note that $\epsilon_v^\rho(k)$, weighted sum of Gaussian variables, is not χ^2 distributed. By moment-matching approximation, $\epsilon_v^\rho(k)$can be approximated as a scaled χ^2 distribution described by

$$\epsilon_v^\rho(k) \sim \frac{1}{1+\rho}\chi^2_{n_\rho}, \tag{4.19}$$

where the number of degrees of freedom is

$$n_\rho = \frac{n_z(1+\rho)}{1-\rho} \tag{4.20}$$

with n_z for the dimension of the measurement $\mathbf{z}(k)$.

Given that $\epsilon_v^\rho(k)$ is approximately χ^2 distributed under *H0*, the test of the hypothesis *H0* is a χ^2 test, i.e., *H0* is rejected, if

$$\epsilon_v^\rho(k) > \lambda = \frac{1}{1+\rho}\chi^2_{n_\rho}(1-\alpha), \tag{4.21}$$

where α is the probability of false alarm.

In summary, $\epsilon_v^\rho(k)$ is a *maneuver detector* which declares a maneuver detection if (4.21) is true. The main advantage of χ^2 test based maneuver detector is its simplicity, because only innovations of a filter are needed to be monitored. Even if the detection λ can be chosen according to (4.21) to theoretically guarantee a limited false alarm probability (α), it needs to

be tuned based on subjective evaluation of the results obtained from offline Monte Carlo simulation runs.

4.4.4.4 Quintic polynomial model and jerk estimation

When a target is on nonmaneuvering state, the constant acceleration model (4.20) is used for trajectory prediction, which is consistent with *H0*.

When a maneuver detection is declared, trajectories of the target are predicted based on quintic (fifth order) polynomials. Having good characteristics such as time-continuous curvature and jerk-optimality[17], quintic polynomials are widely applied for on-road vehicle trajectory prediction and planning (Werling et al. 2010; Lawitzky, Wollherr, and Buss 2012; Houenou et al. 2013). A quintic polynomial can be regarded as a realization of a maneuver that transfers the vehicle from an initial state to a target state. This target state can be interpreted as a goal to be achieved by a maneuver. In case of a lane changing maneuver, the target state can be the lateral position of the desired lane center. For a braking maneuver, it can be a desired velocity.

Just as in the case of maneuver detection, the states used to determine polynomial parameters do not contain the curve abscissa $s(t)$. The velocity profile is hence a quartic polynomial:

$$\dot{s}(t) = c_4 t^4 + c_3 t^3 + c_2 t^2 + c_1 t + c_0 \tag{4.22}$$

We propose an analytical method to determine the coefficients in (4.22). Including the duration of the maneuver t_1, there are six undetermined parameters $[c_0, c_1, \dots, c_4, t_1]$. The initial state vector is defined by $[\dot{s}(0) \quad \ddot{s}(0) \quad \dddot{s}(0)]^T$ and the end state vector is $[\dot{s}(t_1) \quad 0 \quad 0]^T$. The motivations for using such initial and end states are stated as followed:

- The initial jerk $\dddot{s}(0)$ as the initial tangent of acceleration curve depicts the tendency of the movement evolution. As shown in Fig. 4.14, the bigger the jerk's absolute value is, the less time is needed to reach final steady state, nevertheless, the stronger the acceleration variation is. In this sense, the jerk describes the compromise between the comfort and the aggressivity of a maneuver.
- The end state describes the goal of the maneuver. The value of the only freedom $\dot{s}(t_1)$ — final steady velocity — can be decided according to the driving context. For a merging vehicle, $\dot{s}(t_1)$ can be set to its leader vehicle's velocity. For an on-ramp accelerating vehicle, its target velocity can be assumed to be the ramp speed limit. In the highway stop scenario, $\dot{s}(t_1)$ can be made zero.

[17]It can be proven that the time-integral of the square of jerk is convex (Takahashi et al. 1989).

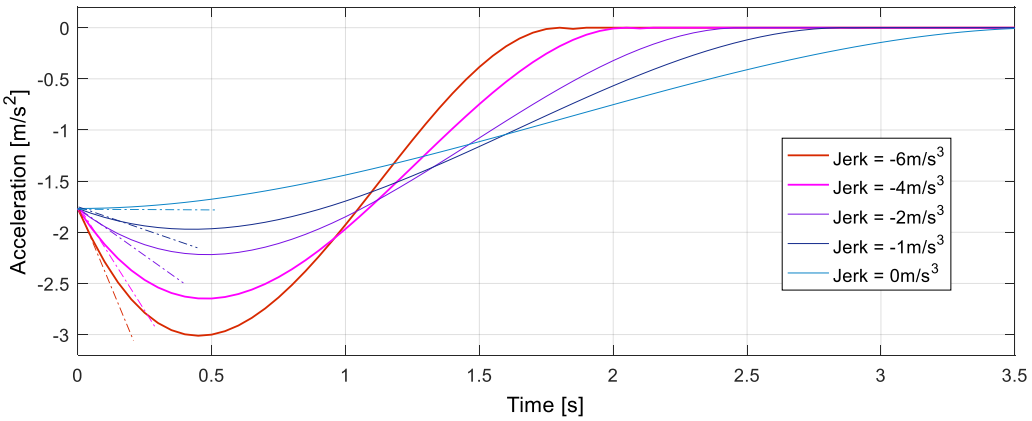


Figure 4.14 A family of the acceleration curves with different initial jerks.

Since no exteroceptive sensor can directly measure the jerk of the target vehicle, it needs to be estimated. We employ the *input estimation* approach to estimate the jerk of a maneuvering target (Bar-Shalom, Li, and Kirubarajan 2001, 428). The basic idea behind this input estimation approach is that large innovations arising in a filter can be compensated by a series of unknown inputs. If the supposed unknown input is assumed constant during a sliding window of measurements, it can be proved that the filter innovation is a linear measurement of input. Thus, the input can be estimated via *linear least squares* method from the historical innovations.

In order to estimate the unknown input, the constant acceleration model (4.11) is augmented to

$$\mathbf{x}(k+1) = F\mathbf{x}(k) + Gu(k) + \boldsymbol{w}(k), \tag{4.23}$$

where input $u(k) = \dddot{s}(k)$ and $G = \left[\frac{1}{2}T^2 \quad T\right]^T$. In contrast to $\mathcal{F}$ assuming constant acceleration, (4.23) and (4.13) forms a novel filter $\mathcal{F}^*$ assuming a constant jerk input. If a maneuver is detected at time k, given the window size of fading-memory sum s_ρ, the maneuver onset time is hence $k - s_\rho$. During the entire maneuvering range, i.e., as long as $\epsilon_v^\rho(k) > \lambda$ holds, the jerk is estimated within a sliding window with the same size of s_ρ via the following linear regression form:

$$\mathbf{y} = \Psi u + \boldsymbol{\epsilon}, \tag{4.24}$$

where the "measurement" $\mathbf{y} = \left[\mathbf{v}^{\mathrm{T}}(k) \ldots \mathbf{v}^{\mathrm{T}}\left(k - s_\rho + 1\right)\right]^T$ is a stacked vector storing the innovations of the filter $\mathcal{F}$ assuming constant acceleration, Ψ is a stacked matrix:

$$\Psi = \left[\Psi(k)^{\mathrm{T}} \ldots \Psi\left(k - s_\rho + 1\right)^{\mathrm{T}}\right]^T. \tag{4.25}$$

Ψ can be obtained through elementary manipulation of the filter $\mathcal{F}^*$. The "noise" $\boldsymbol{\epsilon}$ is a stacked vector storing the innovations of the filter $\mathcal{F}^*$. Since constant input u is considered non-random,

$\boldsymbol{\epsilon}$ is zero mean and white whose covariance matrix is a block-diagonal covariance matrix $S = \text{diag}[S(i)]$, where $i = k, k-1, \ldots, k-s_\rho+1$, $S(i)$ is the covariance of $\mathbf{v}(i)$.

The input estimate can thus be obtained by solving (4.25) via least squares method. The estimate is given by:

$$\hat{u} = (\Psi^{\mathrm{T}} S^{-1} \Psi)^{-1} \Psi^{\mathrm{T}} S^{-1} \mathbf{y} \tag{4.26}$$

with the resulting covariance matrix

$$L = (\Psi^{\mathrm{T}} S^{-1} \Psi)^{-1}. \tag{4.27}$$

With the estimated jerk $\dddot{s}(k)$ and $\dot{s}(t_1)$ that can be inferred from the driving context, the initial states and the end states provide six equality constraints under which the six parameters can be uniquely determined.

4.5 Implementation and experiments

4.5.1 Demonstration of the scene representation

The proposed concept of scene representation was implemented into a Simulink S-function (MathWorks 2016) via C-programming. The input interface consists of the ego vehicle state vector (4.2) and a list of tracks containing the state vectors of the tracked traffic vehicles (4.3). The output interface contains an array of vectors corresponding the matrix in Fig. 4.12. Each vector in this array corresponds to a state vector of the traffic vehicle in the Frenet Frame (4.6). In addition, a vector holding map-matching information is pre-allocated in the output interface. The contents in this vector are flexible according to the downstream functions.

The S-function "scene representation" was co-simulated with SCANeR Studio (OKTAL 2015), a commercial driving simulation software. The SCANeR Studio provides a virtual traffic scene and smart sensor models that feed the state vectors of the detected traffic vehicles to "scene representation". Fig. 4.15 illustrated a congested traffic scene and a screenshot of the index matrix of traffic vehicles. The AD vehicle is surrounded by a green circle and is indexed by "0". The representation of the relative relations to traffic vehicles into a matrix form is intuitive.

Fig. 4.16 shows a fluid traffic scene with a mainline roadway and a ramp. The on-ramp vehicle with the index of "4" is captured in the ramp matrix (case in orange).

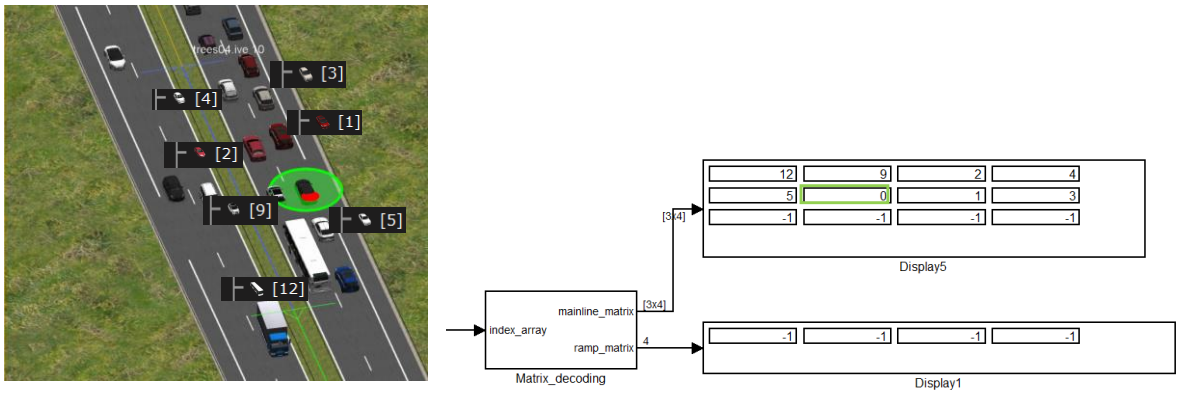


Figure 4.15 Illustration of a congested traffic scene and the index matrix of tracked traffic vehicles. (The case with "-1" means no vehicle in the corresponding zone.)

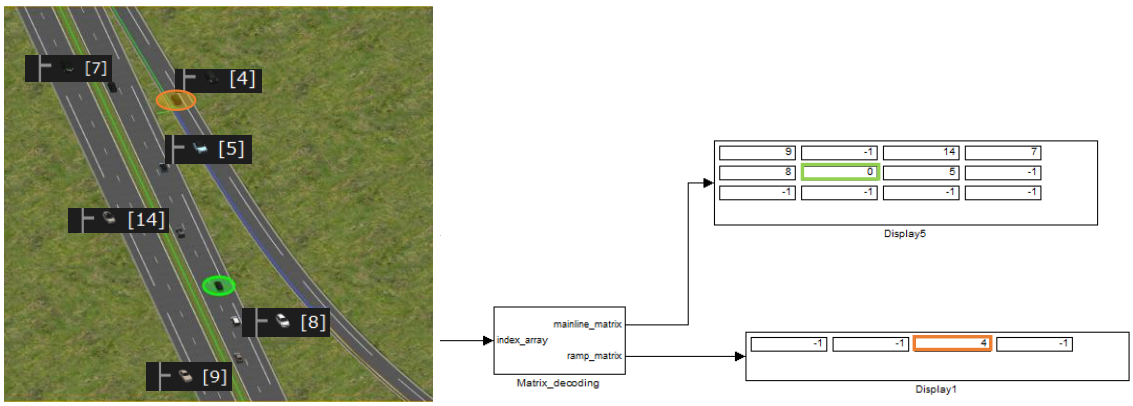


Figure 4.16 Illustration of a fluid traffic with a mainline road and a ramp. The ego vehicle is surrounded by a green circle. The on-ramp vehicle is highlighted in orange.

4.5.2 Evaluation of the adaptive trajectory prediction method

4.5.2.1 Experiment setup

The proposed adaptive prediction method was evaluated by simulation. We utilized the data collected from our previous driving simulator experiments. To investigate the ability of the prediction method to deal with dynamic situations, we purposely extracted trajectories involving strongly varying vehicle dynamics from a highway merging scenario. In this scenario, the target vehicle first accelerates to reach the speed limit of the ramp and then merges in a slowly dense mainline traffic. Once merged into the small gap in the mainline, the target vehicle needs to decelerate to adapt its speed to the leader. The dataset for this evaluation contains 12 trajectories realized by human drivers. Each sample in the dataset corresponds to a state vector (4.1). Fig. 4.17 shows the velocity and longitudinal acceleration samples.

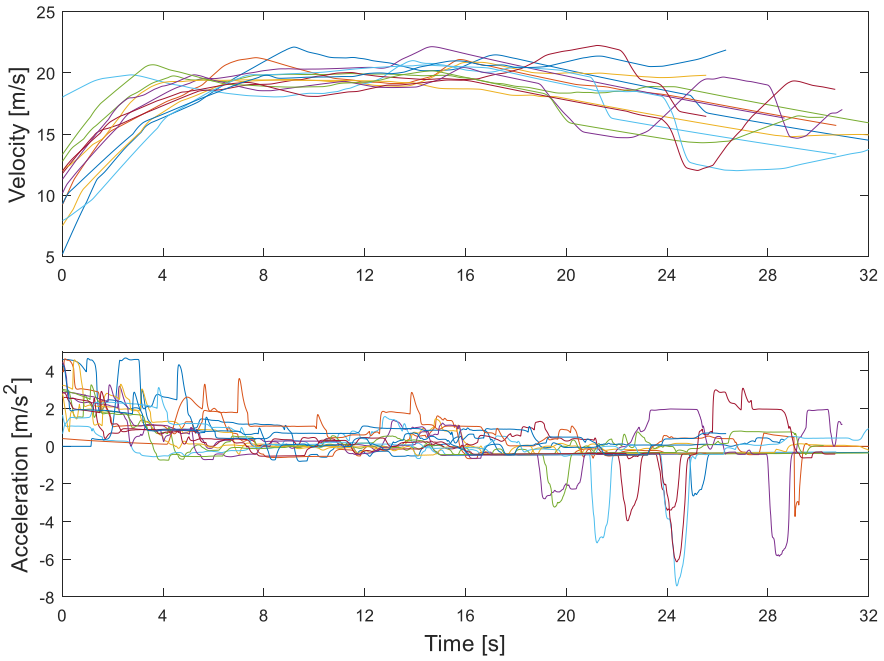


Figure 4.17 Longitudinal acceleration and yaw rate samples in the database

The samples in the dataset were firstly converted into the Frenet frame following (4.6). Then samples were replayed to feed the trajectory prediction algorithm. In order to improve the simulation fidelity, process noise and measurement noise were added in the simulation. The general stochastic simulation structure is illustrated in Fig. 4.18.

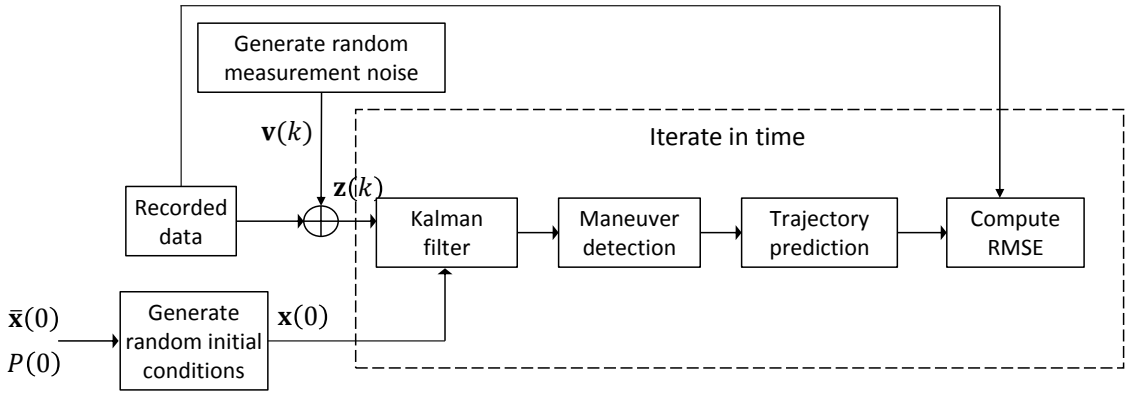


Figure 4.18 Simulation structure

The parameters used for simulation are summarized in Tab. 4.2.

Table 4.2 Parameters used in the simulation

Symbol	Description	Value [units]
T	Prediction update step	0.05 [s]
ρ	Fading factor	0.8

λ	Threshold for maneuver detection	18.3
$\sigma_{\dot{s}}$	Noise on longitudinal velocity	0.2 [m/s]
$\sigma_{\ddot{s}}$	Noise on longitudinal acceleration	0.3 [m/s^2]
$\tilde{q}$	Process noise power spectral density	0.2 [m^2/s^5]

4.5.2.2 Results

We start with a qualitative analysis of the prediction results for a hard braking maneuver in the data set. Fig. 4.19 shows the results on state estimation, maneuver detection and jerk estimation. The hard braking maneuver is correctly detected. Note that the delay on maneuver termination detection is more important than that on maneuver onset detection. This is due to the fact that the fading-memory sum needs more time to disperse large innovation accumulation during maneuver period. Since there is no jerk sensor in the driving simulation software, the true jerk was obtained via the numerical differentiation on the recorded acceleration. It demonstrates that the input estimation method can effectively estimate the jerk in this scenario.

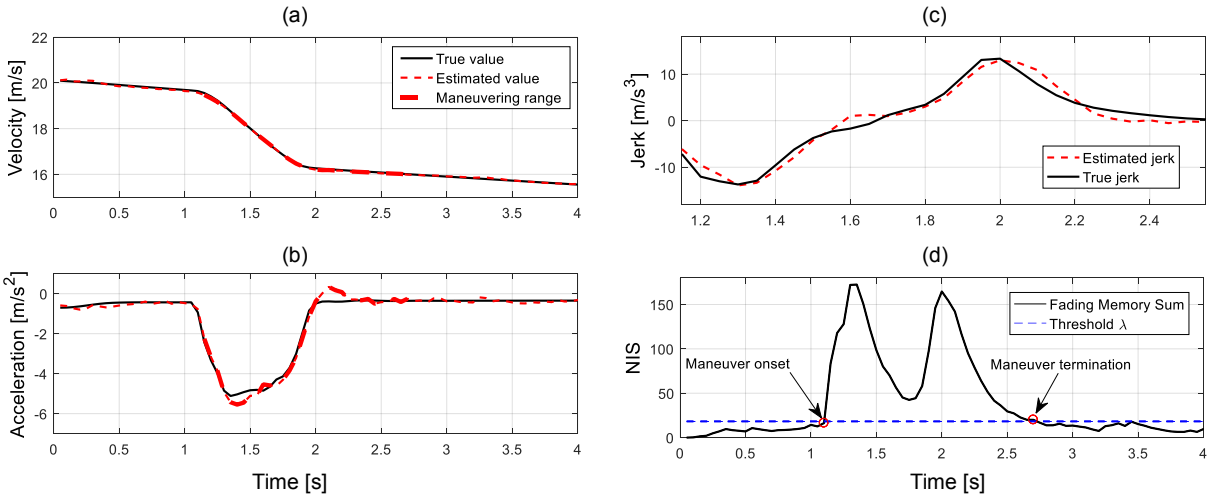


Figure 4.19 (a) Maneuver velocity profile with its estimates; (b) Maneuver acceleration profile with its estimates; (c) Normalized Innovation Squared (NIS) for maneuver detection; (d) Jerk estimation

Fig. 4.20 shows the two snapshots of the trajectory prediction for this braking maneuver. The prediction based on the constant acceleration model is denoted by "CA prediction, whereas that the proposed adaptive prediction method is labelled as "ADV prediction". The prediction horizon in this example is 3.5 s. It is clear to see that the adaptive prediction method successfully captures the tendency of the movement by incorporating the jerk. What's more, the predicted trajectory is constrained by the final state inferred from the scenario (the velocity of the lead vehicle in this case).

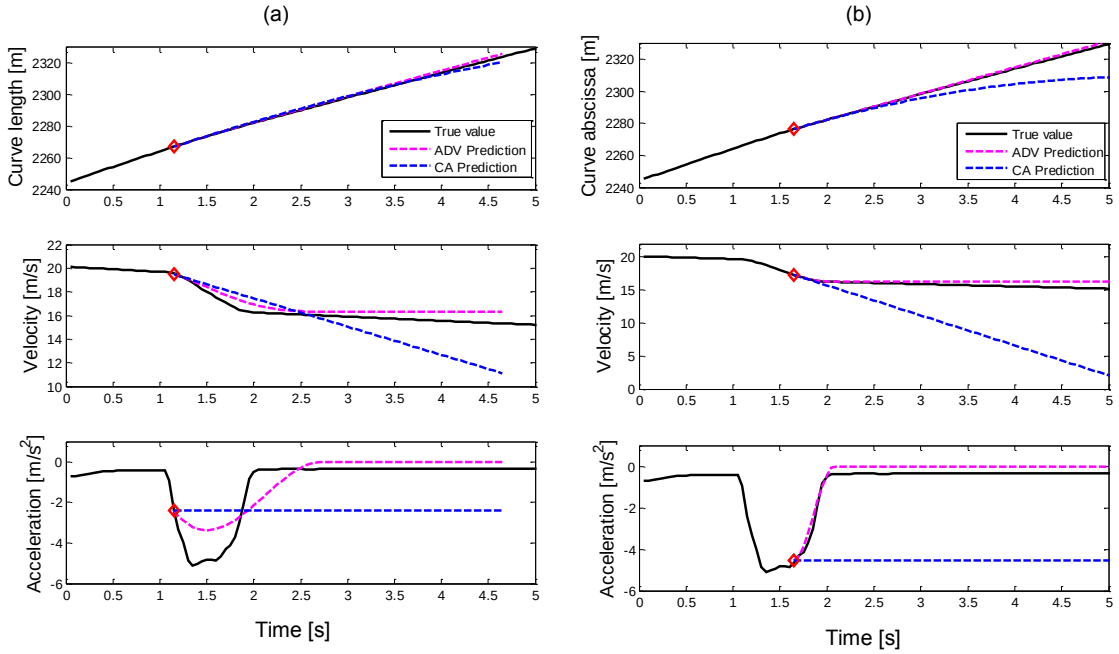


Figure 4.20 Exemplary prediction results for a braking maneuver: (a) t= 1.1 s (b) t= 1.65 s

To give a quantitative evaluation on the prediction performance, we calculated the *root mean square error* (RMSE) between the predicted trajectory and the true trajectory. The RMSE is defined as follows:

$$RMSE(p) = \sqrt{\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}|p'(i+j|i) - p(i+j)|^2}, \tag{4.28}$$

where M stands for the number of the samples used for the test, N for the number of the samples within a prediction horizon t_h, $p'(i+j|i)$ for the predicted value at the $(i+j)^{th}$ time step by starting from the true value $p(i)$ at the i^{th} time step.

We ran tests with the prediction horizons $t_h = [1, 2, \dots, 5]$ seconds. Moreover, the constant acceleration model is used as a reference for the comparison. Based on the results of maneuver detection, we compared the prediction results only on the range when maneuver detection was declared. Tab. 4.3 shows the test results. For $t_h = 1$s, the ADV and CA have almost the same high performance. It is consistent with the statement that the constant acceleration (CA) model performs well for short-term prediction (refer to Section 4.4.3). As the prediction horizon increases, the gap of the prediction accuracy between these two methods increases sharply. The ADV method yields much better prediction accuracy for long-term prediction horizons.

Table 4.3 Error comparison between ADV and CA (constant acceleration) based methods

t_h [s]	Method	$RMSE(s)$ [m]	$RMSE(\dot{s})$ [m/s]
1	ADV	0.45	0.70
	CA	0.43	0.75
2	ADV	1.17	1.03
	CA	1.42	1.68
3	ADV	2.04	1.29
	CA	3.12	2.72
4	ADV	2.91	1.46
	CA	5.53	3.64
5	ADV	3.86	1.65
	CA	8.49	4.49

We are also interested in the performance of the whole adaptive prediction framework using the method of Houenou et al. for lateral prediction (refer to Fig. 4.13). We thus tested this framework for the entire data set (12 trajectories) and compared the results with the CTRA model which is a curvilinear model for planar motion prediction (refer to Tab. 4.1). In this test, we calculated the RMSE of the Euclidian distances between the predicted positions and true real positions in the Cartesian frame, which is denoted by $D = \|\mathbf{x}_{2\mathrm{D}}'(i+j|i) - \mathbf{x}_{2\mathrm{D}}(i+j)\|$. Tab. 4.4 listed the results for long-term prediction horizons. The results show that the whole framework significantly reduces the long-term prediction errors compared to the CTRA model which assumes constant input.

Table 4.4 Error comparison between ADV (with lateral trajectory prediction) and CTRA based methods

t_h [s]	Method	$RMSE(D)$ [m]
3	ADV	1.61
	CTRA	4.82
4	ADV	2.20
	CTRA	5.24
5	ADV	2.99
	CTRA	6.11

4.6 Conclusion

In this chapter, we presented our concept of a situation assessment framework for AD systems. We firstly proposed an approach for highway driving scene representation. The resulted matrix-form scene model serves as an interface between the perception layer and the planning and control layer of an AD system. In this scene model, the dynamic states of traffic vehicles are transformed into the road-based Frenet frame. In this way, the motion of a traffic vehicle is tightly related with road curves, which facilitates the AD system to assess its relative position and future motion. The scene representation part ends with a qualitative representation for the description of highway traffic scenes. The Frenet coordinates of the traffic vehicles are mapped into the symbolic vocabulary based on their relative positions to the ego vehicle. A high-precision digital map plays a pivotal role in the scene modelling process.

In the second part of this chapter, we presented an adaptive vehicle longitudinal trajectory prediction method. Inspired by the maneuver-based method proposed by Houenou et al. (2013), our method adapts the trajectory prediction to vehicle longitudinal maneuvers. While the constant acceleration model is used to represent the uniform motion of a nonmaneuvering vehicle, a maneuvering vehicle's trajectory is modelled as a quintic polynomial. In order to make a reliable long term prediction, the estimated jerk and context information are integrated in the polynomial to characterize the maneuver's aggressivity and goal respectively. The overall method was tested on recorded human driving data from a simulator in a dynamic highway merging scenario. The results show the proposed method has higher prediction accuracy than the constant acceleration based method in such a dynamic scenario.

The developed situation assessment function will be used by the two cooperative control frameworks in Chapter 5 and 6 respectively.

5 PRINCIPLE FOR MANEUVER COOPERATION: COOPERATIVE MANEUVER PLANNING

5.1 Introduction

In Chapter 3, we have proposed two principles and a general architecture for driver-vehicle cooperation. Chapter 5 and 6 present two case studies in which we design two cooperative control systems that implement these two principles. In each case study, we follow the UCD process. We design the system in a concrete use case considering potential user needs. We develop the system functions based on the hierarchical cooperative control architecture which is adapted to the use case. Finally, the developed prototype is evaluated in a user study.

This chapter presents the implementation of the *principle for maneuver cooperation*. We define highway merging management as a use case for maneuver cooperation, given the multiple maneuver alternatives available for the AD vehicle to interact with an on-ramp merging vehicle. In this use case, we adapt the hierarchical cooperative control architecture for vehicle longitudinal control. At the tactical level, we aim to develop a maneuver planning function which allows the driver and the system to share their maneuver plans to handle a merging vehicle. The final decided plan shall be executed automatically by the system functions at the operational level.

This chapter is organized as follows. Section 5.2 presents the use case of highway merging management. The adaptation of the system architecture for the use case is then addressed in Section 5.3. Section 5.4 presents a cooperative maneuver planning function which is modelled

as a hierarchical finite state machine (HFSM). Section 5.6 describes the design of HMI for maneuver cooperation. Section 5.7 and 5.8 are dedicated to the evaluation. Section 5.7 presents a computer simulation study to illustrate the potentials of this principle in managing highway merging situations, whereas Section 5.8 presents a user study with 22 participants in which the cooperation principle and the designed HMI were evaluated.

5.2 Use case: highway merging management

Strong interaction exists among road vehicles at a highway entry section. Constrained by the end of acceleration lane, on-ramp vehicles have to merge into the mainline. However, they should give way to those vehicles already on the mainline according to traffic regulations. This special configuration leads to different interaction patterns between merging vehicles and mainline vehicles. As observed in daily life, some merging vehicles filter in by forcing the mainline vehicles to decelerate. In an inverse case, some mainline vehicles voluntarily decelerate to let merging vehicles in. In some countries, drivers follow the so-called "zipper merge" convention in traffic jam. In this convention, vehicles alternate between passing and yielding near the lane closure area in a zipper fashion (Cassidy and Ahn 2005).

These complicated interactions among vehicles motivate us to select highway merging management as a use case for driver-vehicle cooperation design. As shown by Fig. 5.1, an AD vehicle encounters a merging vehicle at a highway entry section. Multiple possible interaction patterns with the merging vehicle could result in the interference between the driver and the AD system on maneuver plan (e.g., pass, yield or lane change), thus creating potential user needs to intervene. The principle for maneuver cooperation which enables the system and the driver to share their maneuver plans may be useful for the driver in this case. Moreover, the AD system could benefit from the help from the driver to socially interact with the merging vehicle. For instance, the driver in the AD vehicle may detect that the driver in the red vehicle invites him to pass while his vehicle wants to yield. By maneuver cooperation, the driver can indicate the system to pass and a blocked situation could be avoided.

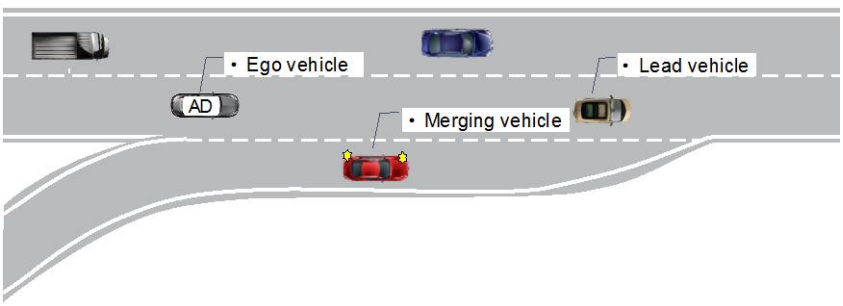


Figure 5.1. A typical driving scene on highway entry section

5.3 System architecture and assumptions

To simplify the problem, we focused on vehicle longitudinal control in this use case. This simplification can be justified by the fact that interactions between mainline vehicles and merging vehicles on this zone are mainly influenced by their longitudinal positions and dynamics. Therefore, we aim to design a *cooperative longitudinal control system* to realize the maneuver cooperation in this use case. Another strategy to deal with the merging situation—performing a lane change will be addressed in Chapter 6.

We recall the hierarchical cooperative control architecture proposed in Chapter 3 in which the information flows in maneuver cooperation are highlighted (Fig. 5.2).

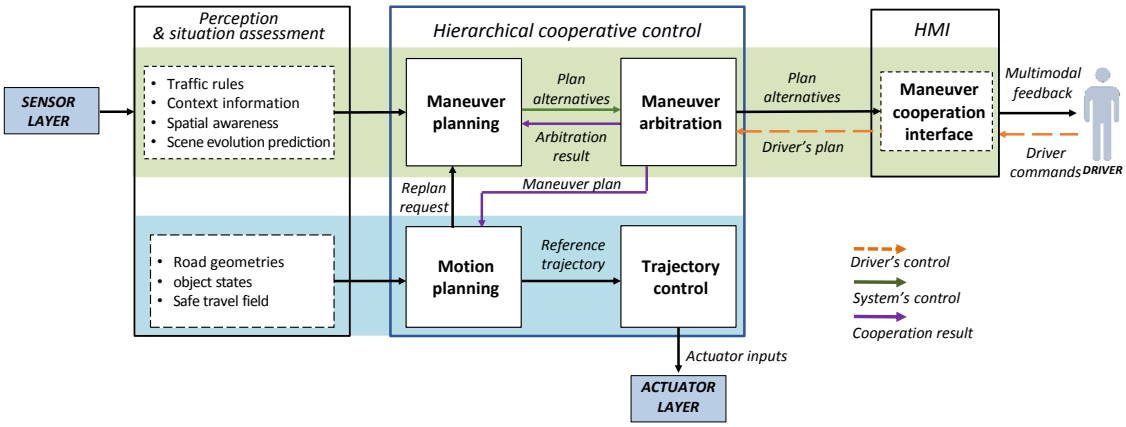


Figure 5.2. Information flows in the hierarchical cooperative control architecture (the strategical level is omitted for clarity)

Based on this general architecture, we derive a more specific functional architecture for the cooperative longitudinal control system. This architecture is sketched in Fig. 5.3. The information flows in Fig. 5.2 indicate the key role of *maneuver planning* and *maneuver arbitration* functions. In this use case, they are integrated within a function named *cooperative maneuver planning*. This function is responsible for generating maneuver plans, "pass" or "yield", to operational-level functions on the one hand and interacting with the human driver on the other hand. The functions at the operational level are realized by an adapted ACC controller. In addition to classical cruising and car following functions, the ACC in this architecture can perform the maneuver plan generated from the tactical level thanks to a transient trajectory generation function. The situation assessment function is the same one developed in Chapter 4.

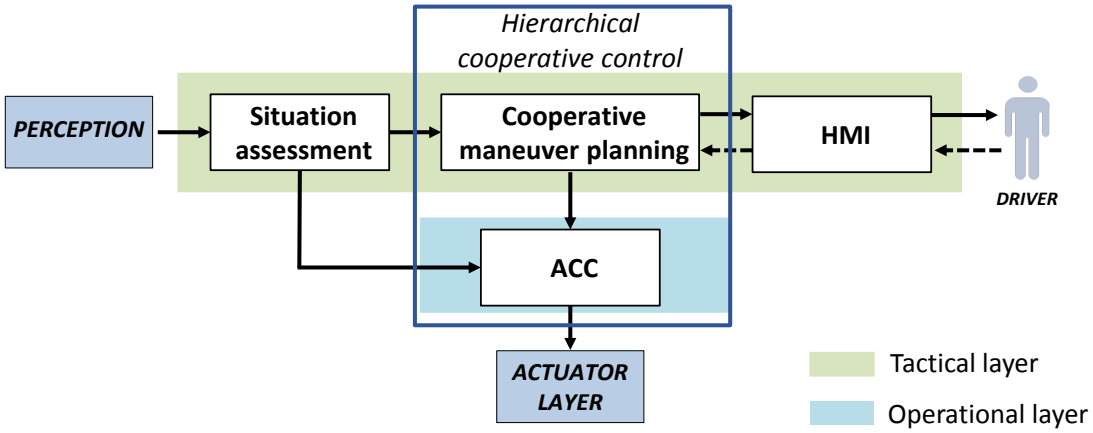


Figure 5.3. Architecture of the cooperative longitudinal control system for the use case

We assume that the system automatically keeps the vehicle in the rightmost lane. This function can be realized by the cooperative steering control system developed in Chapter 6. As for the traffic environment, we assume that there is only one merging vehicle on the ramp in the current development stage.

5.4 Cooperative maneuver planning

5.4.1 Previous works and the design choice

It is well recognized that congestions arise frequently at highway merging section. In order to improve the traffic efficiency, most of research works on highway merging management focused on how to coordinate vehicles in this zone, either by optimizing physical infrastructure layout or by using emerging *vehicle-to-vehicle* (V2V) and *vehicle-to-infrastructure* (V2I) communication technologies (Scarinci and Heydecker 2014; Rios-Torres and Malikopoulos 2016). Few works proposed "standalone" control strategies for an AD vehicle to interact with manually driven vehicles (without V2V and V2I). Wei, Dolan, and Litkouhi (2013) addressed this topic in highway merging scenarios. They developed an optimal motion planning framework which planned a velocity profile with the lowest cost as a strategy to handle a merging vehicle. The simulation results showed a decrease of the occurrence of unsafe situations compared to a standard ACC controller. In the algorithm of optimal motion planning, the cost function was a combination of four different costs penalizing long travelled distance, large acceleration, small gap and fuel consumption. However, the cost weights are usually determined by trial and error tuning.

As argued in Section 3.2.3, behavior-based maneuver planning fits driver-vehicle cooperation, because the situated behavior is easy for the human driver to understand. Moreover, if the AD

vehicle behaves as a normal human-driven vehicle in interaction with a merging vehicle, its behaviors are expected to be anticipated or understood by the merging vehicle. Therefore, it is of interest to view how human drivers interact with each other in the highway merging scenario. Several simulation models on human merging and yielding behaviors (Hidas 2005; Choudhury, Ramanujam, and Ben-Akiva 2009) decomposed this interaction into two phases. In the first phases, the mainline vehicle decides whether to keep its original cruising/car-following state or to yield to the merging vehicle by decelerating. At the same time, the merging vehicle checks if the gap with the mainline vehicle is safe enough to merge. In this phase, both vehicles manifest their intentions by small amounts of acceleration or by social cues. The second phase corresponds to the stage when the situation becomes clear and each one engages an action to terminate the interaction. Either the merging vehicle initiates the merge and the mainline vehicle yields, or the former decelerates to merge behind and the latter passes.

We use HFSM—a behavior-based formalism to model the maneuver planning strategy in this use case. In the HFSM, a maneuver plan can be modelled as a state and the driver's intervention can be integrated as an event for a transition in the HFSM. We design the hierarchy of states and the transitions between them according to the human's driving strategy as introduced above. As pointed out by Chen et al (2009), two main limits of HFSM-based behavior generation are "its reduced robustness to handle unexpected situations and the lack of 'creative' solutions/behaviors". A common solution to mitigate this drawback is to set a specific exception-handling mode to bring the vehicle into a known state.

5.4.2 Overview of the HFSM-based maneuver planning

Fig. 5.4 depicts the planning strategy modelled as HFSM. Two meta-states *highway cruising/car-following* and *highway merging management* constitute the first-level representation. Before meeting a merging vehicle, the AD system is in *highway cruising/car-following* where the function of the AD system in longitudinal dynamic control is the same as that of a standard ACC. Once the AD system detects that the presence of a merging vehicle could influence its original state, it enters *highway merging management* and begins to interact with the merging vehicle.

The *highway merging management* is broken into two phases. The system firstly enters *intention phase* in which the system adjusts the ego vehicle's distance to the merging vehicle according to its intention—*intended pass*, *intended yield* and *no intention*. *Decision phase* corresponds to the phase in which the system engages an action, *engaged pass* or *engaged yield*, to terminate the interaction with the merging vehicle.

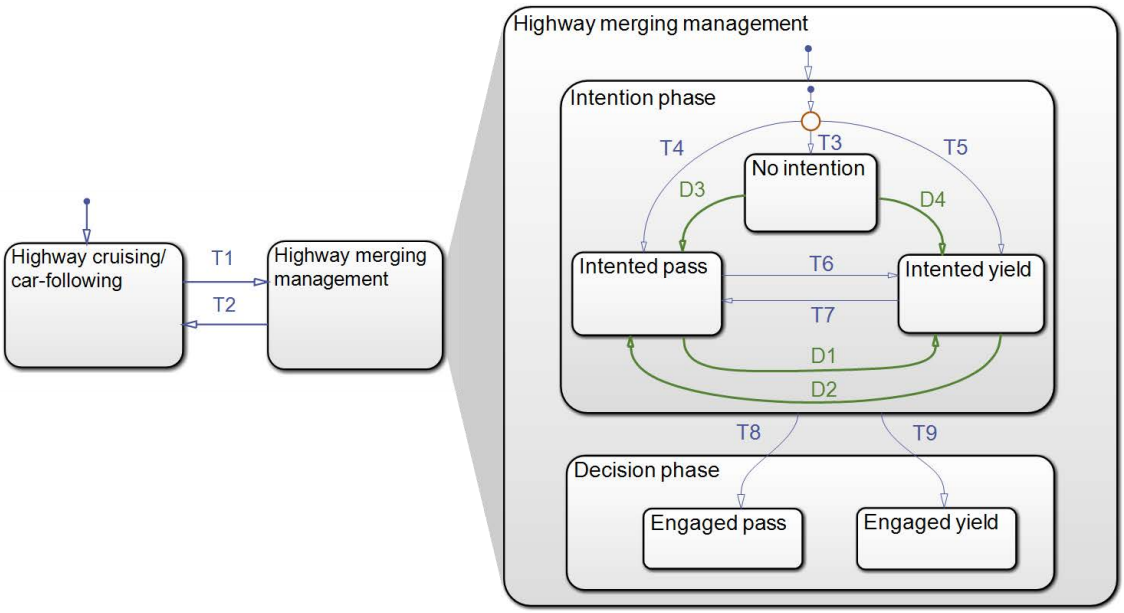


Figure 5.4. HFSM-based maneuver planning for highway merging management

The cooperation occurs in *intention phase*. When the system has its intended maneuver, e.g., *intended pass* is active, the sub-state corresponding to the alternative maneuver—*intended yield* in this case—is declared as being available to the driver. The driver can trigger the transition towards the alternative (D1 or D2 in Fig. 5.4). This alternative hence becomes the system's intended maneuver. *No intention* is set for congested traffic. With *no intention* active, both *intended pass* and *intended yield* are shown available to the driver. The driver can enable either the transition D3 or D4 in Fig. 5.4.

After explaining the general strategy and the state hierarchy, we will present conditions of the transitions in the HFSM and system's behavior in each state.

5.4.3 Transition conditions

5.4.3.1 Main factors influencing state transitions

There are two main factors that influence the state transitions in the HFSM: the gap between the merging vehicle and the ego vehicle in the longitudinal dimension and the merging vehicle's

lane change maneuver in the lateral dimension. Let s_{mv} denote the gap[18]. In a merging scenario, a merging vehicle can have two locations in the matrix-form scene representation proposed in Section 4.3, as shown in Fig. 5.5. It can be either located in one of the cases 12-15 on the ramp or in one of the cases 8-11 on the acceleration lane[19].

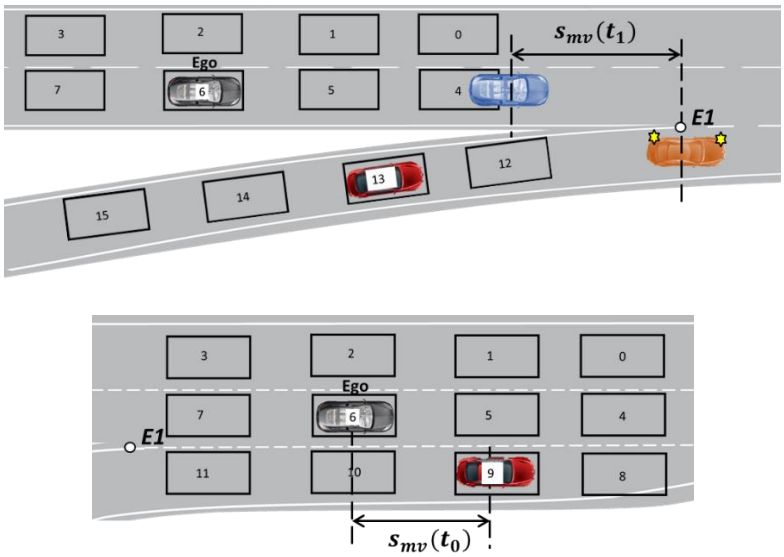


Figure 5.5. Two configurations of a merging scene: a merging vehicle is located either in a case on the ramp (up) or on the acceleration lane (bottom)

When the merging vehicle is located on the ramp, the AD system predicts the future gap at the instant t_1 when the merging vehicle surpasses E1 (Fig 5.5-up), the first possible point to merge. In this way, the AD system takes into account the current dynamics of the merging vehicle, especially considering that a merging vehicle often accelerates on the ramp. To calculate $s_{mv}(t_1)$, the system predicts the trajectory of the merging vehicle and simulates the trajectory of the ego vehicle within a prediction horizon $\mathcal{T}$. The trajectory of the merging vehicle is predicted using the method proposed in Section 4.4. The trajectory of the ego vehicle in the same horizon is obtained by iterating the constant acceleration model. If at the time sample t_k in the horizon, the predicted position of the merging vehicle begins to exceed E1, then the gap $s_{mv}(t_k)$ at t_k is used to approximate $s_{mv}(t_1)$. This method is illustrated in Fig. 5.6. In summary, depending on the merging vehicle's location in the scene, either the current gap $s_{mv}(t_0)$ or the

[18] Distance gap measures the curvilinear distance between the tail of the lead vehicle and the front bumper of the ego vehicle (SAE 2015). In our case, given that the coordinates of the merging vehicle are represented in the Frenet frame whose origin is the projection of the ego vehicle's CoG on the road curve (Section 4.3.3), its longitudinal component s_{mv} can be directly used as a measure of gap. The car body lengths of two vehicles are compensated in the condition thresholds.

[19] To simplify the expression, we index the cases in the matrix rather than use their semantic meanings.

future gap $s_{mv}(t_1)$ is used in the transition conditions. They are uniformly denoted by $s_{mv}(t_i)$ with $i = 0$ or 1 hereafter.

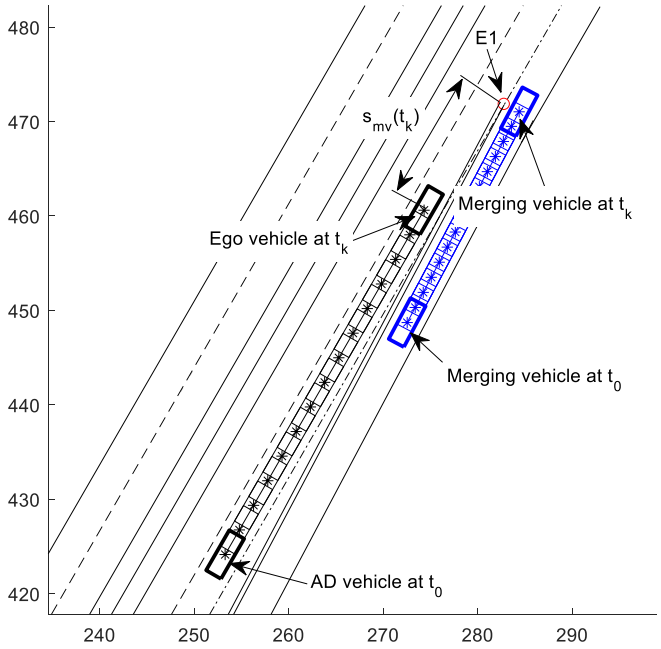


Figure 5.6 Illustration of the calculation of $s_{mv}(t_1)$

In the lateral dimension, the system monitors the lateral deviation and the heading error of the merging vehicle to the lane centreline. These measurements are used to detect the lane change maneuver of the merging vehicle. Fig. 5.7 illustrates how the gap s_{mv} and the lane change detection are used to enable different sub-states in the HFSM. Details will be explained in the following sections.

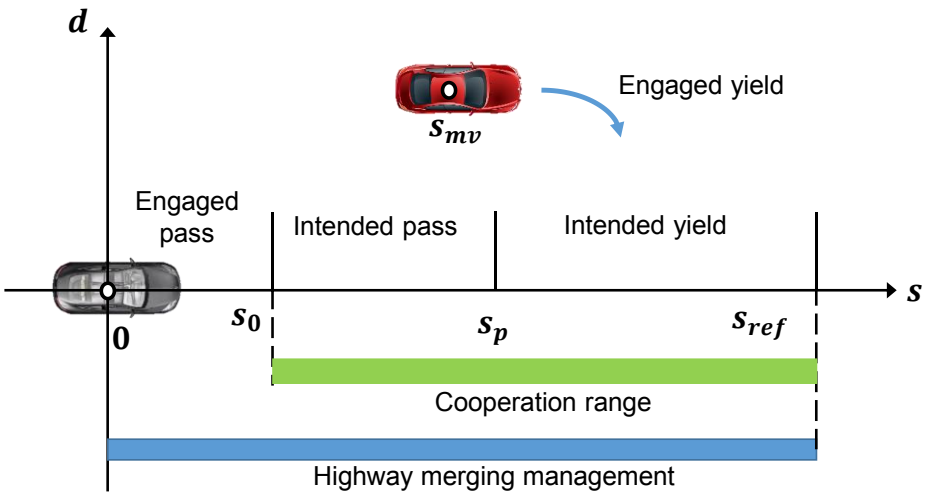


Figure 5.7. Principle of the state transitions based on the gap and the lane change maneuver of the merging vehicle

5.4.3.2 Entry and exit conditions for highway merging management

The transition from *highway cruising/car-following* (S_{hcc}) to *highway merging management* (S_{hmm}) is expressed as:

$$\text{T1: } S_{hcc} \rightarrow S_{hmm}, \quad \text{if } 0 \leq s_{mv}(t_i) \leq s_{ref}, \ i = 0 \text{ or } 1, \tag{5.1}$$

The upper bound s_{ref} is the reference distance the ego vehicle maintains with the lead vehicle in car-following mode, which is defined according to the *constant time heady policy*:

$$s_{ref} = h\dot{s}_{ego} + s_0, \tag{5.2}$$

where $\dot{s}_{ego}$ is ego vehicle's longitudinal velocity, h is a time constant called time headway and s_0 is a constant distance defined by

$$s_0 = (l_{mv} + l_{ego})/2 + s_{off}, \tag{5.3}$$

with l_{mv} for the car body length of the merging vehicle, l_{ego} for that of the ego vehicle and s_{off} for the safety margin. This condition can be interpreted as that the possible merge of the merging vehicle could cut a safe reference distance of the AD vehicle, therefore the AD vehicle needs to handle the merging vehicle.

The transition to exit *highway merging management* is determined by the following conditions: either the merging vehicle is surpassed by the AD vehicle, or it moves beyond s_{ref}, or it merges into the mainline. These conditions can be compactly expressed as

$$\text{T2: } S_{hmm} \rightarrow S_{hcc}, \ \text{if } s_{mv}(t_0) < 0 \ \vee \ s_{mv}(t_0) > s_{ref} \vee \ d_{mv}(t_0) < d^{la}, \tag{5.4}$$

where $d_{mv}(t_0)$ is the normal component of the Frenet coordinates of the merging vehicle and d^{la} is the coordinate of the lane border the merging vehicle crosses during the merging. d^{la} can be directly extracted from the digital map.

5.4.3.3 Transitions in intention phase

We address uncongested traffic and congested traffic separately, considering different interaction patterns between human drivers in these two kinds of situation. In a high-speed uncongested scenario, two vehicles have limited time to negotiate before the merging vehicle reaching the end of acceleration lane. Showing AD vehicle's intention to the merging vehicle helps to avoid the ambiguity. For this reason, the system chooses one between *intended pass* and *intended yield* in *intention phase*.

On the contrary, in a low-speed congested traffic, on-road vehicles have more time to communicate and negotiate. The intention to pass or to yield of a human driver is more

influenced by social factors. He may yield to the merging vehicle by courtesy or by social conventions. In other cases, he may temporally reduce the gap ahead in order to avoid being passed by everyone. Based on this reflection, we set a *no intention* state for congested traffic in which the choice to pass or yield is left to the driver.

In the current work, a threshold on the speed of the AD vehicle (30 km/h) is used to distinguish between the two traffic situations. In the future works, a more robust scenario classification method like the one developed by Reichel et al. (2010) will be studied. We use a flag γ_{traf} to discriminate between uncongested (with value of one) and congested traffics (with value of null). Hence, the condition for the transition towards *no intention* (S_{ni}) is straightforward:

$$\text{T3:} \rightarrow S_{ni}, \text{ if } \gamma_{traf} = 0, \tag{5.5}$$

In the uncongested traffic, we set a threshold s_p on $s_{mv}(t_i)$ to decide the transition to *intended pass* (S_{ip}) and *intended yield* (S_{iy}). The decision-making principle is that if the gap is large enough for the merging vehicle to accept, the AD vehicle behaves cooperatively by increasing the gap to facilitate the merge of the merging vehicle. Otherwise the AD vehicle maintains its initial control task—cruising or car-following. We use the *minimum acceptable space gap* model proposed by Hidas (2005) to determine s_p. Thus, s_p characterizes the minimum gap the merging vehicle accepts to merge. It is expressed by

$$s_p = s_0 + \begin{cases} c_l(\dot{s}_{ego} - \dot{s}_{mv}) & \text{if } \dot{s}_{ego} > \dot{s}_{mv} \\ 0 & \text{otherwise} \end{cases}, \tag{5.6}$$

where c_l is a constant parameter which is similar as the time-to-collision (TTC). The first transitions to *intended pass* (T4) and to *intended yield* (T5) are modelled as

$$\text{T4:} \rightarrow S_{ip}, \text{ if } s_{mv}(t_i) < s_p(t_i) \wedge \gamma_{traf} = 1, \ i = 0 \ or \ 1, \tag{5.7}$$

$$\text{T5:} \rightarrow S_{iy}, \text{ if } s_{mv}(t_i) \geq s_p(t_i) \wedge \gamma_{traf} = 1, \ i = 0 \ or \ 1. \tag{5.8}$$

The system can switch between S_{ip} and S_{iy} to account for the changes in the situation due to the dynamic behaviors of the merging vehicle. The transitions between them are represented by:

$$\text{T6:} S_{ip} \rightarrow S_{iy}, \text{ if } s_{mv}(t_i) \geq s_{min}(t_i) + \Delta s \wedge \gamma_{traf} = 1 \wedge \gamma_{Ep} = 0, \ i = 0 \ or \ 1, \tag{5.9}$$

$$\text{T7:} S_{iy} \rightarrow S_{ip}, \text{ if } s_{mv}(t_i) < s_{min}(t_i) - \Delta s \wedge \gamma_{traf} = 1, \wedge \gamma_{Ey} = 0, \ i = 0 \ or \ 1, \tag{5.10}$$

where Δs is a constant representing the hysteresis, and γ_{Ep} and γ_{Ey} are two flags that indicate whether the driver has already triggered D2 ($S_{iy} \rightarrow S_{ip}$) and D1 ($S_{ip} \rightarrow S_{iy}$) respectively. The rational is that once the system accepts the transition triggered by the driver, it should not trigger the transition backward to its original intention.

The cooperation with the driver occurs in *intention phase*. Driver's commands through a proper HMI are modelled as the events to trigger the transitions D1, D2, D3 and D4.

5.4.3.4 Strategies in *decision phase*

While the system communicates its intention to the merging vehicle by adjusting the gap in *intention phase*, the system engages "pass" or "yield" in a definitive way in *decision phase*. If the actual gap with the merging vehicle is sufficiently small and the velocity of the AD vehicle is higher than that of the merging vehicle, the system's state transitions to *engaged pass*. This transition is formulated as

$$\text{T8: } S_{in} \rightarrow S_{ep}, \quad \text{if } s_{mv}(t_0) < s_0 \wedge \dot{s}_{mv}(t_0) < \dot{s}_{ego}(t_0), \tag{5.11}$$

where S_{in} stands for *intention phase* and S_{ep} for *engaged pass*. The threshold s_0 has already been defined in (5.3).

The transition to *engaged yield* is triggered if a lane change maneuver of the merging vehicle is detected. For lane change detection, the method proposed by Houenou et al. (2013) is implemented. The idea is to monitor a metric that measures the lateral and heading deviations of the path of the merging vehicle to the lane center within a history window. If this divergence metric exceeds a predefined threshold, then the lane change detection is declared. As this method is also used to detect the human driver's lane change intention in the control cooperation, we will present the details in Section 6.5.3. The transition to *engaged yield* (S_{ey}) is expressed by

$$\text{T9: } S_{in} \rightarrow S_{ey}, \text{ if } D^{\rho}(t_0) > D_0, \tag{5.12}$$

where $D^{\rho}(t_0)$ the divergence metric and D_0 is a threshold.

Remark

It is possible that the merging vehicle cuts in when the AD vehicle is in *engaged pass* sub-state. Given that s_0 is small in the reference distance model (leaving a gap of 2-5 m), the cut-in of a high-speed merging vehicle with a gap smaller than s_0 may result in a collision. We assume that this critical situation is handled by a collision avoidance system.

5.4.4 System behaviors

At each sub-state, the system generates a maneuver plan for the ACC controller. This maneuver plan is translated into a target velocity ($\dot{s}_{targ}$) and a target distance (s_{targ}) for a "virtual leader" in the ACC controller.

In uncongested traffic, a "pass" maneuver means that the AD vehicle keeps its initial task—cruising at the road speed limit v_{limit} or follows the lead vehicle ($\dot{s}_{lv}$, s_{lv}) if there is one; "yield" means that the AD vehicle takes the merging vehicle as the lead vehicle and follows it with the reference distance. In congested traffic, considering the limited space for the AD vehicle, the system reduces the distance with the lead vehicle by $\frac{1}{4}h\dot{s}_{ego}$ at *intended pass*, while it increases the distance by $\frac{1}{2}h\dot{s}_{ego}$ at *intended yield*. The target velocities and distances at each sub-state are summarized in Tab. 5.1.

Table 5.1. Target velocities and distances at each sub-state in HFSM

Sub-state	Uncongested traffic		Congested traffic
	Without lead vehicle	With lead vehicle	
Intended pass / engaged pass	$\dot{s}_{targ} = v_{limit}$ $s_{targ} = s_{vl}$*	$\dot{s}_{targ} = \dot{s}_{lv}$, $s_{targ} = s_{lv}$	$\dot{s}_{targ} = \dot{s}_{lv}$, $s_{targ} = s_{lv} - \frac{1}{4}h\dot{s}_{ego}$
Intended yield	$\dot{s}_{targ} = \dot{s}_{mv}$, $s_{targ} = s_{mv}$	$\dot{s}_{targ} = \dot{s}_{mv}$, $s_{targ} = s_{mv}$	$\dot{s}_{targ} = \dot{s}_{lv}$, $s_{targ} = s_{lv} + \frac{1}{2}h\dot{s}_{ego}$
Engaged yield	$\dot{s}_{targ} = \dot{s}_{mv}$, $s_{targ} = s_{mv}$	$\dot{s}_{targ} = \dot{s}_{mv}$, $s_{targ} = s_{mv}$	$\dot{s}_{targ} = \dot{s}_{mv}$, $s_{targ} = s_{mv}$
No intention	N. A.	N. A.	$\dot{s}_{targ} = \dot{s}_{lv}$, $s_{targ} = s_{lv}$

*s_{vl} denotes the relative distance to the virtual vehicle. By setting s_{targ} always equal to s_{vl}, the virtual leader tracks only the target velocity v_{limit} in case that there is no actual lead vehicle. We refer the reader to Section 5.5.1 for details of the virtual leader scheme.

5.5 ACC controller

5.5.1 Virtual leader scheme

An ACC controller operates in two control modes depending on the presence of a lead vehicle. It tracks a desired speed in a "cruising mode" if no lead vehicle is detected. It follows a lead

vehicle maintaining a safe distance in a "car-following mode". One way to implement these two modes is to switch between two different controllers (Winner 2012, 620). A main drawback of such a mode-switching scheme is the difficulty to control vehicle's transient behavior when the mode is switched from the one to the other (Bageshwar, Garrard, and Rajamani 2004). Especially in this use case, the merge of the merging vehicle in front of a "cruising" AD vehicle changes the mode to the car-following mode abruptly, which can lead to a strong variation of the acceleration of the ego vehicle. To resolve this problem, the ACC in this use case employs the *virtual leader scheme* (see Fig. 5.8). In the cruising mode, the ACC tracks a virtual vehicle whose speed is the same with the desired speed. When there is a lead vehicle, the speed and the position of the virtual vehicle are set to be those of the actual lead vehicle. In this way, the ACC always uses a single control algorithm to follow the virtual lead vehicle, thus making the switching between the modes unnecessary.

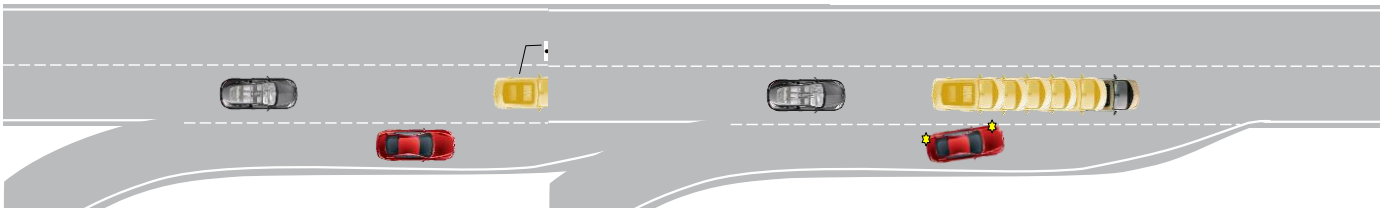


Figure 5.8. Illustration of the virtual leader scheme. Left: the AD vehicle follows a virtual leader in cruising mode; right: the virtual leader merges from the actual lead vehicle to the merging vehicle during the yield maneuver of the AD vehicle.

A central question to implement the virtual leader scheme is how to generate transient trajectories for the virtual leader, e.g., from the states of the old lead vehicle to those of the new lead vehicle. Nouvelière (2002) used a smooth trajectory generated by a first-order filter, whereas Kim (2012) developed a LQ controller to make the virtual lead vehicle move smoothly from an initial state to the desired state. In this thesis, the trajectory generation for the virtual leader is formulated as an optimization problem in the MPC framework.

5.5.2 Principles of MPC

Model predictive control is a finite-horizon optimal control approach that iteratively minimizes a cost function defined for a plant model subject to state and input constraints (Camacho and Bordons 2007). Thanks to its capability to handle state and input constraints online, MPC is well suited for trajectory generation. Subsequent improvements in both computing systems and algorithm efficiency have enlarged the range of applications in which real-time MPC can be applied (Klančar and Škrjanc 2007; Howard, Green, and Kelly 2010; Houska, Ferreau, and Diehl 2011).

Assuming the plant is modelled in discrete time by

$$\dot{\mathbf{x}}(k+1) = f\big(\mathbf{x}(k), \mathbf{u}(k)\big), \tag{5.13}$$

with the state vector $\mathbf{x}(k)$ and the input vector $\mathbf{u}(k)$, the optimization problem to be solved by MPC can be mathematically formulated as:

$$\begin{aligned}
&\text{Minimize}: J_{N_p}\big(\mathbf{x}(0), U(0)\big)\\
&\text{Subject to:}\\
&\qquad \dot{\mathbf{x}}(k+1) = f\big(\mathbf{x}(k), \mathbf{u}(k)\big), \;\; k = 0, \dots, N_p - 1,\\
&\qquad \mathbf{x}(k) \in \mathcal{X}, \quad k = 0, \dots, N_p - 1,\\
&\qquad \mathbf{x}\big(N_p\big) \in X_f,\\
&\qquad \mathbf{u}(k) \in \mathcal{U}, \;\; k = 0, \dots, N_c - 1.
\end{aligned} \tag{5.14}$$

$J_{N_p}(\mathbf{x}(0), U(0))$ is called *cost function* which incorporate objectives of optimization (in most cases it handles with multiple objectives). N_p is called *prediction horizon*, it is a time span in which J_{N_p} is to be optimized. The first parameter in the cost function is $\mathbf{x}(0)$ denoting the current state vector of the model. The second one $U(0)$ called *optimization variable* denotes a sequence of inputs, i.e.,

$$U(0) = [\mathbf{u}(0), \dots, \mathbf{u}(N_c - 1)]^T, \tag{5.15}$$

with N_c called *control horizon*. The optimization variable $U^*(0)$ that results in the smallest value of the cost function is called *optimal solution*. The part below "subject to" describes the constraints needed to be satisfied during the optimization. $\mathcal{X}, \mathcal{X}_f$ and $\mathcal{U}$ denote the sets on states, on terminal states and on input respectively. In practice, we attempt to formulate a convex optimization problem because there exist efficient methods for solving this kind of problem (Boyd and Vandenberghe 2004). In a convex optimization problem, $\mathcal{X}, \mathcal{X}_f, \mathcal{U}, J_{N_p}$ as well as the plant model (5.13) need to be convex. Assuming that the optimization problem (5.14) is resolved at t_0, yielding an optimal solution $U^*(0)$, only the first element $\mathbf{u}(0)$ is applied to control the plant. This process will be repeated at the next time step t_1, leading to a *receding horizon* control strategy.

5.5.3 MPC-based transient trajectory generation

The goal of the MPC controller is to generate a trajectory for the virtual leader from initial states to reach target states provided by the *cooperative maneuver planning* function (Tab. 5.1). The virtual leader is modelled as a double integrator with the acceleration as input:

$$\mathbf{x}_{vl}(k+1) = A\mathbf{x}_{vl}(k) + Bu_{vl}, \tag{5.16}$$

where

$$A = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} \frac{1}{2}T^2 \\ T \end{bmatrix}, \tag{5.17}$$

with T for time sample size. $\mathbf{x}_{vl}$ is the state vector $[s_{vl}(k) \quad \dot{s}_{vl}(k)]^T$ which composes the trajectory of the virtual leader. The task of the MPC controller is to make the trajectory converge to the reference $\mathbf{r}(k) = [s_{targ}(k) \quad \dot{s}_{targ}(k)]^T$. A quadratic cost function over a prediction horizon of N_p time samples is defined as

$$\begin{aligned} J_{N_p}(\mathbf{x}_{vl}(0), U(0)) = &\sum_{k=0}^{N_p-1} (\mathbf{x}_{vl}(k) - \mathbf{r}(k))^T Q(\mathbf{x}_{vl}(k) - \mathbf{r}(k)) \\ &+ u_{vl}^T(k) R u_{vl}(k) + \Delta u_{vl}^T(k) S \Delta u_{vl}(k), \end{aligned} \tag{5.18}$$

where *Q, R, S* represent diagonal weighting matrices penalizing the deviation from $\mathbf{x}_{vl}(k) = \mathbf{r}(k)$, $u_{vl}(k) = 0$ and $\Delta u_{vl}^T(k) = 0$. The cost function requires a reference $\mathbf{r}(k)$ within the entire prediction horizon. The *cooperative maneuver planning* function generates $\mathbf{r}(0) = [s_{targ}(0) \quad \dot{s}_{targ}(0)]^T$. If $\mathbf{r}(0)$ is the state vector of the lead vehicle or merging vehicle, the reference $\mathbf{r}(k)$ takes the predicted trajectory of the lead vehicle or merging vehicle using the method in Section 4.4. In the case of *intended pass* without lead vehicle, the reference can be obtained simply by integrating $\mathbf{r}(0)$ in the prediction horizon.

Inequality constraints on states and input of the virtual leader are defined as

$$\begin{aligned} 0 \le \dot{s}_{vl}(k) \le v_{limit}, \\ a_{min} \le u_{vl}(k) \le a_{max}, \\ j_{min}T \le \Delta u_{vl}(k) \le j_{max}T, \ k = 0,1 \dots N_p - 1, \end{aligned} \tag{5.19}$$

where a_{min} and a_{max} denote the minimum and maximum acceleration, and j_{min} and j_{max} denote the minimum and maximum jerk. With (5.19), the MPC takes into the constraints of the road speed limit, and the riding comfort (acceleration and jerk).

After linear matrix manipulation, (5.16), (5.18) and (5.19) can be formulated as an online quadratic optimization problem. The optimal solution is then solved using conventional optimization routines (Bemporad et al. 2002). At each time step, only the first element of the optimal solution, denoted by u_{vl}^* is applied for (5.16) as the input. This process is repeated at subsequent time steps. The resulted trajectory of the state vector $\mathbf{x}_{vl}$ is the trajectory of the virtual leader.

Fig. 5.9 shows simulation results of the MPC-based trajectory generation to perform a yield maneuver. At the beginning, the AD vehicle is cruising, therefore the virtual leader moved at 25m/s (Fig. 5.9 (c)). At 4s, the system decides to yield to a merging vehicle which is 20m behind the virtual leader (Fig. 5.9-middle (b)). The optimal control u_{vl}^{*} shown in Fig. 5.9 (a) makes the virtual vehicle smoothly merge to the merging vehicle, with lead gap and the speed difference converging to zero.

5.5.4 Feedback controller for trajectory tracking

This section deals with the design of feedback controller to track the optimal trajectory provided by the MPC component. The control law used to ensure that the AD vehicle follows a reference distance to the virtual leader is similar to that proposed by Ioannou and Chien (1993). This reference distance s_{ref} is defined by the *constant time headway policy* following (5.2). The control law is expressed:

$$a_{des} = \frac{1}{h}\left(\dot{s}_{vl} - \dot{s}_{ego}\right) + \frac{\lambda}{h}\left(s_{vl} - s_{ref}\right), \tag{5.20}$$

where a_{des} is the desired acceleration to be tracked by actuator controller and h is the same time headway in (5.2). The design parameter λ should be set positive to make the error to the reference distance converge to zero. In the cruising mode, $\dot{s}_{vl}$ is set to the road speed limit v_{limit} and the position of the virtual leader is updated by integrating the speed. In the car-following mode, $\dot{s}_{vl}$ and s_{vl} are set to those of the actual lead vehicle. To perform different maneuvers, the controller tracks the trajectory of the virtual leader determined by the aforementioned MPC approach.

In the actuator control-loop, the desired acceleration a_{des} is translated to the throttle angle or braking force by an inverse model approach based on a powertrain model and a braking system model (Nouvelière 2002).

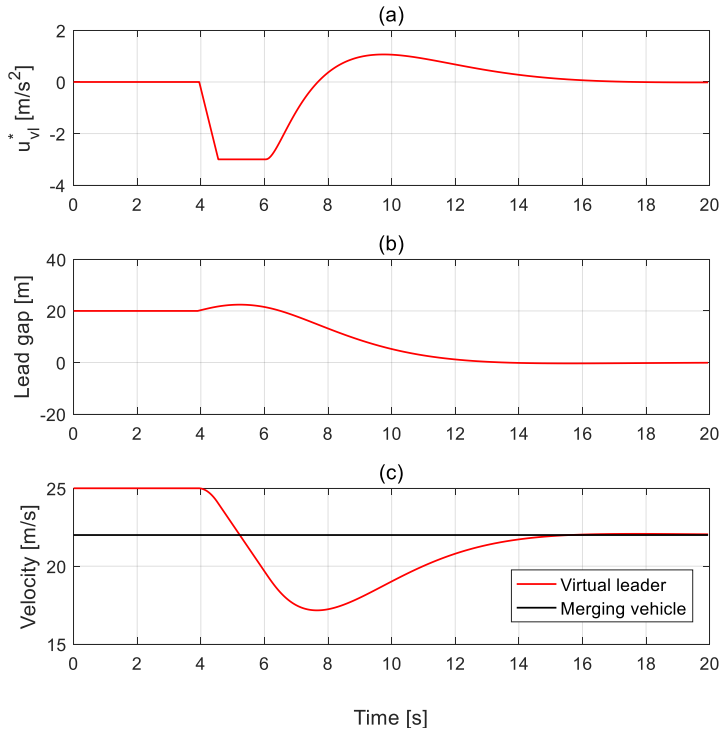


Figure 5.9. Simulation results of the transient trajectory generation for a yield maneuver. (a) Optimal acceleration of the virtual leader, (b) lead gap to the merging vehicle, (c) vehicle velocities.

5.6 Interface design

In addition to the interaction logic at the system functional level, HMI is another key factor enabling efficient cooperation between the driver and the system. The design decisions are formulated as the following three HMI goals:

- *Showing the driving context*. This principle is consistent with the concept of *common frame of reference* introduced in Section 2.2.3. The driving context serves as a common reference on which the driver and the AD system share their intentions. This principle is implemented by a representation of merging scene in a windshield Head-Up Display (HUD, Fig. 5.10-left) and a yellow semi-transparent rectangle tracking the target merging vehicle (Fig. 5.10-upper right). This yellow rectangle is the simulation of augmented reality. The appearance and disappearance of these HMI elements are consistent with the entry and the exit of *highway entry merging management* in the HFSM.
- *Showing the intention and available alternative*. First, we used triangle to symbolize maneuver, with triangle(s) forwards for “pass” and backwards for “yield”. Then we designed color codes to distinguish maneuver states. Three *blue* triangles represent an “intended/engaged” maneuver, whereas a single *green* one represents an “available”

alternative. Intention and alternative symbols are shown within the representation of merging scene in the HUD (Fig. 5.10-left).

- *Providing a way for the user to choose an alternative*. This principle was implemented by two capacitive backlit buttons (Fig. 5.10-lower right). The up button means "pass" and the down button "yield". As long as a maneuver is available, the corresponding button twinkles in green and remains active. The press on an active button will trigger a transition to the alternative in the HFSM (Fig. 5.4). As for an acknowledgement, the pressed button will become blue (lasting 2s). If the user presses when the green light is off, this button will temporally become red (2s) to signify a refusal.

Fig. 5.10 shows a synthesis of the HMI prototypes installed on the driving simulator.

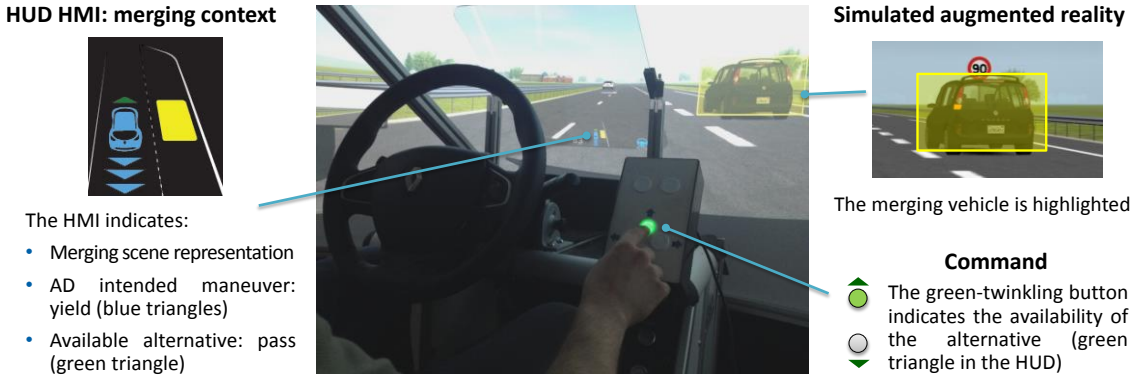


Figure 5.10. Designed HMI for the maneuver cooperation principle

5.7 Simulation study

A computer simulation of the designed system was conducted in the first place. The objective was to demonstrate the capability of the system to deal with the merging situation and potential benefits of driver-vehicle cooperation in this use case.

5.7.1 Setup

5.7.1.1 Implementation and software environment

SCANeR Studio served as a simulation platform which provided the virtual driving environment (including road network and traffic vehicles), sensor models and the interface with Simulink models. We implemented the HFSM of cooperative maneuver planning through Stateflow of Simulink (The MathWorks, Inc. 2017b). This Simulink model ran at 20Hz. The MPC-based trajectory generation was modelled by Yalmip, an open-source Matlab toolbox for optimization problem modelling (Lofberg 2004) and embedded into the Simulink model of cooperative maneuver planning. The online optimization problem was solved using the

mathematical programming solver Gurobi (Gurobi Optimization, Inc. 2017). The feedback controller in the ACC was implemented into another Simulink model which ran at 200Hz.

5.7.1.2 Parameters

Tab. 5.2 quantifies the parameters involved in the designed functions in precedent sections.

Table 5.2. List of the parameters used in the vehicle longitudinal control system in this Chapter

Sysetm	Symbol	Description	Value [units]
Cooperative maneuver planning (HFSM)	h	Time headway in the reference distance s_{ref}	2 [s]
	s_{off}	Offset distance in s_{ref}	5 [m]
	l_{mv}	Car body length of the merging vehicle	5 [m]
	l_{ego}	Car body length of the ego vehicle	5 [m]
	$\mathcal{T}$	Horizon for trajectory prediction	3 [s]
	c_f	Coeifficent in the minimum acceptable space gap	1.5 [s]
	Δs	Marginal distance in the hysteresis	1 [m]
	T	Time sample size	0.05 [s]
	ρ	Fading memeory factor	0.8
	D_0	Threshold on the divergence mertic	1
MPC	N_p	Time sample number in the prediction horizon	30
	Q	Weight matrix on tracking performance	$\begin{bmatrix} 2 & 0 \\ 0 & 10 \end{bmatrix}$
	R	Weight on control	30
	S	Weight on control rate	800
	v_{limit}	Road speed limit	25 [m/s]
	a_{max}	Maximum acceleration	3 [m/s^2]
	a_{min}	Minimum acceleration	-3 [m/s^2]
	j_{max}	Maximum jerk	5 [m/s^3]
	j_{min}	Minimum jerk	-5 [m/s^3]
ACC controller	λ	Parmameter in the feedback control law	0.2

5.7.1.3 Scenario modeling tool

Modelling such a highly interactive merging scenario imposes a new challenge: the scenario must be *interactive* and *reproducible* at the same time. “Interactive” implies that the merging

vehicle should naturally adapt its behaviors to the AD vehicle. Nevertheless, the scenario should be reproducible in the sense that each participant of the user test (which will be presented in Section 5.8) can encounter the same kind of situation. For the chosen use case, a target merging vehicle needs to be generated such that it always meets the AD vehicle with a configurable relative gap across different test runs. In the meanwhile, its microscopic merging behavior shall be controllable.

To meet this challenge, a generic scenario modelling tool prototype developed in the project LAR was used in this study. This tool is based on the traffic and the scenario modules of the SCANeR studio software (That and Casas 2011). It has three major functions:

- *Meeting control*: to ensure that the target merging vehicle always meets the AD vehicle under configurable conditions;
- *Gap control*: to control the speed of the target merging vehicle so that it keeps resting in proximity in front of the AD vehicle;
- *Lane change control*: to control the decision of the target merging vehicle to merge behind or in front of the AD vehicle.

5.7.1.4 Scenarios

With the scenario modelling tool, we modelled the following three scenarios:

- *S1-nominal merging in fluid traffic*: as shown in Fig. 5.11, the AD vehicle drives at the road speed limit in right-most lane. The merging vehicle behaves in a deterministic way, i.e., it merges in front or behind the AD vehicle according to its relative distance to the AD vehicle. The motivation of modelling this scenario is to compare the performance of the designed system with that of a standard ACC controller.
- *S2-hesitant merging in fluid traffic*: the AD vehicle meets a "hesitant" merging vehicle that does not initiate to merge even though the AD vehicle decelerates to enlarge the gap. In this scenario, we simulated an input of the driver to change the intention of the system. By this, we demonstrated the cooperation principle in uncongested traffic.
- *S3-hesitant merging in congested traffic:* the AD vehicle follows its lead vehicle with a small gap in a congestion. A merging vehicle arrives in front of the AD vehicle and waits the AD vehicle to let it merge into the lane. In this scenario, we simulated a driver who made a courtesy yield.

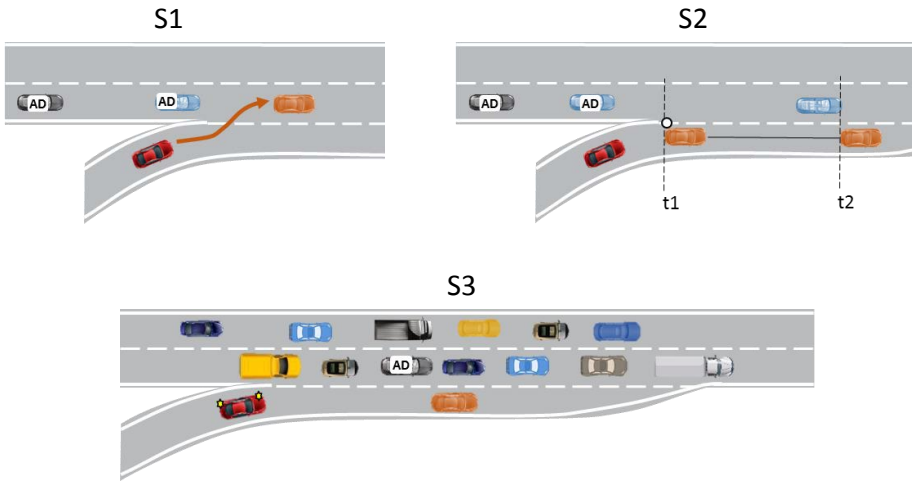


Figure 5.11. Illustration of three test scenarios: S1-nominal merging in uncongested traffic, S2-hesitant merging in fluid traffic and S3-hesitant merging in congested traffic

5.7.2 Results

5.7.2.1 S1-nominal merging in fluid traffic[20]

Within this scenario, we compared our design—cooperative longitudinal control system—to a standard ACC. The standard ACC was designed by Rajamani (2006, 153), which had a PID controller for speed control in the cruising mode and the same feedback controller (5.20) as our system in the car-following mode. This ACC considers only the lead vehicle in the same lane, so it does not react to the merging vehicle until the latter crosses the lane border.

Fig. 5.12 shows the simulation results of these two systems. Since the merging vehicle keeps accelerating on the ramp, the AD vehicle with the designed system anticipates that the merging vehicle could merge when it enters the acceleration lane according to (5.6). Therefore, the system state transitions to *intended yield* at 35.7s (Fig. 5.12(a)). Meanwhile, it begins to follow the merging vehicle by decelerating. Even if the gap s_{mv} is small, the controller does not cause strong deceleration thanks to the trajectory generated by the MPC. When the merging vehicle initiates the merge at 38.6s (Fig. 5.12(c)), the AD vehicle has already followed the merging vehicle.

The AD vehicle equipped with the standard ACC runs under the same initial conditions. After entering the acceleration lane, the merging vehicle cuts in at 43.1s (Fig. 5.12(f)). The AD vehicle reacts lately at 45.6s (Fig. 5.12(d)) until the merging vehicle enters the same lane.

[20] Two videos of the simulation are available at https://youtu.be/PGLCau-Cm5w and https://youtu.be/dfbVJQUYuhU.

Moreover, the small relative distance due to the cut-in engenders a braking of the AD vehicle which can be quite uncomfortable for on-board passengers.

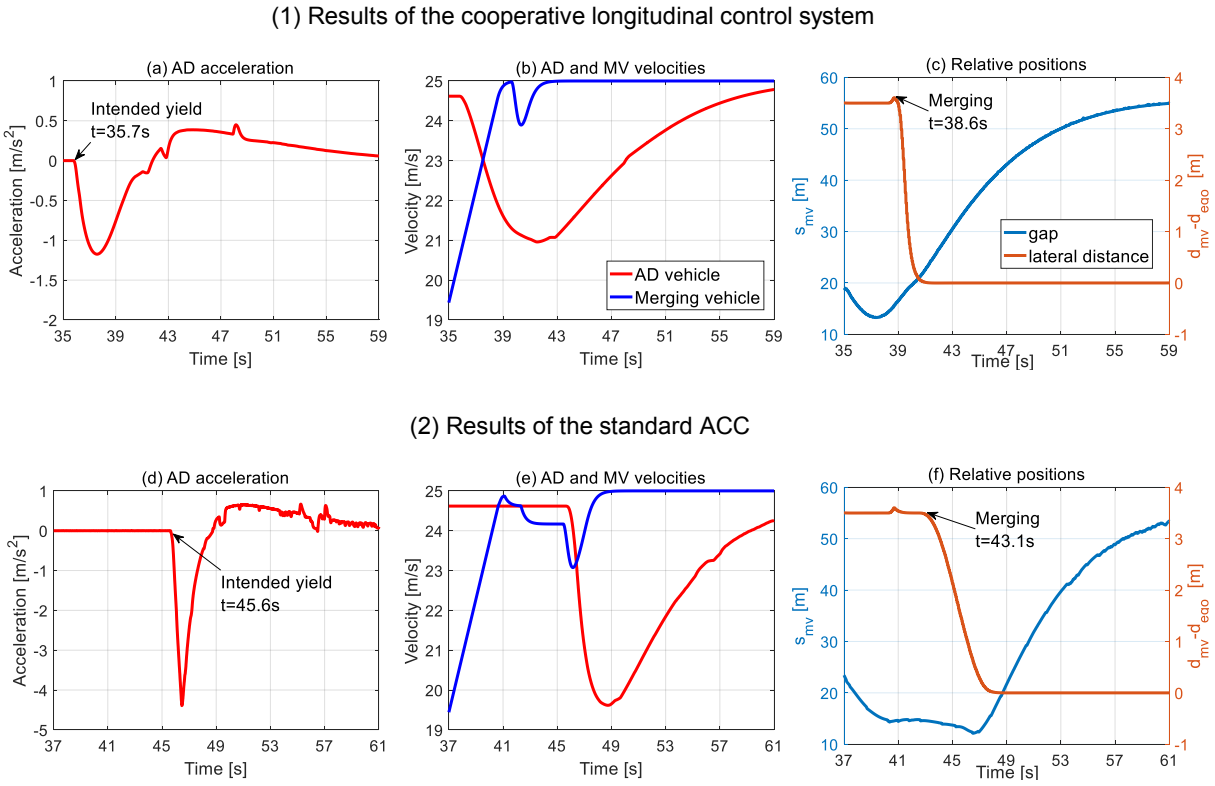


Figure 5.12. Comparison of the simulation results of the cooperative longitudinal control system and a standard ACC in S1

The simulation results demonstrate two strengths of the designed system: 1) the system can anticipate the future scene configuration and thus actively interacts with the merging vehicle by manifesting its intention: pass or yield; 2) the MPC generates smooth trajectories to perform the maneuver, thereby ensuring vehicle ride comfort.

5.7.2.2 S2-hesitant merging in fluid traffic[21]

Fig. 5.13 shows the simulation results in S2. The AD vehicle intends to yield to the merging vehicle at first, as shown in the plot "maneuver states" in Fig. 5.13. At the meantime, the alternative "pass" is declared as "available". However, the hesitant merging vehicle decreases its velocity even though it is still higher than that of the AD vehicle. We simulate a driver in the AD vehicle who captures the yield intention of the merging vehicle. A transition to *intended* pass is triggered at 39.3s (Fig. 5.13 (d)) by the driver's press on the "pass" button in the HMI. The AD vehicle begins to accelerate to reach the initial velocity again (Fig. 5.13 (a)). At 51.6s,

[21] A video of the simulation is available at https://youtu.be/pMhNti_p8SY.

the AD vehicle surpasses the merging vehicle and the latter merges behind. The simulation results show that the cooperation with the driver can help the AD system make an appropriate decision.

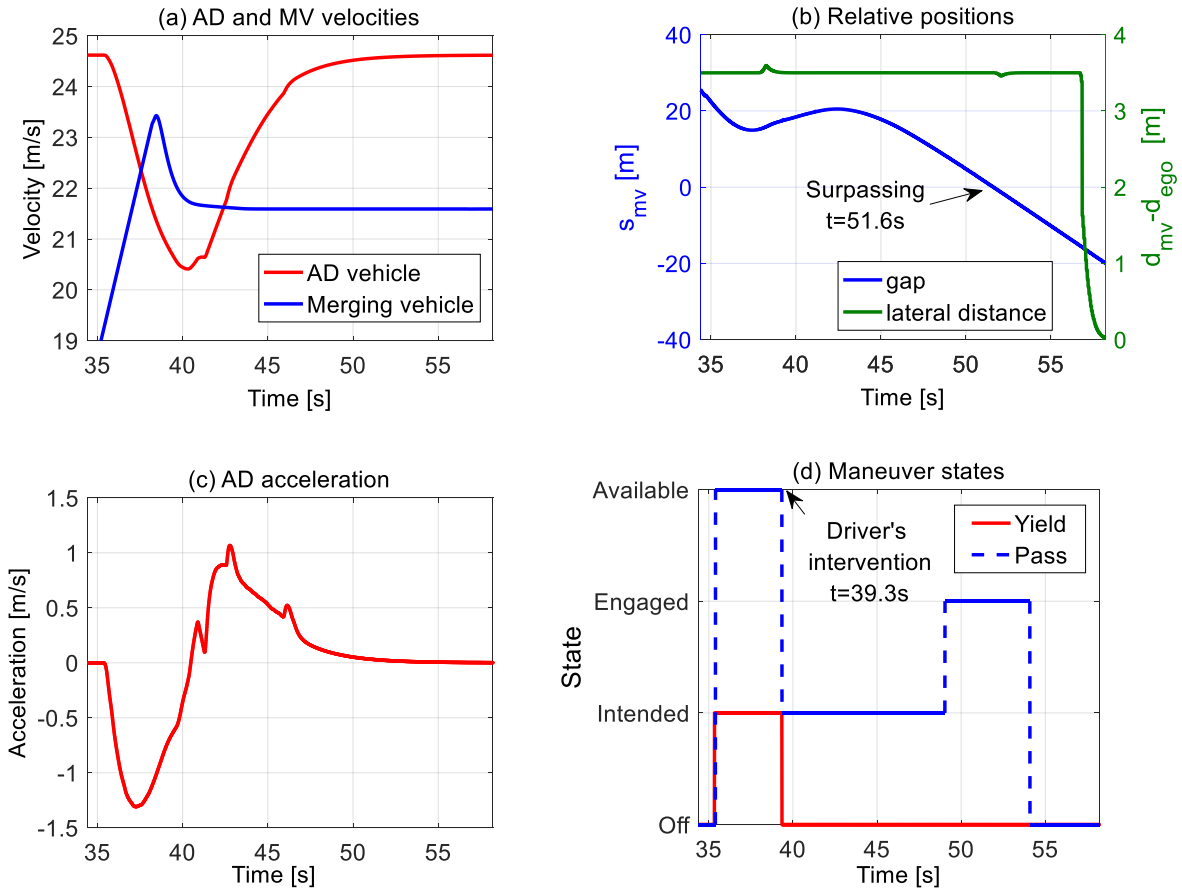


Figure 5.13. Simulation results with driver-vehicle cooperation in S2

5.7.2.3 S3-hesitant merging in congested traffic[22]

In S3, the AD vehicle drives in a congested traffic with an average speed around 4m/s. A merging vehicle arrives from behind and rests in a proximity of the AD vehicle (see Fig. 5.14 (a) and (b) around 50s). Based on the transition rule, both "pass" and "yield" maneuvers are declared "available". We simulate a driver who selects the "yield" at 51s (Fig. 5.14 (d)). To manifest its intention "yield", the AD vehicle slightly decelerates to increase the gap with the lead vehicle. When the lead vehicle detects the intention of the AD vehicle, it accelerates to merge in front starting from about 55s (Fig. 5.14 (a), around) without influencing the AD vehicle (e.g., forcing it to stop). Through this example, we demonstrate the potential of driver-vehicle cooperation in managing the social interaction with a human-driven vehicle.

[22] A video of the simulation is available at https://youtu.be/gQTG2vMn-RI.

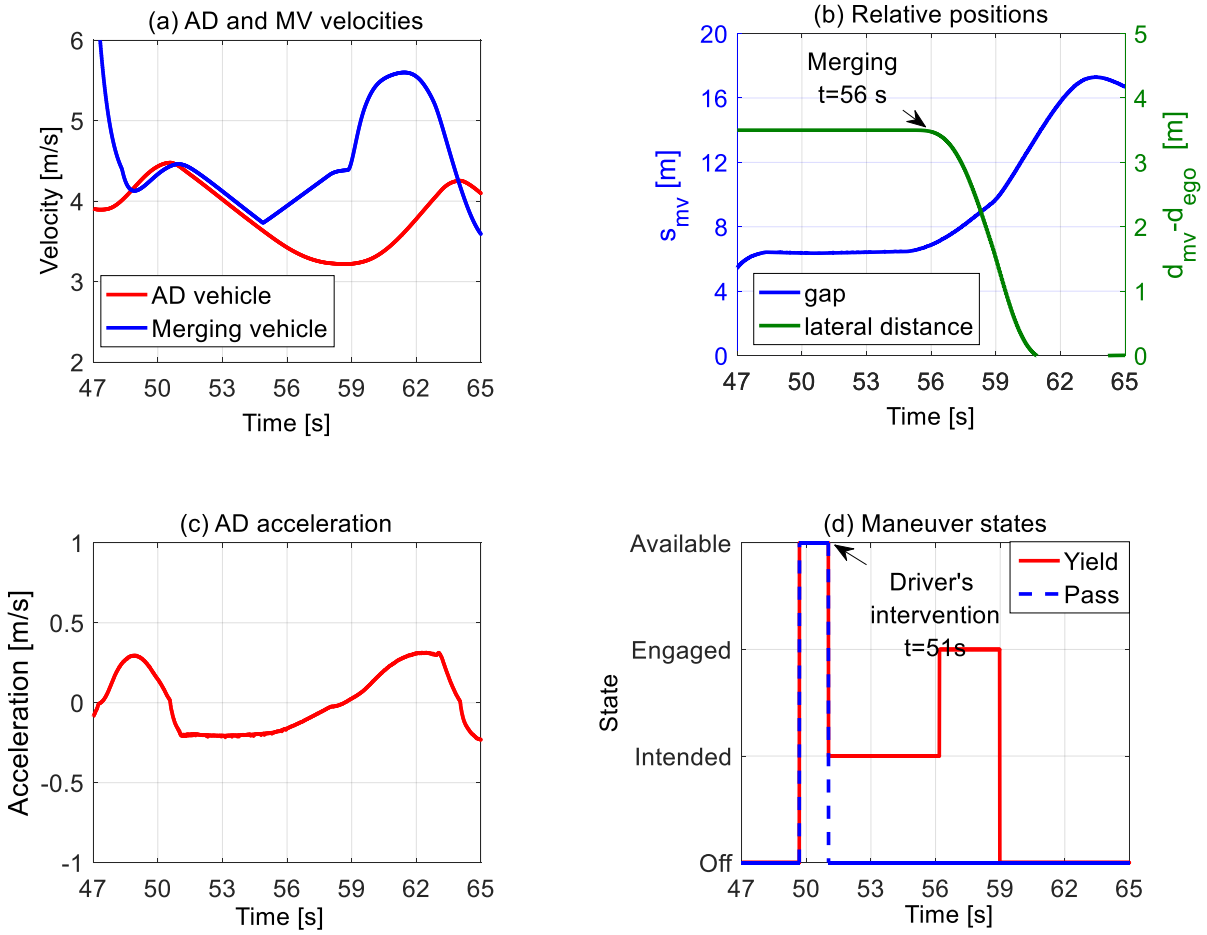


Figure 5.14. Simulation results with driver-vehicle cooperation in S3

5.8 User study

In the previous section, we have implemented the cooperative longitudinal control system in the simulation environment and obtained the preliminary results on system's performance. This section presents the evaluation of the principle for maneuver cooperation through a user study. This study is oriented to test user experience, hence a main objective of the user study is to investigate how future users will perceive the cooperation principle and HMI. What's more, this test examined whether cooperation facilitates the interaction of the AD vehicle with the merging vehicle in this use case.

5.8.1 Objectives

Within this user study, we intended to evaluate the following three aspects of the proposed cooperation principle:

- *Objective 1*: to evaluate the *intuitiveness* of the proposed cooperation principle and its HMI;
- *Objective 2*: to assess the *user's performance on cooperation* through the HMI;

- *Objective 3*: to assess the effects of cooperation on the *interaction between the AD vehicle and the merging vehicle*.

5.8.2 Methods

5.8.2.1 Participants

Twenty-two participants, average age of 41.3 years (ranging from 24 to 61) took part in the experiment. They were employees of Renault Technocentre and IRT SystemX. They have a driving license of 22.45 years on average and drove on average 5.8 days/week.

5.8.2.2 Apparatus

The driving simulator "Dr SiHMI" of the IRT SystemX ("Dr SiHMI Platform" 2016) was used in the experiment. The simulator uses SCANeR Studio as a simulation platform in which are integrated different modules such as the AD system, the scenario modelling tool and HMI controllers. The visual system of the simulator is composed of three projectors and a curved screen that can cover a field of view in horizontal 170° and vertical 40°. The simulator cockpit is modular and thus facilitates the prototyping and integration of HMI solutions.

We adapted two scenarios modelled in the simulation study (Section 5.7.1): hesitant merging in fluid traffic (referred to as Fluid) and in congested traffic (referred to as Congestion). According to the transition rules in the HFSM, the AD system has its intended maneuver (*intended pass* or *intended yield*) in Fluid, while in Congestion the AD system is in the *no intention* state. This implies that the subject has an alternative to choose in Fluid but two alternatives in Congestion. In both scenarios, the AD mode is active by default so that each subject can totally be disengaged from vehicle control. Fig. 5.15 illustrates two scans of driving scene in Fluid and Congestion.

Figure 5.15. Baseline scenarios: Fluid (left) and Congestion (right)

For this type of scenario in which the ego vehicle needs to interact with other traffic vehicles, it is of interest to vary behaviors of traffic vehicles. Otherwise, a same merging behavior across all the test runs could not only decrease the immersion of subjects but also lead to strong

memory effects that may bias the assessment of cooperation performance. As a solution, we used a random variable generator to generate random parameters for the scenario modelling tool. In this way, based on a baseline scenario (used as a template), different test scenarios can be generated automatically.

5.8.2.3 Procedure and instructions

Prior to the test drive, the subjects were familiar with the AD system and the driving simulator in a training drive. The test drive was divided into two phases. The objective of the Phase 1 was to evaluate the intuitiveness of the HMI and the cooperation principle. In this phase, each subject participated in three test runs. Each run consisted of a Fluid and a Congestion scenario. The first run served as a reference in which none of HMI was activated. Before this run, we explained briefly the functionality of the system (full automatic control) and asked the subject to observe the driving scene as a new user. In the second run, we added HMI displays (HUD and the yellow rectangle). We informed the subject of the new added HMI displays but did not explain their meaning. In the third run, we activated the button command interface in addition to HMI displays. We simply indicated the driver that he had a new interface allowing him to change the intention of the system, however, without any instruction on how to use the buttons. As illustrated by the schema in Fig. 5.16, the second run was set to assess how the subjects understood the HMI displays whilst the third run was dedicated to evaluate their comprehension of the cooperation principle. At the end of Phase 1, the subjects were asked to fill the prepared questionnaires.

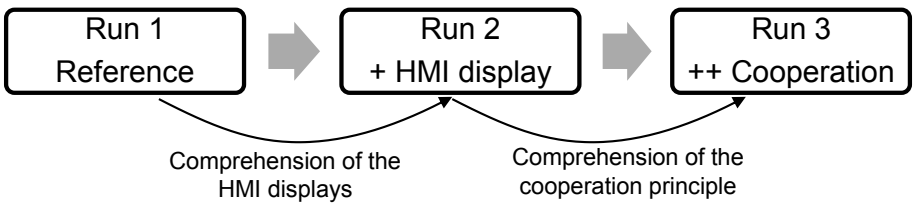


Figure 5.16. Schema of the procedure in Phase 1

The Phase 2 aimed to test subject's performance on cooperation. Therefore, HMI displays and the buttons were both active. Each subject participated in four sequences of Fluid and four sequences of Congestion in a random order. Before the start of Phase 2, we explained the HMI displays and the correct way to use buttons. Furthermore, each subject was instructed to use these buttons according to their needs.

5.8.2.4 Data collection and metrics

Both quantitative and qualitative data were collected. The quantitative data were collected through data log from the simulator and a questionnaire. To obtain the qualitative data, we

employed "thinking aloud" technique and conducted interviews. Results were principally derived from the quantitative analysis, while qualitative data were used to give complementary information.

To evaluate how subjects perceived the cooperation principle and their performance on cooperation, we proposed two metrics concerning button use. According to the button state when it was pressed, we defined three types of button pressing:

- **P**ressing **A**vailable **A**lternative maneuver (PAA): a subject pressed the button corresponding to an available alternative. It characterizes a good use.
- **P**ressing **I**ntended maneuver (PI): a subject pressed the button corresponding to system's intended maneuver. It can be interpreted that the subject shared the same intention with the system.
- **P**ressing **P**re-**I**ntention (PPI): a subject pressed a button before the system showed its intended maneuver (in Fluid) or available alternatives (in Congestion).

The first metric of button use was the distribution of button press types which indicates the successful rate (represented by the ratio of PAA).

The second metric was related to the time on button use. Within each record of a test run, by setting the time when an alternative became available as the origin (t=0), we computed at which time a subject pressed a button for the first time. The statistical information on the time of the first button press was exploited in different ways in Phase 1 and Phase 2. In Phase 1, this information allowed us to assess whether a subject was aware of the time window of an alternative's availability. In Phase 2, given that subjects had known the cooperation principle through our explanations, the average time they needed to press the button is a metric of their efficiency on cooperation.

Concerning Objective 3, we queried the data within a time span in which the AD vehicle was interacting with the merging vehicle. The time span is delimited by the duration when the state *highway merging management* was active. The selected metrics within this time span are shown in Tab. 5.3. The interaction duration is denoted by t_{inter}. Due to the limited range available for the merging vehicle on the acceleration lane, the longer t_{inter} is, the more urgent the merge of the merging vehicle is, and hence the more critical the situation is for the AD vehicle. The speed ratio reflects the ability of the AD system to maintain its initial speed during the interaction. At last, the acceleration variation is a metric for comfort.

Table 5.3. Metrics used to evaluate the driving performance

Metrics	Description
t_{inter}	Length of the time span in which the AD vehicle was interacting with the merging vehicle
v_{fin}/v_0	Ratio of the end speed of the AD vehicle to the initial speed
Δa	Difference between the maximum and the minimum accelerations of the AD vehicle

5.8.3 Results

5.8.3.1 Intuitiveness of the HMI and cooperation principle

Subjective evaluation of the HMI intuitiveness was made based on the results of the questionnaire (see Fig. 5.17). Among all the HMI elements, the simulated augmented reality, the yellow rectangle tracking the merging vehicle, has received the highest rating (M = 3.68, SD = 0.57) in terms of the ease of understanding. The representation of merging context in HUD-HMI was rated as "rather easy" (M = 2.95, SD = 0.90). The rather high SD indicates that the answers were varied. According to the verbal protocols, several subjects were confused with the meaning of arrows. They interpreted the blue arrows as actual accelerations of the ego vehicle. Due to the small size of the pictogram limited by the HUD projection area, 12 subjects out of 22 reported unaware of the green arrow. As for the button command, while the button function was better understood by subjects (M = 3.36, SD = 0.66), the color code of the button was worse rated (M = 2.73, SD = 1.03). Owning to the position of the button interface, several subjects did not associate the twinkling green state of a button with the appearance of the green arrow in the HUD-HMI. Consequently, those subjects were confused with the meaning of the green color.

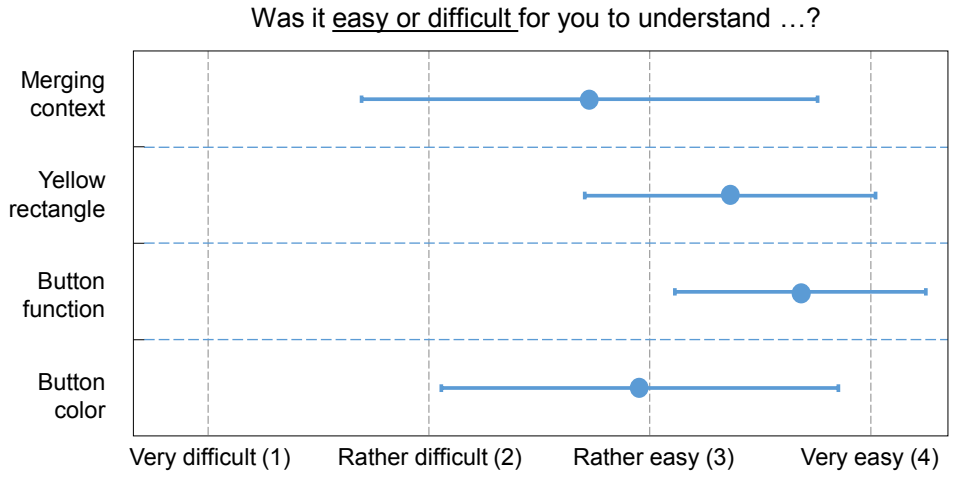


Figure 5.17. Subjective evaluation of the intuitiveness of the HMI

To assess how users perceived the cooperation principle, we first examined the distribution of the types of button pressing in Phase 1. As shown in Fig. 5.18, PAA accounted for only 21% of the presses in Fluid. It suggests that subjects did not well understand the logic to change the intention of the system. Rather, the high ratio of PI (50%) reveals that subjects expected that pressing system's intention would have effect. Compared to Fluid, the logic in Congestion was better perceived by the subjects given the 53% for PAA.

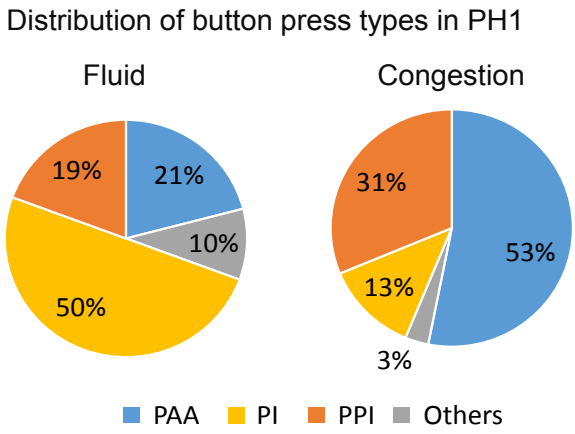


Figure 5.18. Results on button press distribution in Phase 1 (PH1)

The distribution of the time of the first button presses in Phase 1 is represented in form of the histogram in Fig. 5.19. In each histogram, the presses falling within the range of [-SD, SD] were counted and shown. The time window of the availability of the alternative was underlined in green and its initial time served as the origin of time axis. The average time of the first button press as well as the SD were indicated too. It can be clearly seen that the button presses in Fluid were more dispersed, whilst the presses in Congestion were more concentrated to the average value. In view of the one-third presses falling out of the time window of the "available pass" in Fluid, it can be inferred that subjects did not well perceive this time windows in Fluid.

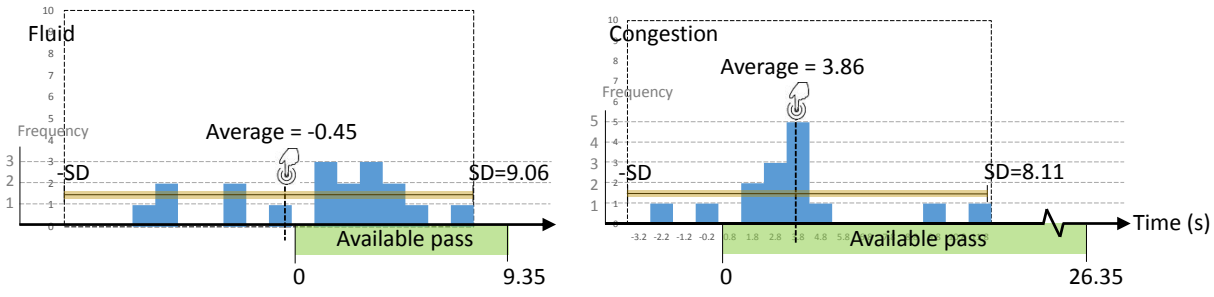


Figure 5.19. Histograms of the time of the first button press in Phase 1with the interval of 1s (left: Fluid; right: Congestion).

5.8.3.2 Cooperation performance

The overall performance of the subjects on button use was improved in Phase 2, given the 62% and the 86% for PAA in Fluid and Congestion respectively in Fig. 5.20. However, PI in Fluid

still occupied a non-negligible part (27%) compared with in Congestion. It indicates that several subjects had tendency to confirm the system's intention, even though they were informed of the correct button use—to press an available alternative. With a more important part of PAA, the performance in Congestion was better than in Fluid.

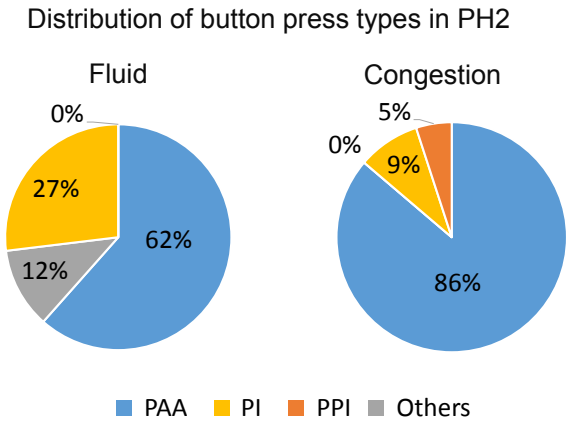


Figure 5.20. Results on button press distribution in Phase 2 (PH2)

Fig. 5.21 shows that the distributions of the time of the first button press in both scenarios of Phase 2 were more concentrated to the average than in Phase 1. What's more, most button presses fell into the time windows of the alternative's availability. It means that the subjects perceived better this time window and hence understood better the cooperation principle. Coincidently, the averages of both two scenarios were equal to 3.87s.

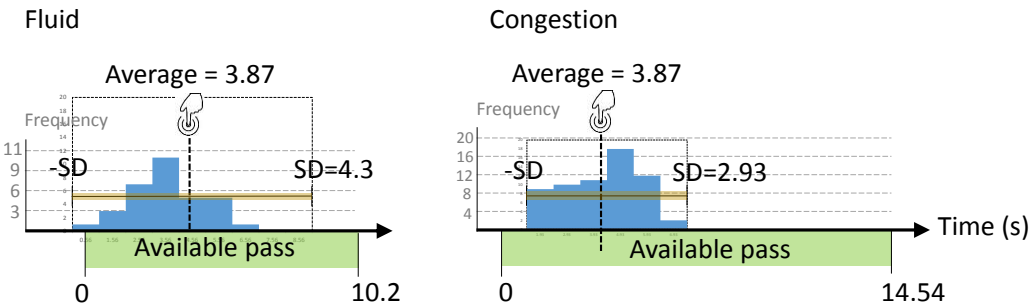


Figure 5.21. Histograms of the time of the first button press in Phase 2 with the interval of 1s (left: Fluid; right: Congestion).

5.8.3.3 The effects of cooperation on the interaction with the merging vehicle

Tab. 5.4 shows the descriptive statistics of the performance metrics of the AD system in interaction with the merging vehicle in Phase 2. The sample of system's performance without cooperation in either Fluid or Congestion consists of nine randomly generated scenarios; the sample of system's performance with cooperation (41 in Fluid and 59 in Congestion) is made of the test runs with effective cooperation, i.e., with a PAA, performed in these nine scenarios

of Fluid or Congestion. An independent-samples t-test was conducted to compare system's performance with and without cooperation (see Tab. 5.5). In Fluid, all the performance metrics yielded better results with cooperation than without cooperation, manifested by shorter interaction time (t_{inter}, p < 0.03), higher speed-keeping ability (v_{fin}/v_0, p < 1.19E-11) and smaller acceleration variation (Δa, p < 0.04). In Congestion, t_{inter} with cooperation was significantly reduced (p < 5.56E-04). The ratio of the final speed to the initial speed (v_{fin}/v_0) with cooperation was higher, however, without significant difference (p < 0.06). The average acceleration variation was stronger with cooperation in Congestion. It can be explained by the fact that the AD vehicle manifested its intention to the merging vehicle by small amounts of acceleration because of the driver's intervention.

Table 5.4. Means and standard deviations for the metrics on the performance of the AD system in Phase 2

Source	Number	Metrics					
		t_{inter} (s)		v_{fin}/v_0 (%)		Δa (m/s^2)	
	n	*M*	*SD*	*M*	*SD*	*M*	*SD*
Fluid – without cooperation	9	20.87	5.52	69.26	5.01	2.05	0.77
Fluid – cooperation	41	16.04	3.00	98.18	9.04	1.40	0.64
Congestion – without cooperation	9	19.63	7.00	78.70	15.46	0.88	0.29
Congestion – cooperation	59	7.17	4.04	90.47	20.45	1.28	0.29

Table 5.5. T-test for equality of means (independent samples)

Scenario	Metrics	*t*	*df*	*p**	Mean differences
Fluid	t_{inter} (s)	-2.54	9	0.03	-4.83
	v_{fin}/v_0 (%)	13.23	21	1.19E-11	28.92
	Δa (m/s^2)	-2.38	11	0.04	-0.65
Congestion	t_{inter} (s)	-5.21	9	5.56E-04	-12.46
	v_{fin}/v_0 (%)	2.03	13	0.06	11.77
	Δa (m/s^2)	3.85	11	2.68E-3	0.40

* Two tails with a 5% alpha level

5.8.4 Summary and discussion

We summarize the main results from the user study and give a discussion after each result.

Concerning the HMI, *the simulated augmented reality in the driving scene—the yellow rectangle that tracks the merging vehicle was considered as the one the easiest to understand among all the HMI elements*. Besides other factors that could influence the intuitiveness of an HMI like graphical grammar, we want to highlight the role of the perceptibility, considering that the HMI is dispatched into three areas in the field of view of the driver in the current configuration. According to our observations, most of the subjects monitored the merging vehicle located on the right part of the scene. The verbal protocol suggested that they tended to infer the system's intention from the change of the distance to the merging vehicle. Consequently, the HUD-HMI (in the center of the scene) and the back-lit buttons (near the gear shift lever) were situated in the peripheral vision of those subjects. The low perceptibility of the HUD-HMI and the button interface may influence subject's average level of understanding on the meanings of their contents. Furthermore, subjects needed to switch attention between the merging vehicle and the button command (to check whether a button became green). It could increase their cognitive demand. Thus, it may be beneficial for the usability of HMI to display the essential information in the driving scene, especially in the zone the driver attends to. In our case, it is of interest to show the system's intention and the button's state near the merging vehicle. To achieve this goal, the augmented reality technology is necessary.

With regard to the cooperation principle, the experiment results indicate that *the logic in Congestion (being possible to choose any of the alternatives) was easier to understand than that in Fluid (being possible to choose an alternative other than the system's intention)*. The average performance of the subjects on button use in Phase 2 confirms this conclusion too. Therefore, it seems better to provide a way for the driver just to indicate his intention, regardless of the intention of the system. In case that the driver and the system share the same intention, the system can take the input of the driver as a confirmation. This modification of the cooperation principle will be tested in the future works. As to the temporal aspect of the cooperation performance, the average time needed for the first button press after this button became available was about 3.9s. This implies that the system needs to offer a time window long enough (at least longer than 3.9s under the current HMI configuration) so that the driver has enough time to reason and to give his choice. Of course, time window length depends on the behavior of the merging vehicle, but it can be extended if the AD system is able to predict the situation's evolution with a longer look ahead. Therefore, this imposes high requirements on the situation assessment function of the AD system. On the other hand, how to design an intuitive HMI to reduce the driver's reaction time, especially when he is engaging in non-driving tasks, remains another research question of interest.

The performance metrics of the AD vehicle in interaction with the merging vehicle suggest that *the intervention from the driver could be beneficial to the AD system in terms of managing merging situations*. Nevertheless, it is difficult to generalize the same conclusion to the real world for lack of the knowledge of the validity of these simulated scenarios. As argued by Boer et al. (2015), to evaluate the interaction of the AD vehicle with other road users constitutes one important usage of driving simulators in the AD vehicle design. To achieve this goal, it needs to ensure the realism of the behaviors of other road users. One plausible solution is to integrate state-of-the-art microscopic traffic simulation models into a scenario modelling tool as the one developed in this study. Given that those models are already validated by naturalistic data, they can enhance the realism of behaviors of traffic vehicles in the simulation. In the meanwhile, the traffic vehicles can interact with the ego vehicle according to the scenario.

5.9 Conclusion

In this chapter, we have designed a cooperative longitudinal control framework following the principle for maneuver cooperation. This framework was designed in a use case concerning highway merging management. At the tactical level, we have developed a HFSM-based cooperative maneuver planning function. The states and the transitions in the HFSM were designed *a priori* based on human driving behaviors in the same kind of situation. The HFSM adopts different strategies in uncongested and congested traffic flows. In uncongested traffic, the system explicitly manifests its intention to pass or yield by assessing the current or future gap with the merging vehicle. In congestions, the system can socially interact with the merging vehicle through the cooperation with the human driver. In order to execute a maneuver generated from the tactical level, we adapted an ACC controller with the virtual leader scheme. Particularly, we have proposed a MPC-based trajectory generation method for the virtual leader. By formulating trajectory generation as an online optimization problem, this method is capable of generating a smooth trajectory towards a target state.

For the interaction aspect of the design framework, we have designed a set of HMI solutions for cooperation at the tactical level. In a user study on driving simulator, we evaluated the intuitiveness of the cooperation principle, user's performance on cooperation and the effects of the cooperation on AD vehicle's interaction with the merging vehicle. Test results show the interest of using augmented reality to enhance the perceptibility of HMI. In addition, we discussed user's perception of the proposed cooperation principle and the positive effect of driver-vehicle cooperation on AD vehicle's performance. We also pointed out some future directions to improve the principle for maneuver cooperation and the associated HMI.

6 PRINCIPLE FOR CONTROL COOPERATION: PREDICTIVE SHARED STEERING CONTROL

6.1 Introduction

In the previous chapter, we designed a cooperative longitudinal control system which allows the driver to change system's maneuver plan at the tactical level. This chapter focuses on the problem of control sharing between the driver and the automation at the operational level. This problem is addressed by the principle for control cooperation.

Considering potential interferences between the driver and the automation on vehicle's lane positions or lateral trajectories, we identify highway lane positioning and lane changing as a use case for control cooperation. In this use case, we aim to design a cooperative steering control system. The system performs automatic lane-keeping control in AD mode. When the human driver intervenes in the control-loop, the system shares the control authority with the driver following the cooperation principle. While the shared control occurs at the operational level, the system adapts the target lane for steering control at the tactical level once the driver's steering intention is detected.

We begin this chapter by introducing the use case in Section 6.2. The architecture for the cooperative steering control system is presented in Section 6.3. Section 6.4 describes how to formulate a haptic shared steering control problem in the MPC framework. Section 6.5 presents a tactical-level function named *active lane-change assist* that assists the driver in a lane change maneuver by detecting his intention. Section 6.6 presents a simulation study to demonstrate the

system's functions. It is followed by Section 6.7 which reports a preliminary user study evaluating the driver's steering performance during the control transition.

6.2 Use case: highway lane positioning and lane changing

Drivers may adjust lane positions of their vehicles for many reasons. For example, it is often observed that a car that is passing a large truck purposely deviates from the lane center to increase the lateral distance to the truck for the reason of safety. In low-speed traffic in France, some drivers tend to adjust their vehicles' positions towards the road curb when they see a motorcycle coming in the middle of the road in their side mirrors. In this case, by adjusting their lane positions, the car drivers communicate to the motorist that they are considerate of him. In general, lane-keeping control systems keep the vehicle driving in the center of a lane. A more advanced approach for vehicle guidance is to use a motion planner whose major objective is plan trajectories to avoid collisions in the road. It is difficult, however, to formulate these context-based objectives into a single motion planning framework. Therefore, a driver may have needs to adjust vehicle's lane position in some situations if the AD vehicle always drives in the center of a lane or occasionally swerves to avoid potential collisions.

A lane change maneuver can be regarded as a special case of lane positioning, because lane change literally means changing the vehicle's position into another lane. But lane change means more than vehicle trajectory control, it also involves decision-making process at the tactical level. From the perspective of HMI, it is of interest to study a unified cooperation framework through which the driver cannot only control vehicle's lane position at the operational level but also indicate his intention of lane change to the automation. For this reason, we address highway lane positioning and lane changing in a single use case.

Based on field observations, mainline vehicles sometimes make cooperative lane changes to facilitate the merging of on-ramp vehicles at highway entry section. The solution proposed for this use case can also be applied for the use case "highway merging management" in Chapter 5. In addition to indicate his intention to pass or yield to a merging vehicle, the driver can initiate a lane change through the cooperation principle implemented in this use case.

6.3 System architecture and assumptions

In this use case, we focus on vehicle lateral motion control. Our design objective is hence *a cooperative steering control system*. The system architecture stems from the hierarchical cooperative control architecture proposed in Chapter 3. The information flows through the cooperation principle are highlighted in Fig. 6.1. The purple arrows show that the cooperation

arises at the operational level when the driver intervenes through the control interface. Following the principle, the cooperation can also ascend to the tactical level when the system adapts its maneuver plan to the driver's control.

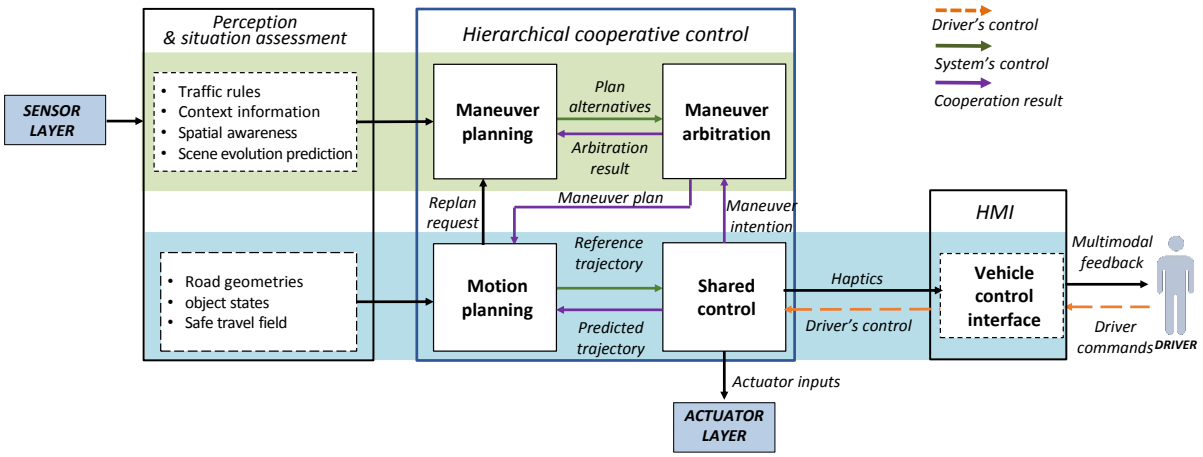


Figure 6.1. Information flows in the hierarchical cooperative control architecture.

Fig. 6.2 shows the architecture of the cooperative steering control system. At the operational level, we propose a single *predictive shared steering control* function that combines motion planning and shared control in the MPC framework. The rationale for this proposition is twofold. Firstly, the main task of the operational-level functions is to perform lane keeping control in this use case. We assume that a high-precision digital map provides information on road geometry based on clothoid curves for the controller to track (Section 4.3.2). It is a reasonable assumption for highway application in that there is almost no intersection on a highway. Secondly, MPC has the potential to merge the trajectory planning and control into a unified framework. By formulating a finite horizon constrained optimal problem, one can use a MPC controller to keep the vehicle within a safe navigational zone while satisfying input constraints, safety constraints, and ride comfort preferences. This concept has been explored in the works of Anderson et al. (2010), Erlien, Fujita, and Gerdes (2016) and Suh et al. (2016).

At the tactical level, an *active lane change assist* function can automatically perform a lane change maneuver if the driver's lane change intention is detected from his actions on the steering wheel[23]. The lane-keeping or lane change maneuver is translated as a specific lane center for predictive shared steering control. The situation assessment function developed in Chapter 4 provides the states of traffic vehicles for the active lane change assist to assess the

[23] Other methods to initiate a lane change maneuver, e.g., via the turn signal are also compatible with this architecture.

feasibility of a lane change maneuver at the tactical level. In the meantime, it sends the road information to the predictive shared controller.

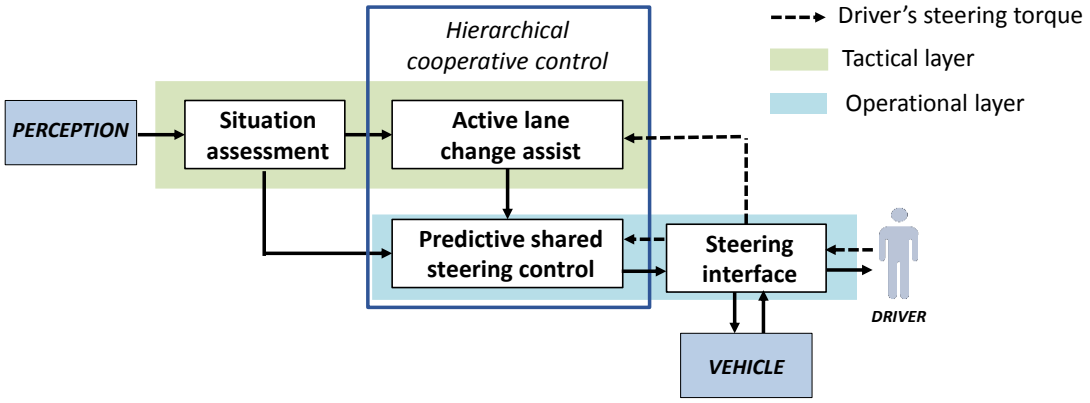


Figure 6.2. Architecture of the cooperative steering control system

The following assumptions are given within the defined use case:

- The AD vehicle operates in normal driving conditions. Concerning the ego vehicle dynamic states, normal conditions imply that they are within a stable handling envelope. Thus, those critical conditions, e.g., in evasive maneuvers (large lateral acceleration) and low road surface friction are not considered. As to the external environment, normal conditions means fluid traffic with steady flux speed. All the traffic vehicles drive within their lanes.
- It is assumed that the AD vehicle's speed is constant in the prediction horizon of the MPC. Even though the open-loop prediction accuracy decreases under this assumption, the update of the prediction model based on the current speed measure in the online optimization can at some extent compensate the prediction error. As it will be demonstrated by the simulation results, the MPC controller is robust to the speed variation.
- All the states of the vehicle model are available and the uncertainty on the states are omitted, thus the predictive shared steering control is formulated in a deterministic framework.
- The vehicle longitudinal control framework developed in Chapter 5 is used to ensure automatic longitudinal control. During a lane change maneuver, the virtual leader used by the ACC controller will join the target front vehicle in the target lane to generate a smooth trajectory for longitudinal control.

Concerning the final authority issue in the control sharing, we follow the assumption made in Chapter 3 that the driver has the final authority in normal driving conditions. We implement a

mode transition rule proposed in the French project "ABV" on vehicle automation (Section 2.5.5). Following this transition rule, the fact that the driver's steering torque exceeds a threshold reveals a strong disagreement between the driver and the automation. In this case, the automation gives all the steering control authority back to the driver by disengaging steering control function. Accordingly, the level of the AD system transitions from SAE level 3 to level 1 where the system assumes only vehicle longitudinal control. The driver shall be notified from the HMI of the transition.

6.4 Predictive shared steering control

6.4.1 Previous works on MPC for automated driving and shared steering control

MPC based approaches have been applied both in trajectory planning and tracking for automated driving. In trajectory planning applications, a mass point model or a kinematic bicycle model is usually used as vehicle model for trajectory prediction (Carvalho et al. 2015). A major design concern is how to formulate a safe navigational zone in terms of state constraints. To make powerful convex optimization tools applicable, it is desirable to formulate a convex navigational zone. Andersen et al. (2010) and Erlien, Fujita, and Gerdes (2016) sampled the boundaries of a presumed safety region to generate constraints on lateral deviations. Nilsson et al. (2015) approximated a non-convex driving corridor by intersecting two polyhedrons. Park et al. (2015) and Qian et al. (2016) incorporated non-convex collision-free constraints into a mixed-integer programming problem (MIQP). Whereas Park et al. used a cell-sequence approach to decompose a non-convex homotopy, Qian et al. combined propositional logic with approximated convex zones to formulate a MIQP.

MPC has also been applied for vehicle steering control to track a reference trajectory. Dynamic vehicle models with tire models are mostly used in this kind of application to predict more precisely vehicle dynamic responses to control inputs. The complexity of using a dynamic model resides in the nonlinear relationship between the tire forces and the vehicle states and inputs. Falcone et al. (2007) developed a full nonlinear MPC (NMPC) for active steering control, however, this NMPC scheme suffers from high computational burdens. In their later works (Falcone et al. 2007), the nonlinear vehicle model was linearized at each current state, thus the computational efficiency was improved. With a linear tire model, the model of vehicle lateral dynamics can be transformed to a linear-parameter varying (LPV) model where the vehicle longitudinal speed v_x is a time varying parameter. Besselmann and Morari (2009) proposed an explicit LPV-MPC approach to exploit this LPV structure. In their approach, the

explicit control law was computed via a closed-loop min-max MPC algorithm based on dynamic programming.

As presented in Section 2.4.1, MPC has been explored to accommodate shared control scheme. Andersen et al. (2013) blended the MPC's and driver's steering commands according to the wheel slip angle which is considered as a metric of vehicle's stability. Erlien et al. (2016) defined a safe driving envelope consisting of a stable handling envelope for vehicle's stability and an environment envelope for collision avoidance. A MPC controller seeks to match the driver's commands if the current driver command allows for a vehicle trajectory within the safe envelope. It intervenes only if such trajectory does not exist. The MPC controllers proposed in these two works controlled steering angles and were implemented in the steering-by-wire infrastructure. In order to make the driver aware of system's activities, haptic feedback was rendered separately from the control loop in both works.

In this thesis, we designed a MPC-based haptic shared control framework. In contrast with the works of Andersen et al. and Erlien et al. which controlled steering angle within steering-by-wire system, the MPC controller in this framework controls steering torque for a conventional steering system with mechanical connections. In this way, the driver can directly feel the controller's steering torque from the steering wheel. Furthermore, this framework exploits the receding horizon strategy in the MPC scheme to adapt system's control to driver's actions. Some shared control frameworks (Saleh et al. 2013; Soualmi et al. 2014) formulated a fixed optimization objective—a compromising between reference tracking quality and control sharing for an infinite horizon and synthetized a control law once offline. The proposed framework iteratively resolves an online finite-horizon optimization problem that takes into account the driver's actual control. At the price of not guaranteeing the global optimality of the solution, the receding horizon strategy updates the solution based on the actual conditions which are usually dynamic.

6.4.2 Problem formulation

For convenience, we restate the optimization problem to be solved by a MPC controller:

$$\text{Minimize}: J_{N_p}\big(\mathbf{x}(0), U(0)\big) \tag{6.1a}$$

Subject to:

$$\dot{\mathbf{x}}(k+1) = f\big(\mathbf{x}(k), \mathbf{u}(k)\big), \quad k = 0, \dots, N_p - 1, \tag{6.1b}$$

$$\mathbf{x}(k) \in \mathcal{X}, \quad k = 0, \dots, N_p - 1, \tag{6.1c}$$

$$\mathbf{x}\big(N_p\big) \in X_f, \tag{6.1d}$$

$$u(k) \in \mathcal{U}, \quad k = 0, \dots, N_c - 1. \tag{6.1e}$$

The main design task is to formulate a haptic shared steering control problem in the form of (6.1). In practice, the cost function J_{N_p} (6.1a) is formulated as a compromising between two competing objectives—small output deviation and small control efforts (Boyd 2008). This compromising can be represented by the following equation:

$$J_{N_p} = \rho J_{out} + J_{in}, \tag{6.2}$$

where J_{out} is the cost on the output deviation with small values characterizing "good" regulation and control; J_{in} denotes the cost on inputs with small values for "using small *control authority*"; ρ gives the relative weighting between J_{out} and J_{in}. It is clear that ρ determines the tradeoff between those two objectives.

From the point of view of HMI, ρ can be used to leverage the allocation of the control authority between the automation and the human driver. The smaller ρ is, the more the MPC controller tolerates the deviation from the reference and the more it penalizes using control authority (J_{in}). In the inverse case, when ρ is bigger, the automation will employ more control authority to penalize the deviation from its proper reference. Thanks to the online optimization property of MPC, we can adapt ρ online such that the automation dynamically shares the control authority to the human driver.

In addition to the weight ρ in the cost function, the constraints in (6.1) can also be exploited in the shared control scheme. The constraints can be set on AD vehicle's lateral positions to define a safe navigational zone. If the driver steers the vehicle towards a lane position out of this zone, the MPC will enforce its control to constrain the vehicle's trajectory within this zone. At the same time, the driver can feel the increased steering torque from the controller. In this way, we tend to achieve a design objective in the proposed principle—to show disagreement if the driver's action could undermine the safety.

6.4.3 Vehicle model

This section presents a vehicle model (6.1b) that takes into account vehicle lateral dynamics, steering system dynamics and vehicle lane positioning.

6.4.3.1 Bicycle model of vehicle lateral dynamics

Bicycle model is a simplified vehicle dynamics model which was firstly reported in the work of Riekert and Schunck (1940). Thanks to its good representativeness of vehicle lateral dynamics, it is widely used in applications of vehicle lateral control. Since it assumes symmetry

dynamic behavior between left and right wheel sides, the structure of vehicle can be simplified as a single track with two wheels shown in Fig. 6.3. Assuming that the longitudinal speed v_x is constant, the bicycle model has two degrees of freedom. In the vehicle fixed frame Λ^V, vehicle's lateral motion is governed by

$$ma_y + mv_x\dot{\psi} = F_{yf} + F_{yr}, \tag{6.3}$$

where F_{yf} and F_{yr} are the total lateral tire forces of the front and rear wheels respectively. Moment balance about the CoG yields the equation for the yaw dynamics as

$$I_z\ddot{\psi} = l_f F_{yf} - l_r F_{yr}, \tag{6.4}$$

where I_z denote the yaw inertia, and l_f and l_r are the distances of the front tire and the rear tire respectively from the CoG of the vehicle. The lateral tire forces $F_{y\star}$ ($\star \in \{f, r\}$) are given by

$$F_{y\star} = C_\star \alpha_\star, \tag{6.5}$$

where $C_\star$ stands for the tire cornering stiffness, and $\alpha_\star$ represents the lateral tire slip angle which is obtained using small angle approximations as,

$$\alpha_f = \delta_f - \frac{v_y + l_f\dot{\psi}}{v_x}, \tag{6.6}$$

$$\alpha_r = \frac{-v_y + l_r\dot{\psi}}{v_x}, \tag{6.7}$$

where δ_f is the front steering angle. Experimental results show that the linear relationship $F_{y\star} - \alpha_\star$ keeps well when $\alpha_\star$ is below 3° (equivalent lateral acceleration a_y of 0.4g) according to Mitschke and Wallentowitz (2004).

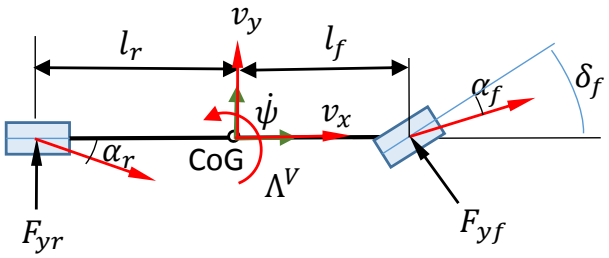


Figure 6.3. Bicycle model notion

6.4.3.2 Steering system model

Fig. 6.4 shows a mechanical model of an *electric power assisted steering* (EPAS) system based on which we implemented haptic shared control. In manual driving mode, the electric motor generates an assist steering torque to aid the driver to perform steering maneuver. In automatic steering control, the AD system controls the electric motor to apply an *overlay torque command*

T_{ctrl} on the steering column in the place of the driver to steer the vehicle. In case of the driver's intervention, T_{ctrl} is superposed to the steering torque of the driver T_d. The steering system dynamics in haptic shared control is the same with that in EPAS. To simplify the model, the torsion of the steering column and the friction in the steering system are omitted. The dynamics of the steering system is modelled as

$$I_{eq}\ddot{\delta}_f + c_{eq}\dot{\delta}_f = i_s(T_d + T_{ctrl}) - \frac{T_{al}}{k_p}, \tag{6.8}$$

where I_{eq} is the equivalent moment of inertia of the steering system, c_{eq} is the equivalent damping, i_s is the steering ratio, T_{al} is the total torque of auto-alignment of two front wheels, k_p is the steering assist coefficient which is a simplified representation of the electric motor's assistant steering torque. In this study, k_p is assumed to be constant. The torque of auto-alignment is mainly due to the tire lateral force F_{yf} and therefore can be approximated by

$$T_{al} = F_{yf}n_f = F_{yf}(n_R + n_k), \tag{6.9}$$

where n_R is the offset between the lateral force acting point and the center of tire contact patch, and n_k is the offset due to the caster of the steering axis (see Fig. 6.5). The method to calculate I_{eq} and c_{eq} is given in Appendix B.

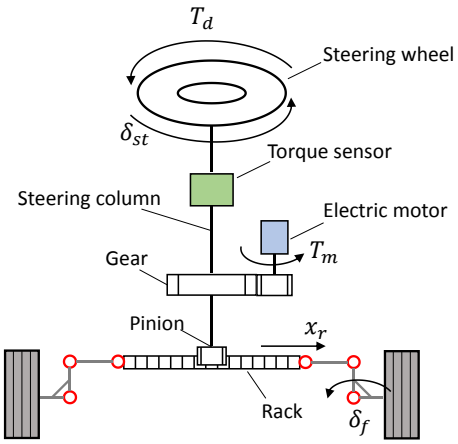


Figure 6.4. Mechanical model of an EPAS system

Equations (6.3)-(6.9) characterize the lateral vehicle dynamics under the controls of the driver and the automation. The state space model of the lateral dynamics based on (6.3)-(6.9) can be written as

$$\dot{\mathbf{x}}_{dyn} = A_{dyn}\mathbf{x}_{dyn} + B_{dyn}\mathbf{u}, \tag{6.10}$$

where $\mathbf{x}_{dyn} = \begin{bmatrix} v_y & \dot{\psi} & \delta_f & \dot{\delta}_f \end{bmatrix}^T$ is the state vector,

$$A_{dyn} = \begin{bmatrix} -\frac{C_f+C_r}{mv_x} & -v_x - \frac{C_f l_f - C_r l_r}{mv_x} & \frac{c_f}{m} & 0 \\ -\frac{C_f l_f - C_r l_r}{I_z v_x} & -\frac{C_f l_f^2 + C_r l_r^2}{I_z v_x} & \frac{c_f l_f}{I_z} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{C_f n_f}{I_{eq} v_x} & \frac{C_f l_f n_f}{I_{eq} v_x} & -\frac{C_f n_f}{I_{eq}} & -\frac{c_{eq}}{I_{eq}} \end{bmatrix}, \tag{6.11}$$

$$B_{dyn} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{i_s}{I_{eq}} \end{bmatrix}. \tag{6.12}$$

In full automatic control, $\mathbf{u}$ is the control of the automation, i.e., $\mathbf{u} = T_{ctrl}$, while in shared control, $\mathbf{u}$ is the total control of both the automation and the human driver, i.e., $\mathbf{u} = T_d + T_{ctrl}$.

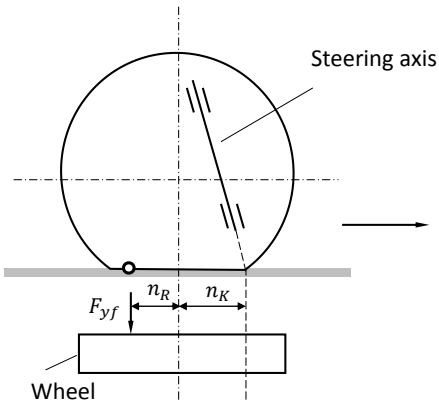


Figure 6.5 Schema used to calculate the torque of auto-alignment

6.4.3.3 Lane positioning

The main objective of the MPC controller is to track a lane centerline. Since the AD vehicle's position is represented in the Frenet frame Λ^{RD} in the scene representation, its lateral offset to a reference lane center Δd and heading error $\Delta\psi$ can be trivially computed. As shown in Fig. 6.6, Δd and $\Delta\psi$ can be obtained via

$$\begin{cases} \Delta d = d - d_{ref} \\ \Delta\psi = \psi - \psi_{ref} \end{cases}, \tag{6.13}$$

where d_{ref} is the normal coordinate of the reference lane center in the Frenet frame. This reference lane is AD vehicle's current lane (Lane 2 at t_1 in Fig. 6.6) in lane-keeping and the target lane (Lane 1 at t_2) in a lane change maneuver.

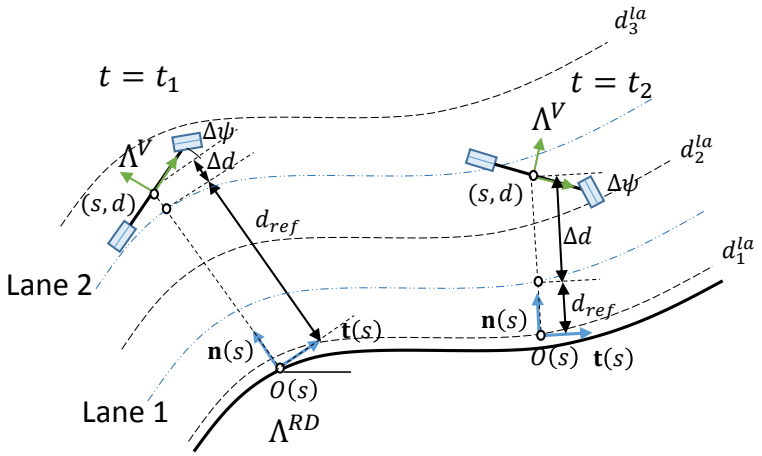


Figure 6.6 AD vehicle's lane position in the local map of the world model developed in Chapter 4

The derivatives of Δd and $\Delta\psi$ are given by

$$\begin{cases} \Delta\dot{d} = v_x \sin(\Delta\psi) + v_y \cos(\Delta\psi) \\ \Delta\dot{\psi} = \dot{\psi} - \kappa(s)\dot{s} \end{cases}, \tag{6.14}$$

where $\kappa(s)$ is the curvature of the road and $\dot{s}$ is the velocity of the projection of the CoG along the road curve which is obtained as

$$\dot{s} = \frac{v_x \cos(\Delta\psi)}{1-\kappa(s)d}, \tag{6.15}$$

Under the small-angle approximation and neglecting the small terms, (6.14) and (6.15) can be simplified as

$$\Delta\dot{d} \approx v_x \Delta\psi + v_y, \tag{6.16}$$

$$\dot{s} \approx v_x. \tag{6.17}$$

In a curved road, $\kappa(s)$ will change as the AD vehicle moves in the prediction horizon. Since road geometries in the digital map are composed of straight segments, circle arcs, and clothoids, the curvature evolution can be expressed by

$$\kappa(s) = \kappa_0 + c_m s, \tag{6.18}$$

where κ_0 is the curvature at vehicle's current position and c_m is the curvature changing rate. The derivative of $\kappa(s)$ with time can be computed by

$$\dot{\kappa} = \frac{d\kappa(s)}{dt} = c_m \dot{s} \approx c_m v_x, \tag{6.19}$$

An issue arises when the AD vehicle crosses different curve segments. In this case, c_m will change stepwise and cause discontinuity in (6.19). To solve this problem, an averaging curvature model proposed by Dickmanns (1988) is used. The basic idea is to use a first-order

system to filter the steps caused by c_m. The filter parameters can be derived by making the lateral deviation of the real road and that of the model equal at a given look ahead distance L. This model is expressed as

$$\dot{c}_{m1} = \frac{3v_x}{L}(c_{m1} - c_{mL}), \tag{6.20}$$

where c_{m1} is the average curvature rate, L is a look-ahead distance, and c_{mL} is the curvature rate at L which can be retrieved from the digital map by range-querying. Interested readers can refer to the work of Dickmanns (2007, 219–22) for derivation details.

The state-space form of the averaging curvature model based on (6.19) and (6.20) is given as

$$\dot{\mathbf{x}}_{cur} = A_{cur}\mathbf{x}_{cur}, \tag{6.21}$$

with the state vector $\mathbf{x}_{cur} = [\kappa \quad c_{m1} \quad c_{mL}]^T$ and the matrix

$$A_{cur} = \begin{bmatrix} 0 & v_x & 0 \\ 0 & -\frac{3v_x}{L} & \frac{3v_x}{L} \\ 0 & 0 & 0 \end{bmatrix}. \tag{6.22}$$

6.4.3.4 State-space representation of the whole model

The state-space model of the plant for the prediction in MPC can be obtained by concatenating (6.10) representing the lateral dynamics, (6.14) and (6.16) for lane positioning and (6.22) for road curvatures

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}, \tag{6.23}$$

where $\mathbf{x} = \begin{bmatrix} v_y & \dot{\psi} & \delta_f & \dot{\delta}_f & \Delta\psi & \Delta d & \kappa & c_{m1} & c_{mL} \end{bmatrix}^T \in \mathbb{R}^9$. The state matrix $A \in \mathbb{R}^{9\times9}$ can be partitioned as

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}, \tag{6.24}$$

with $A_{11} = A_{dyn} \in \mathbb{R}^{4\times4}$, $A_{12} = \boldsymbol{O}_{4\times2}$, $A_{13} = \boldsymbol{O}_{4\times3}$,

$$[A_{21}|A_{22}|A_{23}] = \left[\begin{array}{cccc|cc|ccc} 0 & 1 & 0 & 0 & 0 & 0 & -v_x & 0 & 0 \\ -1 & 0 & 0 & 0 & -v_x & 0 & 0 & 0 & 0 \end{array}\right],$$

$A_{31} = \boldsymbol{O}_{3\times4}, A_{31} = \boldsymbol{O}_{3\times2}$, and $A_{33} = A_{cur} \in \mathbb{R}^{3\times3}$. Note that $\boldsymbol{O}_{m\times n} \in \mathbb{R}^{m\times n}$ denotes the null matrix. The input matrix B is given as

$$B = \begin{bmatrix} \boldsymbol{O}_{1\times3} & \frac{i_s}{I_{eq}} & \boldsymbol{O}_{1\times5} \end{bmatrix}^T \in \mathbb{R}^9. \tag{6.25}$$

The meaning of $\mathbf{u}$ is the same as in (6.10). Since the variable v_x is a parameter of the matrix A, (6.23) is a non-linear model. Assuming that v_x remains constant in the prediction horizon, (6.23) is a linear time invariant system. Since the MPC is formulated in the discrete time framework, (6.23) is discretized using the zero-order hold method.

6.4.4 Constraints

We impose the constraints on some vehicle's states and the control input to meet the design specifications in terms of safety and passenger's riding comfort.

6.4.4.1 Constraints on AD vehicle's lateral offset to the lane center

The AD vehicle's position needs to be constrained in a safe navigational zone for collision avoidance. In this use case, we use *lane-based constraints* as a simplified form. The outermost lane borders are used as default lateral constraints. As illustrated in Fig. 6.7 (a), this placement allows the driver to adjust vehicle's trajectory or change lane in a large space. However, if an adjacent lane is occupied by vehicles and merging in this lane may result in collisions with in-lane vehicles, the constraint is temporally set on the border of the occupied adjacent lane, as showed in Fig. 6.7 (b). The decision for constraint placement is made during the assessment of the lane change feasibility in the active lane change assist. The corresponding algorithm will be presented in Section 6.5.2.

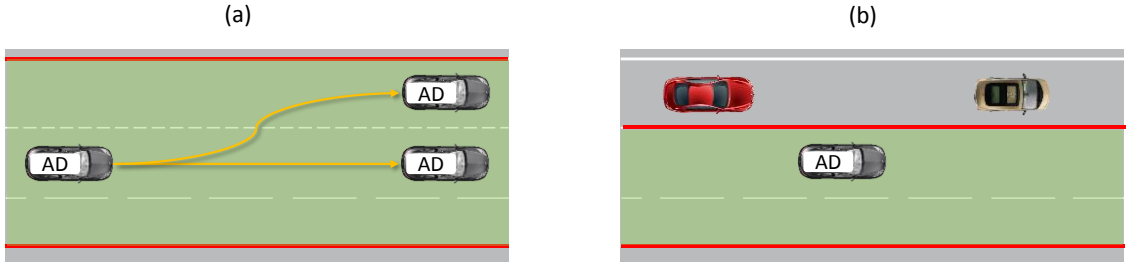


Figure 6.7 Lane-based constraints on vehicle's lateral offset: (a) the constraints are placed on the outermost lane borders to prevent road departure; (b) if the merge in an adjacent lane causes potential dangers, the border to this lane is set as constraint.

The constraints on lateral offset to the lane center are represented as

$$\Delta d_{min} - \epsilon \leq \Delta d(k) \leq \Delta d_{max} + \epsilon, \quad k = 0,1,\dots,N_p - 1, \tag{6.26}$$

where Δd_{min} and Δd_{max} are the lower and upper bounds, ϵ is slack variable which "softens" the constraints to ensure the feasibility of the optimisation problem. According to the definition of the lane-based constraints, Δd_{min} and Δd_{max} are given by

$$\begin{cases}\Delta d_{min} = -\frac{w_{la}-w_{ego}}{2} - (1-\gamma_r)d_r^{la} \\ \Delta d_{max} = \frac{w_{la}-w_{ego}}{2} + (1-\gamma_l)d_l^{la}\end{cases}, \tag{6.27}$$

where w_{la} and w_{ego} are lane width and the ego vehicle's width, d_r^{la} and d_l^{la} are the distances of the current lane center to the rightmost and leftmost lane center respectively. The two binary variables γ_r and γ_l are given by the assessment function for the lane change feasibility at the tactical level. For example, $\gamma_r = 0$ means a right lane change is feasible, thus the constraint on the minimum lateral offset is loosened. In the contrary, $\gamma_r = 1$ enforces this constraint.

Remark

The lane-based constraints are proposed for the designed use case which assumes that all the traffic vehicles drive in their lanes. This method cannot guarantee a collision-free control solution in a general highway driving situation, e.g., the avoidance of a stalled obstacle in between lanes cannot be handled by setting constraints on the lane border. A generic method is first to approximate a safe corridor by using geometries, e.g., using cell decomposition techniques (Anderson et al. 2013; Park, Karumanchi, and Iagnemma 2015), then to sample the boundaries of the corridor in the prediction horizon (usually assuming constant longitudinal speed) to generate two dynamic constraint vectors:

$$\begin{cases}\Delta\mathbf{d}_{min}(k) = \begin{bmatrix}\Delta d_{min}(k) & \Delta d_{min}(k+1) & \cdots & \Delta d_{min}(k+N_p-1)\end{bmatrix}^T \\ \Delta\mathbf{d}_{max}(k) = \begin{bmatrix}\Delta d_{max}(k) & \Delta d_{max}(k+1) & \cdots & \Delta d_{max}(k+N_p-1)\end{bmatrix}^T.\end{cases} \tag{6.28}$$

Evidently, (6.28) can be integrated in our MPC framework to yield a more generic solution.

6.4.4.2 Constraints on ride comfort

Normal driving conditions in terms of vehicle dynamics can be characterized by small lateral accelerations. Based on field experimental results, Wegschweider and Prokop (2005) identified a zone delimited by longitudinal and lateral accelerations in the "g-g graph" to define a normal driving type (Fig. 6.8). Based on this zone, we constraint the maximal lateral acceleration as:

$$|a_y| = |\dot{\psi} v_x| \leq 0.4g, \tag{6.29}$$

therefore,

$$|\dot{\psi}| \leq \dot{\psi}_{max} = \frac{0.4g}{v_x}. \tag{6.30}$$

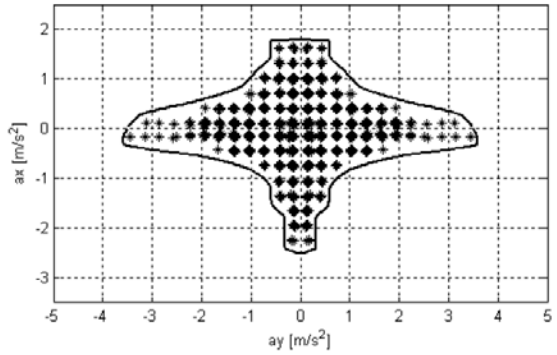


Figure 6.8 "G-g" graph that depicts normal driving conditions (Wegschweider and Prokop 2005)

6.4.4.3 Constraints on control input

In steady state corning, the steering torque on the steering column can be computed as

$$T_{str} = \frac{mn_f l_r}{k_p i_s (l_f + l_r)} a_y. \tag{6.31}$$

Given that a_y is limited by 0.4g in normal driving conditions, the necessary maximum steering torque in our case is

$$T_{str} = \frac{mn_f l_r}{k_p i_s (l_f + l_r)} (0.4g) \approx 5.2\ (Nm). \tag{6.32}$$

We thus define the steering torque limit T_{max} as 6 Nm

$$|T_{ctrl}| \leq T_{max}. \tag{6.33}$$

Note that the steering torque bounded by T_{max} can always be overruled by a human driver. Considering the actuator limit, the increment of T_{ctrl} is bounded as

$$|\Delta T_{ctrl}| \leq \Delta T_{max}. \tag{6.34}$$

With the sample size of 0.05 s, ΔT_{max} is set as 0.5 Nm.

6.4.5 Cost function and shared control policy

The main control objective of the MPC controller is to track a reference lane center. A performance vector for reference tracking is defined as

$$\mathbf{z} = C\mathbf{x} = \begin{bmatrix} v_y & \dot{\psi} & \dot{\delta}_f & \Delta\psi & \Delta d \end{bmatrix}^T, \tag{6.35}$$

with $C = (c_{i,j}) \in \mathbb{R}^{5\times 9}$ in which the values of $c_{1,1}$, $c_{2,2}$, $c_{3,4}$, $c_{4,5}$ and $c_{5,6}$ are equal to one and the rest is null. Following (6.2), the cost function in this optimization problem is defined as

$$J_{N_p}(\mathbf{x}(0), U(0), \rho) = \sum_{k=0}^{N_p-1} \mathbf{z}(k)^T \rho Q \mathbf{z}(k) + \mathbf{u}(k)^T R_u \mathbf{u}(k) + \Delta\mathbf{u}(k)^T R_{\Delta u} \Delta\mathbf{u}(k) + S\epsilon^2, \quad (6.36)$$

where $Q \in \mathbb{R}^{5\times5}$, $R_u \in \mathbb{R}$ and $R_{\Delta u} \in \mathbb{R}$ represent diagonal weighting matrices penalising deviations from $\mathbf{z}(k) = \mathbf{u}(k) = \Delta\mathbf{u}(k) = \mathbf{0}$, S denotes the penalty on constraint violations in (6.26), and $\rho \in [0,1]$ is a dynamic weighting factor that leverages the control allocation between the controller and the driver. It is referred to as *shared control policy* hereinafter.

Based on the cooperation principle, the first objective for the shared control is to ensure that the driver can override the automation with ease. Therefore, when the driver's steering intention is detected, ρ is set to zero. In this way, whenever the vehicle's lateral trajectory satisfies the lateral position constraints (6.26), the MPC controller minimizes its control to give the driver the most control authority. When the driver releases the steering wheel, ρ is set to one. The MPC controller then takes over the control to ensure a seamless control transition. The variation of ρ is expressed as:

$$\begin{cases} \rho = 0, & if\ \gamma_{sw} > 0 \ \wedge\ |T_{dr}| > T_{thre} \\ \rho = 1, & if\ \gamma_{sw} = 0 \ \vee\ |\gamma_{ma}| = 1 \end{cases}, \quad (6.37)$$

where γ_{sw} is the output of a steering wheel hand position sensing system, T_{thre} is a positive threshold to detect the driver's steering intention and γ_{ma} is the maneuver state in the active lane change assist function. To detect the driver's presence in the control-loop, we first employ a steering wheel hand position sensing system to detect the driver's hand contact on the steering wheel. $\gamma_{sw} > 0$ means that at least one hand is detected and $\gamma_{sw} = 0$ means hands off. However, a driver may put his hands on the steering wheel while loosely following the rotation of the steering wheel. In order to detect the steering intention of the driver, a threshold T_{thre} is set on the driver's steering torque. Enache et al (2010) used a threshold 2 Nm to discriminate an "inattentive" driver ($|T_{dr}|$< 2 Nm) from an "attentive" driver in lane keeping scenarios. In our case, we define T_{thre} as 1 Nm based on the recorded data in a preliminary simulator test. In order to hand over the control to the controller, the driver needs to release the steering wheel ($\gamma_{sw} = 0$). Another case in which the controller actively resumes the control authority is to aid the driver to perform a lane change maneuver ($|\gamma_{ma}| = 1$). This case will be presented in Section 6.5.

The shared policy ρ is then followed by a first-order filter with time lag t_ρ to generate a smooth variation between zero and one.

6.4.6 MPC formulation

The following constrained finite-time optimal control problem is formulated based on the development in the previous sections:

$$\text{Minimize: } J_{N_p}(\mathbf{x}(0), U(0), \rho) = \sum_{k=0}^{N_p-1} \mathbf{z}(k)^T \rho Q \mathbf{z}(k) + \mathbf{u}(k)^T R_u \mathbf{u}(k) + \Delta\mathbf{u}(k)^T R_{\Delta u} \Delta\mathbf{u}(k) + S\epsilon^2 \quad (6.38a)$$

Subject to:

$$\dot{\mathbf{x}}(k) = A\mathbf{x}(k) + B(k), \quad k = 0, \dots, N_p - 1, \quad (6.38b)$$

$$\Delta d_{min} - \epsilon \leq \Delta d(k) \leq \Delta d_{max} + \epsilon, \quad k = 0,1, \dots, N_p - 1, \quad (6.38c)$$

$$\left|\dot{\psi}(k)\right| \leq \dot{\psi}_{max}, \quad k = 0,1, \dots, N_p - 1, \quad (6.38d)$$

$$|T_{ctrl}(k)| \leq T_{max}, \quad k = 0, \dots, N_c - 1 \quad (6.38e)$$

$$|\Delta T_{ctrl}(k)| \leq \Delta T_{max}, \quad k = 0, \dots, N_c - 1. \quad (6.38f)$$

At each time instant, the optimal control problem is solved online based on the current vehicle state vector $\mathbf{x}(0)$, the shared control policy ρ, dynamic constraints Δd_{min} and Δd_{max}. Only the first element of the obtained optimal control sequence is applied as T_{ctrl} on the steering system.

6.5 Active lane change assist

6.5.1 Architecture

The internal architecture of active lane change assist and the interfaces with other modules are shown in Fig. 6.9. It consists of three sub-functions:

- Lane change feasibility assessment: based on the situation assessment, this module assesses feasibilities of left and right lane changes. The results are sent to maneuver generation and to predictive shared control (into dynamic lane-based constraints).
- Driver's lane change intention detection: this module detects the driver's lane change intention based on the divergence between the AD vehicle's trajectory and the current lane centerline cause by the driver's steering action. Its outputs are two event variables to trigger maneuver state transitions in maneuver generation.
- Maneuver generation: based on the lane change feasibility and the driver's intention detection, this module updates the reference lane center and the shared control policy for predictive shared control.

The following sections will present these three modules subsequently.

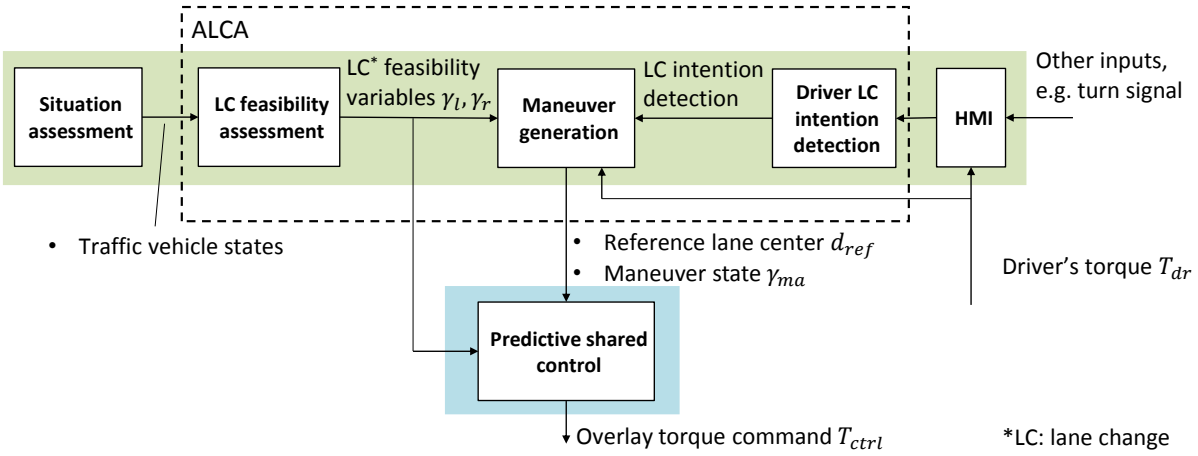


Figure 6.9 Architecture of the active lane change assist (ALCA) and its relation with other functions (steering interface is omitted for clarity)

6.5.2 Lane change feasibility assessment

The method to assess if a lane change is feasible used in this thesis stems from the concept of Lane Change Decision Aid Systems (LCDAS) which warn the ego vehicle driver of potential collisions with other vehicles in the adjacent lane during lane change maneuvers. Nowadays LCDAS are already on market and are specified in the international standard (ISO 17387 2008). The matrix form scene representation facilitates this assessment (Fig. 6.10).

The method is presented in an example of assessing the feasibility of a left lane change maneuver. The method begins with checking if an adjacent lane exists in the digital map. If positive, the method continues to check the occupancy of "left same level" case (case 2). If this case is occupied, i.e., a traffic vehicle exists on the side of the ego vehicle, an immediate lane change is declared infeasible. If case 2 is empty, the system examines case 3. If there is a rearward vehicle in case 3, the danger of a lane change maneuver is measured by the time-to-collision (TTC) metric between the ego vehicle and this rearward vehicle. The TTC formula is

$$TTC = \frac{s_i}{v_{ego} - v_i}, \tag{6.39}$$

where s_i is the relative distance of a vehicle i to the ego vehicle and v_i is this vehicle's speed. Some studies based on naturalistic lane change data recommended a warning signal threshold on TTC between 2 to 6 seconds (Lee, Olsen, and Wierwille 2004; Wakasugi 2005). In our work, this threshold is defined as 3 second, i.e., if the TTC with the vehicle in case 3 is smaller than 3 second, the left lane change is considered infeasible, and the flag γ_l in lane-based constraint is set to one.

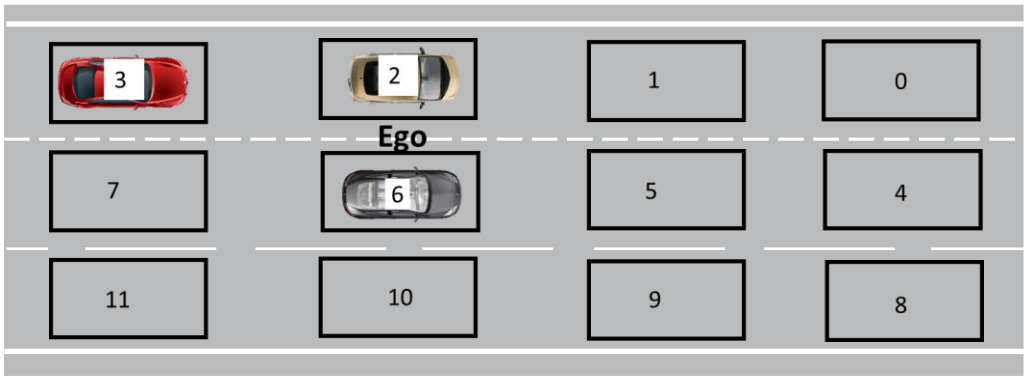


Figure 6.10 Matrix representation for lane change feasibility assessment

6.5.3 Driver's lane change intention detection

Many features can be used in the algorithm for driver's lane change intention detection, such as driver's behaviour (steering wheel angle, head movement), vehicle state (speed acceleration, heading, yaw rate) and environment state (lateral and heading errors to a road, relative distances with other traffic vehicles). In order to estimate a relation between the unobservable driver's intention and some observable features, learn-based approaches are often used, e.g., training a *support vector machine* classifier (Mandalia and Salvucci 2005; Kumar et al. 2013) or training a hidden Markov model (Berndt, Emmert, and Dietmayer 2008; Kuge et al. 2000). Those learned-based approaches were demonstrated to be effective, however, to decide pertinent features and to prepare a representative dataset are not trivial.

We adapted the method proposed by Houenou et al. (2013) to detect the driver's lane change intention. The idea is to monitor the lateral and heading deviations of the position of the AD vehicle to the lane center. Since the AD system performs lane keeping control by default, these deviations should be small. If these deviations to the lane center surpass a threshold and the driver is steering toward the direction to increase this deviation, then the driver's lane change intention detection is declared. The advantage of this method resides in the limited features used for the detection and its intuitiveness. A disadvantage of this method is its sensitivity to the threshold placed on the detector. In thesis, this method is also applied for the detection of the merging vehicle's lane change maneuver in Chapter 5.

At each time instant k, the following quadratic error function is used to measure the divergence between the AD vehicle center position and the lane center:

$$D(k) = w_d \Delta d(k)^2 + w_\psi \Delta \psi(k)^2, \tag{6.40}$$

where w_d and w_ψ are the weights. In the probabilistic framework, the covariance matrices of the noises of $\Delta d(k)$ and $\Delta\psi(k)$ need to be integrated in (6.40). In order to avoid disturbance due to punctual erroneous measurements, the fading memory sum of $D(k)$ is used:

$$D^\rho(k) = \rho D^\rho(k-1) + D(k), \tag{6.41}$$

with ρ denoting fading memory factor.

The detection of driver's left or right lane change intention is signified by a flag δ_l or δ_r. Driver's left lane change intention is detected, i.e., $\delta_l = 1$ if

$$(\gamma_{ma}(k) = 0) \wedge (D^\rho(k) > D_0) \wedge (D^\rho(k) > D^\rho(k-1)) \wedge (T_{dr}(k) > T_{thre}), \tag{6.42}$$

where the condition $\gamma_{ma}(k) = 0$ means the system's current maneuver state is lane keeping. D_0 is a threshold whose value needs to be tuned experimentally. The condition $T_{dr}(k) > T_{thre}$ implies that the driver intends to turn the steering wheel to the left. Correspondingly, driver's right lane change intention is detected, i.e., $\delta_r = 1$ if

$$(\gamma_{ma}(k) = 0) \wedge (D^\rho(k) > D_0) \wedge \left(D^\rho(k) > D^\rho(k-1)\right) \wedge (T_{dr}(k) < -T_{thre}). \tag{6.43}$$

We tuned the parameters in the above described algorithm based on a dataset collected from our previous driving simulator experiments on highway driving. There were totally 36 lane change maneuvers in this dataset. The retained parameters are showed in Tab. 6.1.

Table 6.1 Tuned parameters for driver's lane change intention detection

Symbol	Description	Value [units]
w_d	Weight on lateral deviation	1
w_ψ	Weight on heading (yaw) angle deivation	3000
ρ	Fading memeory factor	0.2
D_0	Threshold on the divergence mertic	3

During the parameter tuning, the focus was laid on the threshold D_0. We proposed three metrics to evaluate the performance of the algorithm—detection rate, false alarm rate and prediction time. The definitions of these metrics are illustrated in Fig. 6.11. The performance metrics with different threshold values are listed in Tab. 6.2 in which the selected value is enclosed.

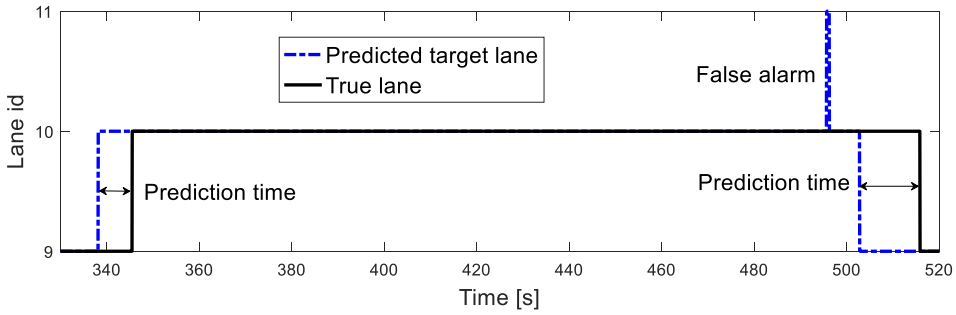


Figure 6.11 Illustration of the proposed performance metrics

Table 6.2 Detection performance metrics under different threshold values

Threshold value	Detection rate	False alarm rate	Average prediction time
1	100%	35.3%	2.7 s
2	100%	10.5%	2.15 s
3	100%	2.63%	1.65 s
4	100%	2.63%	1.45 s
5	89.2%	5.71%	1.2 s
6	78.3%	0 %	1.05 s

6.5.4 Maneuver generation

The maneuver generation function is modelled as a finite state machine which has three lateral maneuver states—lane keeping, left and right lane changes. The current maneuver state is denoted by a discrete variable $\gamma_{ma}(k) \in \{-1,0,1\}$, with "$-1$" for left lane change, "0" for lane keeping and "1" for right lane change. Given the symmetry between the right and the left lane changes, here only the transitions between left lane change and lane keeping are presented. The transition from lane keeping to left lane change, i.e., $\gamma_{ma} = 0 \rightarrow \gamma_{ma} = -1$, is triggered, if the driver's left lane change intention is detected and in the meanwhile the left lane change is considered as feasible. This condition is expressed as

$$\delta_l = 1 \wedge \gamma_l = 0. \tag{6.44}$$

Once the left lane change state is active, d_{ref} for predictive shared control is updated to the lateral offset of the left adjacent lane.

The transition from left lane change to lane keeping can be triggered under two different conditions. The first condition corresponds to a nominal termination of lane change maneuver, while the second condition represents that the driver counter steers the steering wheel to cancel

the triggered lane change maneuver, e.g., due to a false alarm. These two conditions are expressed as:

$$\Big((D^{\rho}(k) < D_0) \wedge \big(D^{\rho}(k) < D^{\rho}(k-1)\big)\Big) \vee (T_{dr} \cdot T_{ctrl} < -\lambda), \tag{6.45}$$

where $T_{dr} \cdot T_{ctrl}$ is an indicator of the negative interference between the driver and the system which was proposed by Sentouh et al (2013), and λ is a positive threshold which is set as 3.5 $N^2.m^2$.

As shown in (6.37), when the lane change maneuver is active ($|\gamma_{ma}| = 1$), the automation resumes the full steering authority. From the viewpoint of HMI, it implies that once the driver initiates a lane change, he can release the steering wheel and the automation performs it automatically.

6.6 Simulation study

This section presents the results of two simulation examples. We firstly performed a computer simulation study to demonstrate the performance of the controller for automatic lane keeping and lane change maneuvers. The second driver-in-the-loop simulation addressed a scenario in which the predictive shared steering controller rendered haptic resistance on the steering wheel to warn the driver of the danger of a lane change maneuver.

The MPC problem (6.38) was modelled by Yalmip (Lofberg 2004) in Matlab and solved by the solver Gurobi (Gurobi Optimization, Inc. 2017) at each time step. In the first example, the controller was connected in closed loop with a high fidelity nonlinear four-wheel vehicle model developed by Renault in Simulink environment. In the second example, the controller was integrated in the same driving simulation environment used in a user study (which will be presented in details in Section 6.7). In both examples, the MPC controller ran at 20Hz. The parameters of the vehicle model and the controller in (6.38) are given in Tab. 6.3.

Table 6.3 List of the parameters used in the MPC-based shared steering controller

Sysetm	Symbol	Description	Value [units]
Vehicle model	m	Total mass of vehicle	2024 [kg]
	I_z	Yaw moment of inertia of vehicle	2800 [kg.m^2]
	l_f	Longitudinal distance from CoG to front tires	1.29 [m]
	l_r	Longitudinal distance from CoG to rear tires	1.6 [m]
	C_f	Cornering stiffness of front tires	85000 [N]

	C_r	Cornering stiffness of rear tires	111380 [N]
	i_s	Steering ratio	16.3
	n_f	Trail length	0.052 [m]
	I_{eq}	Equivalent moment of inertia of the steering system	0.02 [kg.m^2]
	c_{eq}	Equivalent damping of the steering system	5.79 [N/rad/s]
	k_p	Ratio of steering assistance	4
	L	Lookahead distance in the averaging curvature model	37.5 [m]
MPC	N_p	Time sample number in the prediction horizon	30
	N_c	Time sample number in the control horizon	30
	T	Time sample size	0.05 [s]
	Q	Weight matrix on tracking performance	[4.5, 500, 5, 1400, 45]
	R_u	Weight on control	0.5
	$R_{\Delta u}$	Weight on control rate	200
	S	Weight on slack variable	10000
	T_{max}	Maximum steering torque	6 [N.m]
	ΔT_{max}	Maximum steering torque increment	0.5 [N.m]

6.6.1 Example: automatic steering control

This simulation example consists of two scenarios. In the first scenario, the predictive shared steering controller performed a lane change maneuver at 90km/h. The second scenario was to test the lane keeping performance of the controller in a curved road with a minimum radius of roughly 330m. To test the robustness of the controller against the speed variation, the AD vehicle accelerated and decelerated according to a speed profile during the simulation.

Fig. 6.12 shows the simulation results of a lane change performed by the controller. The simulation results suggest that the MPC is capable to perform a smooth lane change maneuver with small control inputs in normal driving situations.

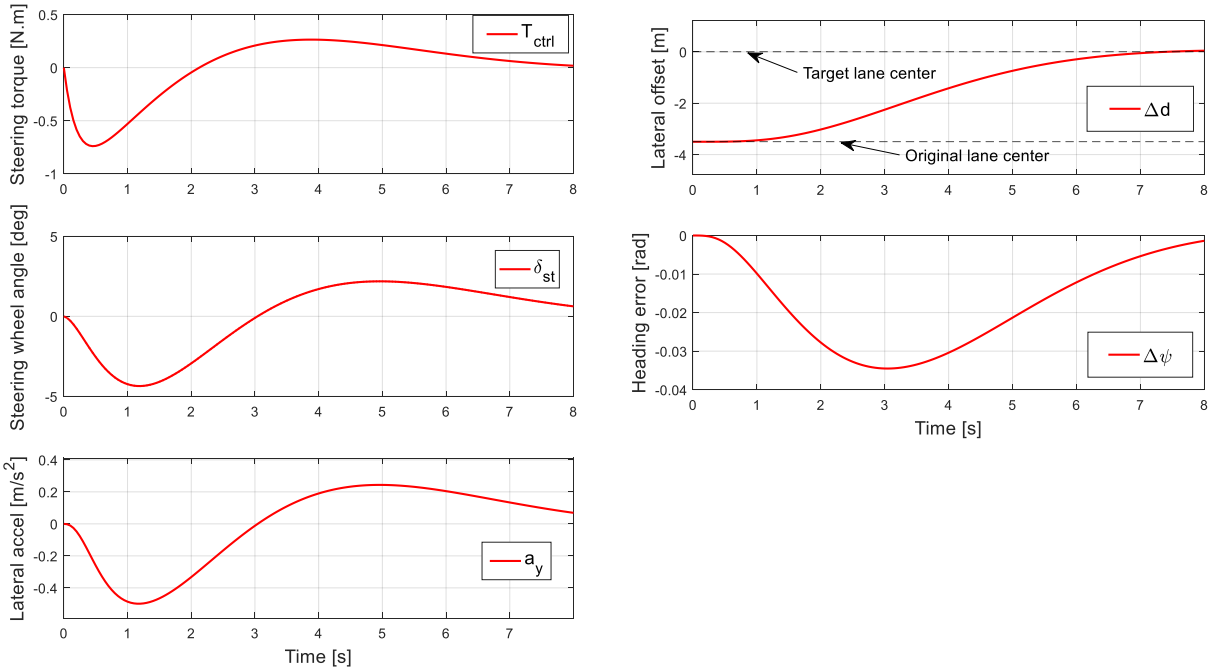


Figure 6.12 Simulation results for a lane change maneuver

Fig. 6.13 and Fig. 6.14 illustrate the performance of the MPC controller to track a curved road with the variation of the longitudinal speed. Even though the prediction model assumes a constant longitudinal speed in the prediction horizon, the MPC updates the linear prediction model based on the new speed measurement at each time instant to compensate prediction errors. As shown by the curve of the steering torque of the MPC in Fig. 6.14, the controller reduced the control as the speed decreased in the phase P1, maintained the control as the speed remained constant in P2, and augmented the control when the vehicle accelerated in P3, and finally maintained the control when the speed became stable again. Meanwhile, the controller kept a good tracking performance (small lateral offsets and heading errors).

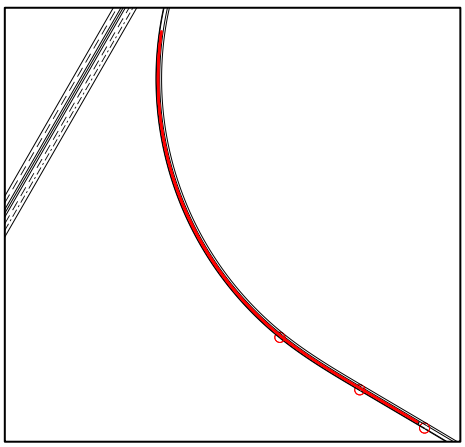


Figure 6.13 Trajectory realized by the MPC controller during the simulation

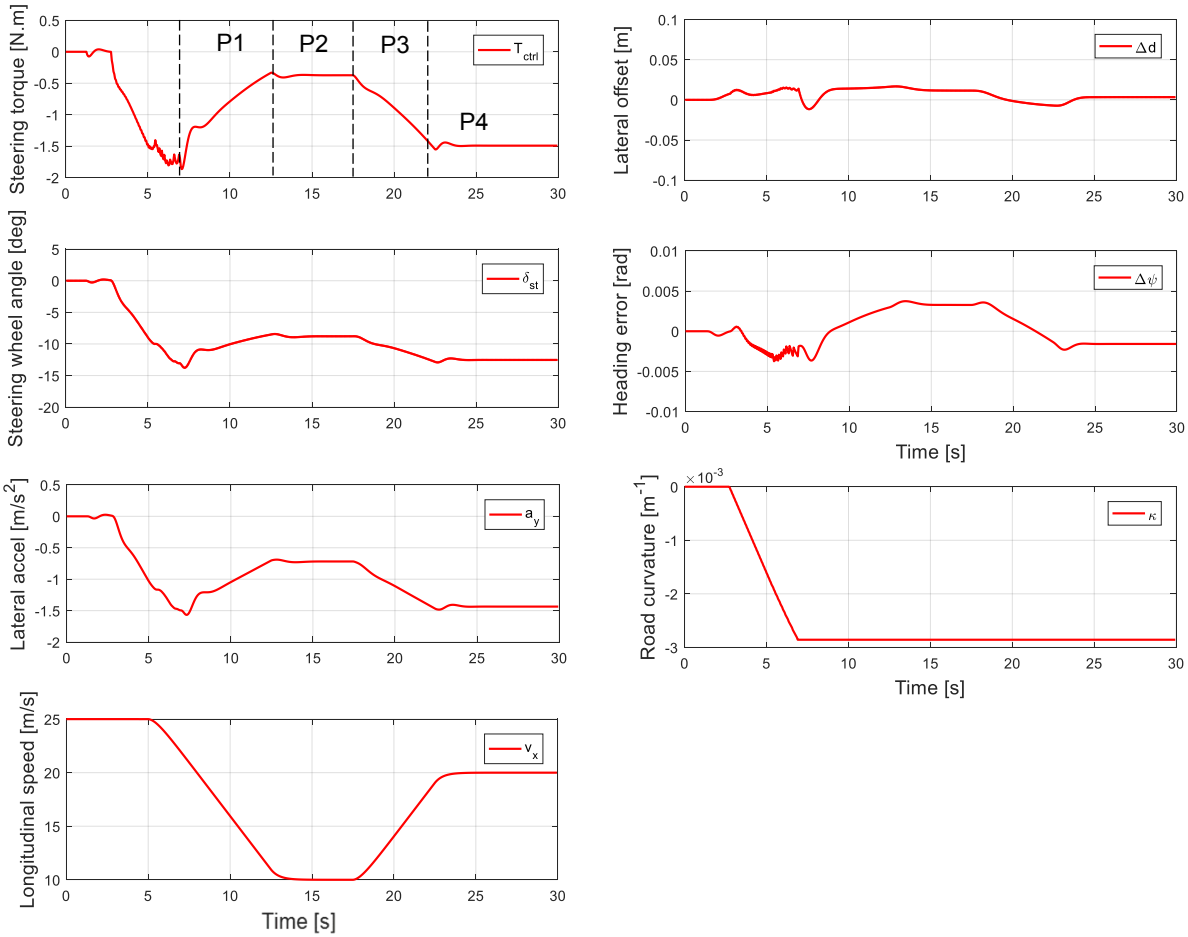


Figure 6.14 Simulation results for lane keeping in a curved road

6.6.2 Example: haptic feedback to prevent a collision due to a lane change maneuver

As a second example, we modelled a scenario on the driving simulator in which an inattentive driver attempted to override the MPC controller to perform a left lane change. In the meanwhile, a left rearward traffic vehicle was approaching at a speed 20 km/h faster than that of the AD vehicle (70 km/h). Fig. 6.15 shows the evolution of the driving scene. At the beginning of the time window of our research ($t \in [170,177]$), the driver had already the full control authority, as indicated by the curve of the shared control policy in Fig. 6.16. At $t_1 = 171.7s$, the TTC with the rearward vehicle fell below 3s. The system deemed that a left lane change was not feasible, thus enforcing the left lane-based constraint. Subject to this lateral constraint, the MPC controller resisted the driver's steering to prevent the AD vehicle from crossing the left lane border, as shown by the curve of the steering torque. As both the amplitude and the changing rate of the control were constrained in the MPC, the resisting torque arose in a smooth maneuver. As a result, the driver could adapt his steering torque to this resistance, thus not provoking the oscillation of the steering wheel. Thanks to the haptic feedback of the MPC, the

driver cancelled the lane change that might result in a collision with the rearward vehicle. Meanwhile the MPC reduced its resistance as the driver steered back the vehicle into the lane.

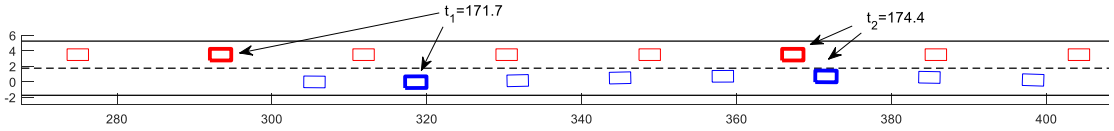


Figure 6.15 Extraction of the trajectories of the AD vehicle (blue one) and the traffic vehicle (red one) in the scenario. Both trajectories were sampled with a time step of 0.7 s. Two samples are highlighted: $t_1 = 171.7s$, $TTC\ t_1 \approx 3\ s$; $t_1 = 174.4s$, $TTC\ t_1 \approx 0\ s$.

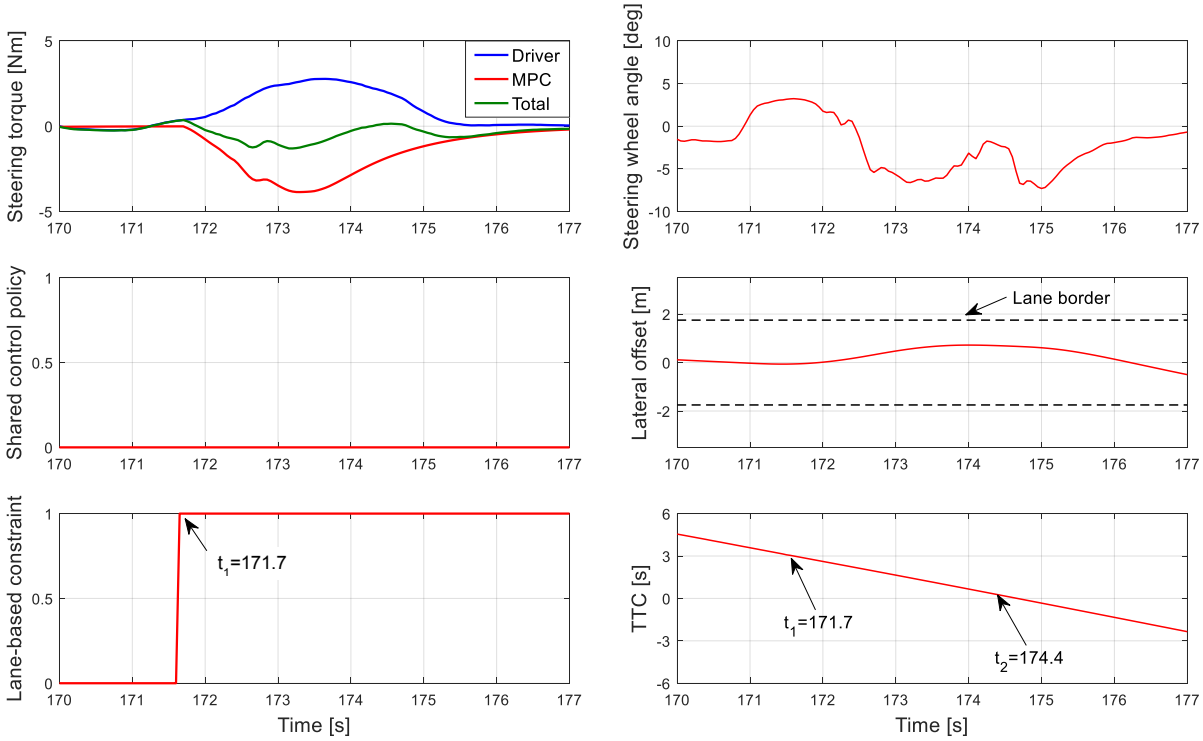


Figure 6.16 Simulation results of a scenario in which the system renders haptic resistance to warn the driver of the danger of a lane change maneuver

6.7 Preliminary user study

6.7.1 Objective

This section describes a preliminary user study of the cooperative steering system that implements the principle for control cooperation. This study aimed to investigate the following aspects of the implemented cooperation principle:

- The ease for the driver to regain the control authority from the operating system and the smoothness of control transition;

- How users interact with the active lane change assist during lane change maneuvers and the effect of the active lane change assist on their performance.

Another aspect of this principle—haptic feedback to convey hazards was not involved in this user study. The reflection on how to design and implement an experimental protocol which incites subjects to perform risky lane change maneuvers is ongoing. This aspect will be evaluated in the future work.

In this study, we compared four configurations which represented four different steering interaction forms. They are listed in Tab. 6.4. By setting the configuration SHC, we created a comparison group to SHC_ALCA in order to assess the effect of the active lane change assist during lane change maneuvers. FUA represents a type of automatic steering controller that does not share the control with the driver. In this configuration, the driver needs to override it with a large steering torque (5 Nm in our study). Finally, HAS represents another extremity in terms of control sharing, since a small amount of the driver's steering action can lead to the deactivation of the system.

Table 6.4 Four steering interaction configurations tested in the user study

Index	Name	Acronym	Description
1	Shared control with active lane change assist	SHC_ALCA	It refers to the system designed in this chapter following the architecture in Fig. 6.4. This configuration features both the "predictive shared steering control" and "ALCA" functions.
2	Shared control	SHC	This configuration has only the "predictive shared steering control" function compared to SC_ALCA. To perform lane change, the driver needs to drive manually into the target lane, then the AD vehicle stays in the new lane.
3	Full autonomy	FUA	This configuration fixes the shared control policy at one. In this way, the predictive shared steering control resumes the full steering authority. It opposed to any driver's control that could cause a deviation to its reference.
4	Haptic switch	HAS	Whenever the driver's steering intention is detected, the predictive shared steering control is disengaged. The driver regains the full steering authority immediately. The driver needs to reactivate the controller before releasing the steering wheel.

6.7.2 Methods

6.7.2.1 Participants

Twelve participants, average age of 34.0 years (ranging from 24 to 61) took part in the experiment. They came from the University of Valenciennes with three different backgrounds: student, engineer and professor. They had between 1 and 37 years of driving experience (mean 13.5 years ± 13.3).

6.7.2.2 Apparatus

A fixed-base driving simulator in a desktop configuration was used in this user study. A LCD screen was used for the visual display. The driving simulation was powered by SCANeR Studio. The AD functions were implemented in Simulink and compiled to a standalone executable which was integrated in the SCANeR platform. A SENSODRIVE SENSO-WheelSD-LC (SENSODRIVE 2017) rendered the torque generated by the MPC controller. The driver's steering torque was estimated from the current in the DC motor of the SENSO-Wheel by its control unit.

6.7.2.3 AD system and HMI

The whole AD system implemented in the driving simulator comprised a mode transition module, the situation assessment module developed in Chapter 4, the vehicle longitudinal control module developed in Chapter 5 and the cooperative steering control module developed in this chapter. The mode transition module allows subjects to switch between a manual driving (MD) mode and an AD mode by a "mode button". Following the discussion in Section 6.3, the AD system disengaged the steering control if the absolute value of the driver's steering torque exceeded a predefined threshold. This threshold was set to 1 Nm for HAS and 5 Nm for the other three configurations. Once the steering control was disengaged, the AD system entered a mode equivalent to SAE level 1 where the system performed only vehicle longitudinal control. To reengage the automatic steering control from this mode, subjects needed to press a "steering control button".

HMI graphics were directly overlaid in the virtual driving scene, as shown in Fig. 6.17. In the top-right corner of the screen, the current driving mode was shown. In the top-middle zone, a blue arrow was displayed to represent the maneuver state (γ_{ma}) of the active lane change assist. In the configuration SHC_ALCA, if a participant initiated a lane change maneuver, this arrow changed its direction to signify the acknowledgement (Fig. 6.17(c)).

6.7.2.4 Test scenarios

The test course in the simulator was composed by a ramp track and a highway mainline track. Each test run was made up of two scenarios. As shown in Fig. 6.18, the starting position of the AD vehicle was located on the ramp track. The AD vehicle first entered in Scenario A—passing a roadwork zone. In this scenario, subjects needed to adjust the vehicle's trajectory towards the left side to keep a safe distance with the road barriers. After this scenario, the AD vehicle merged in a two-lane highway mainline (lane width 3.5 m. The road limit was 90 km/h, but the traffic vehicles in the right lane were made purposely drive at 70 km/h to justify our instruction on overtaking. The gaps between traffic vehicles on the left lane were large enough to ensure that each subject could perform lane change with ease.

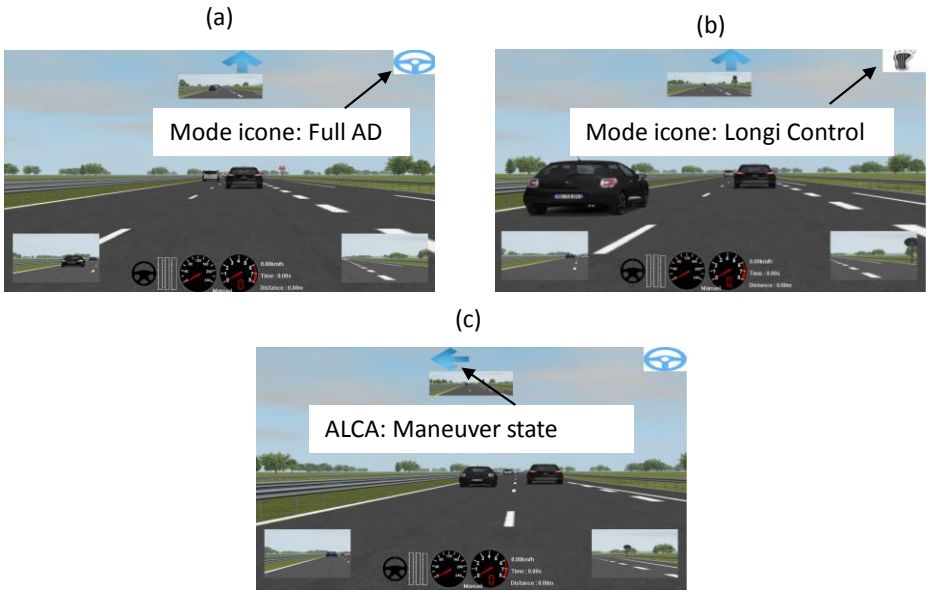


Figure 6.17 Visual HMI used in the user test: (a) mode icon for the Full AD mode; (b) mode icon for the Longi Control mode; (c) icon for the later maneuver state

6.7.2.5 Test procedure and instructions

A within-subject design was employed in this study, i.e., each participant tested all the four configurations. We randomized the order of test configurations to minimize the order effect. Prior to the test drive, the subjects got accustomed to the AD system and the driving simulator in a training course without any traffic. In this phase, we explained the transition rules between different driving modes, especially possible deactivations of the automatic steering control in different configurations. After this training phase, each subject was instructed to drive the test course manually to get familiar with the scenarios. In the meanwhile, each subject was told what they should do in the test runs.

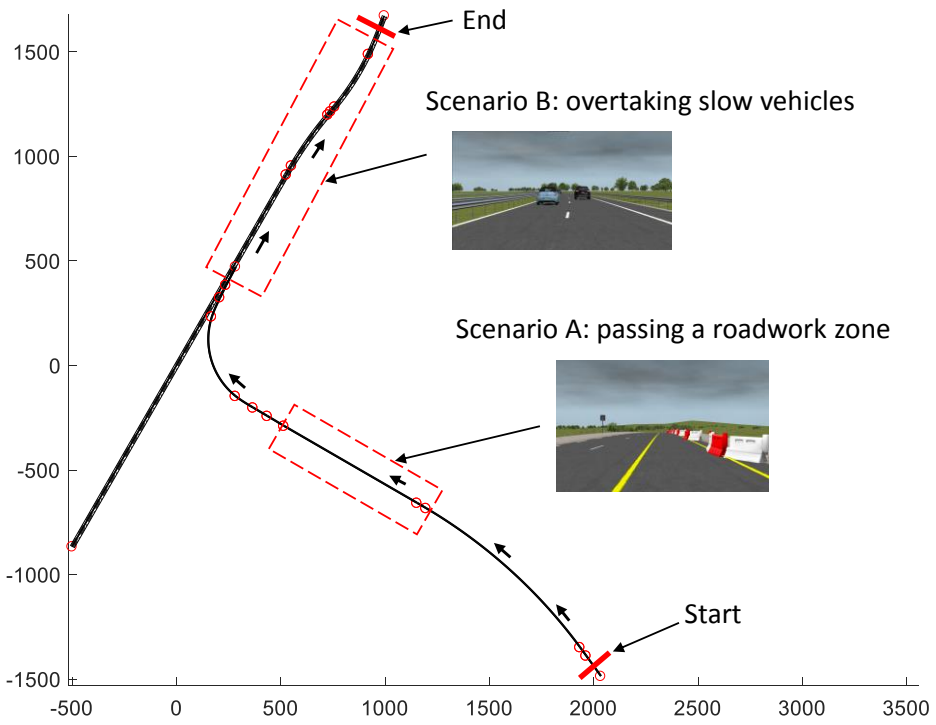


Figure 6.18 Test course and scenarios

Before the test runs, we gave a general instruction—use the automatic steering control as long as possible for all the configurations. This instruction accounted for deactivations of the automatic steering controller which could occur frequently in HAS. If a subject did not reactivate the controller, he would remain in manual steering control that could bias the test results. Moreover, subjects were asked to put their hands off the steering wheel when they did not perform required control tasks. Before the test run of the SHC_ALCA configuration, we described the mechanism of the active lane change assist and the meaning of the blue arrow in the screen. The main test procedure summarized in Fig. 6.19.

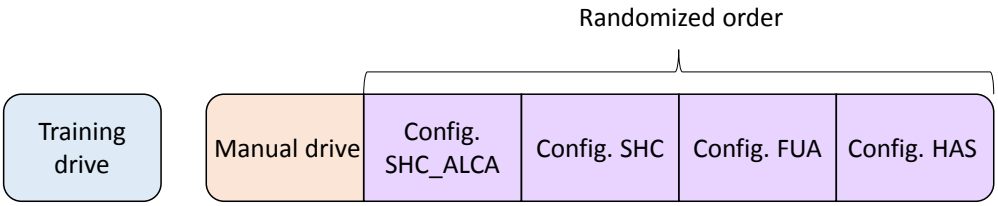


Figure 6.19 Test procedure

6.7.2.6 Metrics

We selected two objective metrics to measure driver's steering performance during the control transition.

The first metric was the *root-mean-square of the driver's steering torque*, $RMS(T_{dr})$ which was defined by

$$RMS(T_{dr}) = \sqrt{\frac{\int_{t_0}^{t_1} T_{dr}^2 dt}{t_1 - t_0}}, \tag{6.46}$$

where t_0 and t_1 are the start time and the end time of a range of interest in the collected data. For the Scenario A (roadwork zone), the range of interest started from the instant when a subject began to turn the steering wheel (detected by the sensor on the steering wheel) until the adjusted lateral trajectory was stabilized. It is illustrated by Fig. 6.20. For the scenario B (highway overtaking), the range of interest was defined as the time window of a lane change maneuver. We used the method for driver's lane change intention detection (Section 6.5) to find the range of a lane change maneuver. $RMS(T_{dr})$ measured the steering effort of a subject to control vehicle's trajectory in the presence of a steering controller.

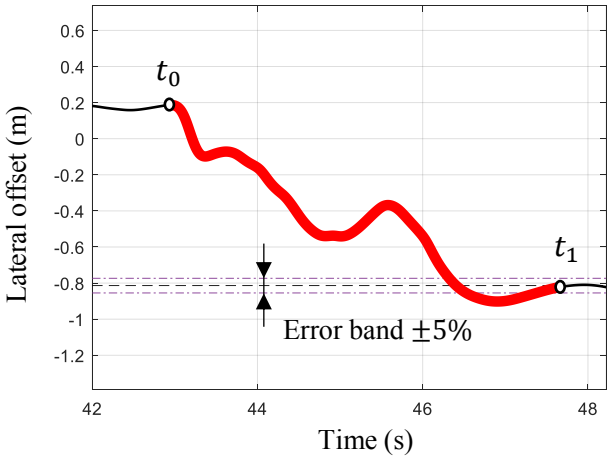


Figure 6.20 Range of interest in the Scenario A

As the second objective metric, we measured *the number of steering wheel reversals* (SWR) in a range of interest. According to the definition given by the standard SAE J 2944 (2015), a SWR occurs when a steering wheel rotates at least Δa deg in one direction and then rotates at least Δa deg in the opposite direction within a moving time window Δt. This definition is illustrated in Fig. 6.21. We used the method proposed by the AIDE project (Östlund et al. 2005) to count SWRs with Δa equals to 3 deg. The same ROIs used to calculate $RMS(T_{dr})$ were applied here for Scenario A and B as well. SWRs represent large steering corrections, therefore, more SWRs means that the control transition is less smooth.

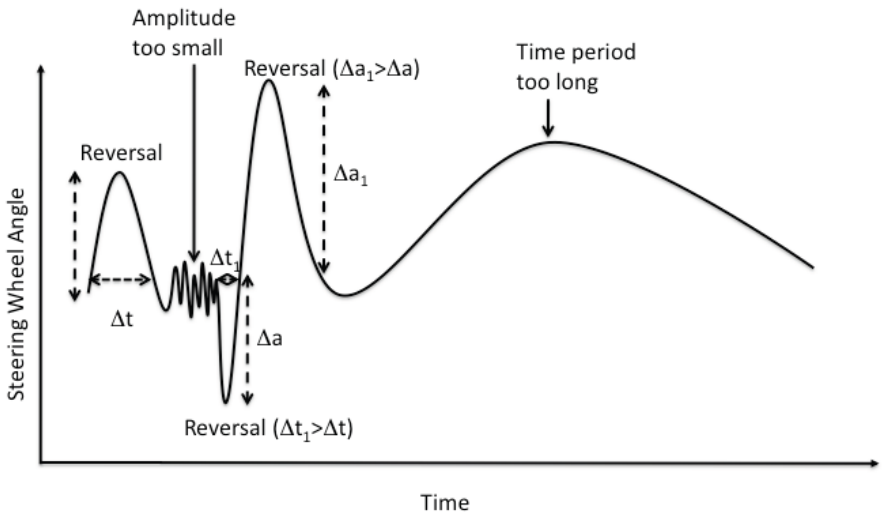


Figure 6.21 Steering reversal (extracted from standard SAE J 2944 (2015))

Subjective evaluation was based on questionnaires. After each configuration, subjects were asked to answer a questionnaire related to their experiences with this configuration. In this questionnaire, they were asked to rate their *efficiency*, *feeling of comfort*, *perceived safety*, and *ease of trajectory control* when they took over the control from the system. Each item had a four-point scale.

6.7.3 Results

6.7.3.1 Results of some typical runs

This section shows the results of some typical runs to illustrate how a subject interacted with the steering controller in different configurations. Fig. 6.22-6.23 show the results for the trajectory adjustment in Scenario A by a same subject through SHC and HAS respectively. In Fig. 6.23, the predictive shared steering controller smoothly reduced its control to give the full authority to the driver according to the change of the shared control policy. As a result, the resultant steering wheel angle remained between -2 deg and 2 deg. Thanks to the hand position sensor, the controller was aware that the driver held the steering wheel and thus did not exert the control to steer the vehicle back to the lane center. This mechanism prevents the intrusion of the controller when the driver exerts only slight control, e.g., when a driver maintains an offset to a lane center in a straight road. In contrast, abrupt deactivations of the controller (system's mode from "1" to "0") in HAS caused SWRs of the steering wheel (exceeding 5 deg in the first SWR and 10 deg in the second). What's more, whenever the driver intervened to correct the vehicle's trajectory, he deactivated the controller and he needed to reengage the controller to benefit the automatic control.

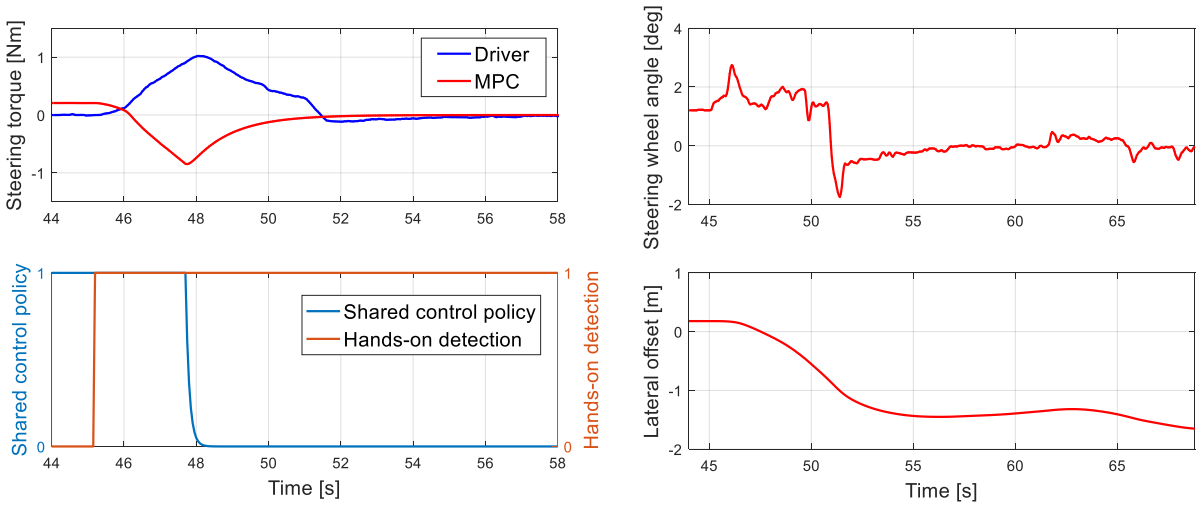


Figure 6.22 Plot of a typical run of configuration SHC in Scenario A

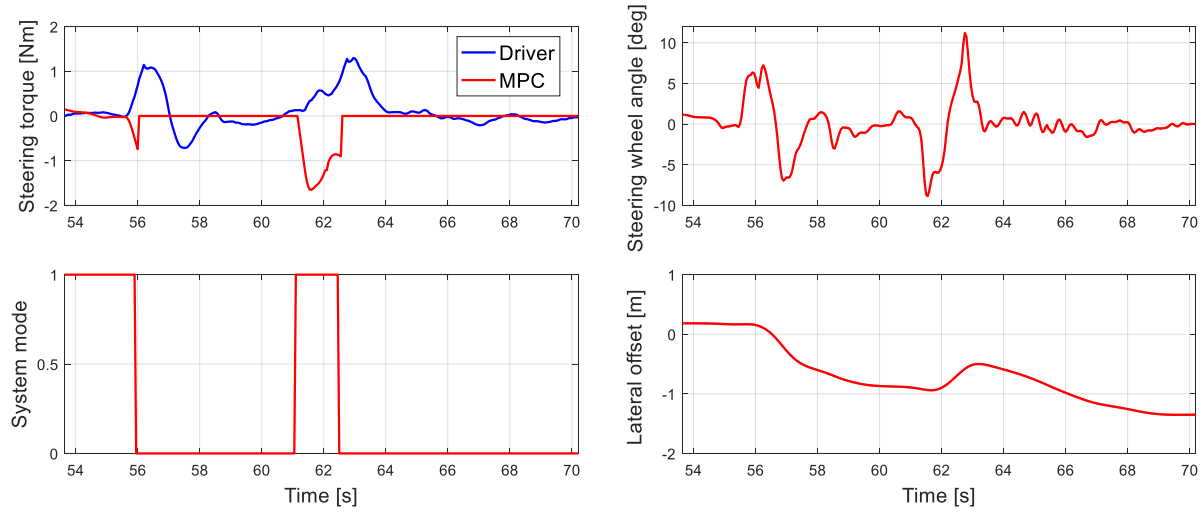


Figure 6.23 Plot of a typical run of configuration HAS in Scenario A

Fig. 6.24-6.25 illustrate two lane change maneuvers performed by a same subject through SHC and SHC_ALCA respectively. In SHC, the subject performed a lane change maneuver with little resistance from the controller. When he released the steering wheel, the controller took over the control to ensure a seamless control transition. In SHC_ALCA, the beginning of the interaction was quite similar as in SHC. When the right lane change intention of the subject was detected by the active lane change assist (see the plot "maneuver state" in Fig. 6.25), the controller resumed the control authority (see the plot "shared control policy" in Fig. 6.25) to perform the lane change maneuver in the place of the subject. Consequently, the subject first reduced his control then released the steering wheel about 3 s before the lane change was over.

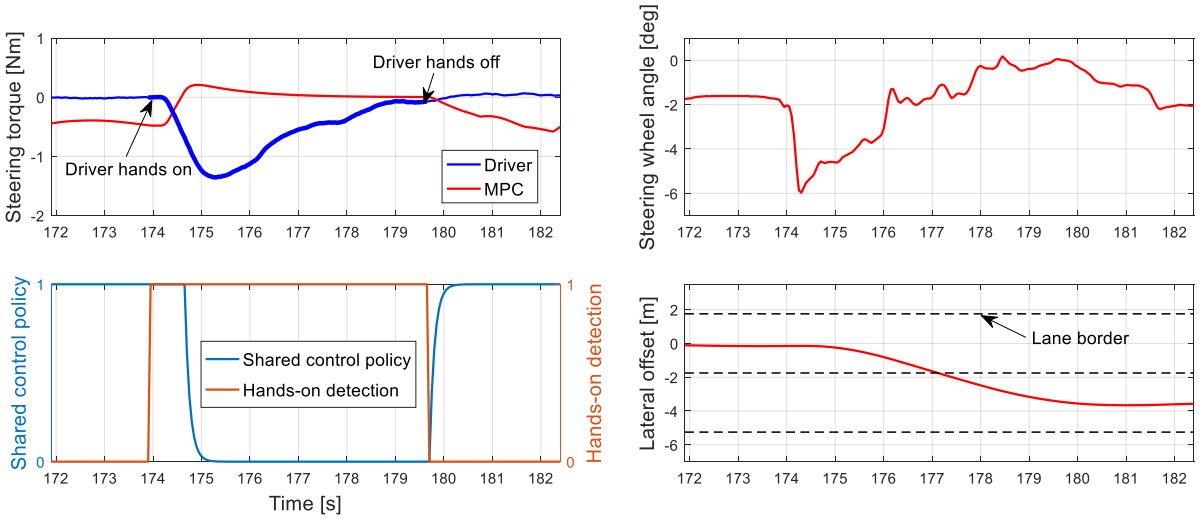


Figure 6.24 Plot of a typical run of configuration SHC in Scenario B

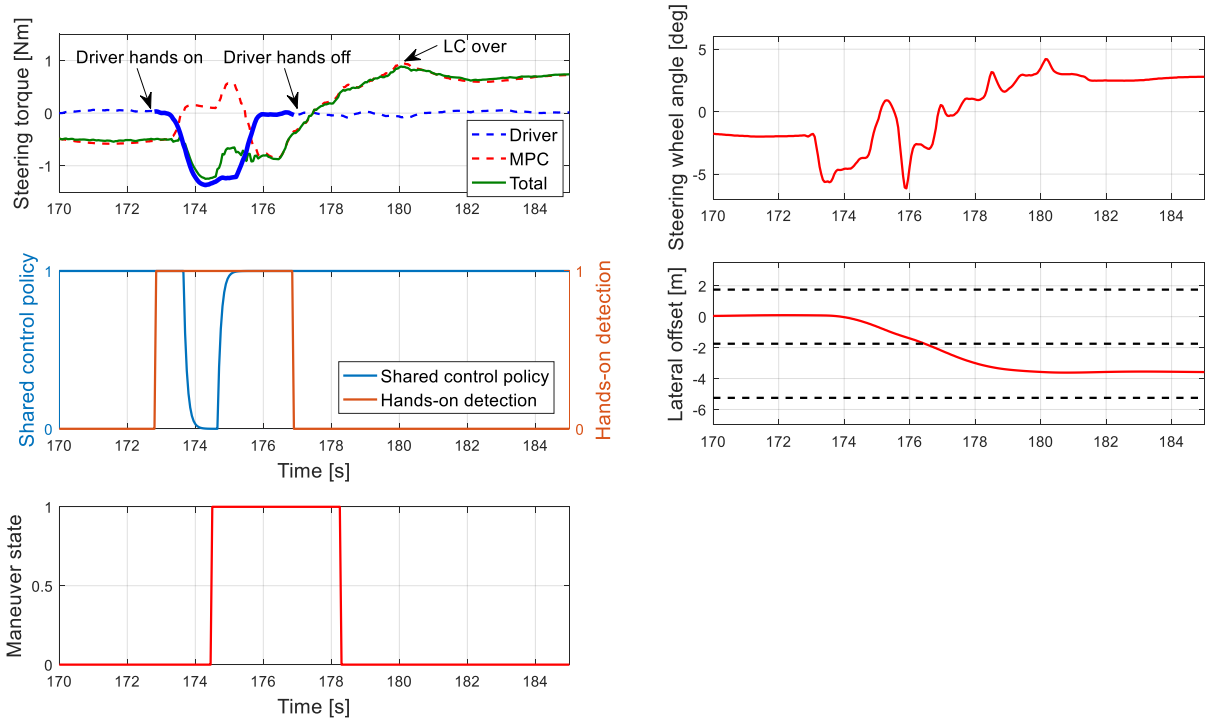


Figure 6.25 Plot of a typical run of configuration SHC_ALCA in Scenario B

6.7.3.2 Statistical results

One-way analysis of variance (ANOVA) was used for statistical analysis with a significance level of 0.05. Turkey HSD tests were used for post-hoc analysis.

Fig. 6.26 shows the results of $RMS(T_{dr})$ in Scenario A and B. There were statistically significant differences in the $RMS(T_{dr})$ among the four configurations in both Scenario A ($F(3,44) = 131.77$, $p < 0.0001$) and Scenario B ($F(3,61) = 48.22$, $p < 0.0001$). As expected, post-hoc Turkey HSD tests showed that the driver's steering effort was significantly reduced in SHC_ALCA, SHC and HAS compared to FUA in both scenarios ($p < 0.01$). For example, in

SHC subjects saved on average 85.2% effort in Scenario A and 62.9% effort in Scenario B compared to FUA. However, the post-hoc analysis revealed that there were no significant differences in the $RMS(T_{dr})$ in Scenario B between SHC_ALCA and SHC. This observation indicates that the active lane change assist did not reduce the driver's steering effort in lane change maneuvers, though it was intended to support the driver to perform lane change maneuver.

Statistical results of SWR are presented in Fig. 6.27. The ANOVA revealed significant effects of the configurations in Scenario A ($F(3,42) = 3.9$, $p < 0.015$). Especially, HAS had the highest average number of SWRs, i.e., the driver made more steering corrections to adjust the vehicle's trajectory when interacting with the steering controller than in the other three configurations. Compared to HAS, SHC significantly reduced the SWRs by 70% ($p < 0.05$), thus leading to a smoother control transition. In Scenario B, there was also a significant difference on the steering wheel reversals ($F(3,61) = 18.05$, $p < 0.0001$). FUA yielded the highest steering reversals in Scenario B. It was mainly due to much stronger resistances a subject received from the steering controller during a lane change maneuver. Some subjects even triggered the deactivation of the steering controller by exceeding the maximum torque threshold. In contrast, deactivations at small steering torques in HAS did not have significant effects on the SWR (with no significant difference compared to SHC) during lane change maneuvers. The SWR in SHC_ALCA had wide variance between subjects (SD =1.16). It means that some subjects attempted to correct controller's actions rather than follow them.

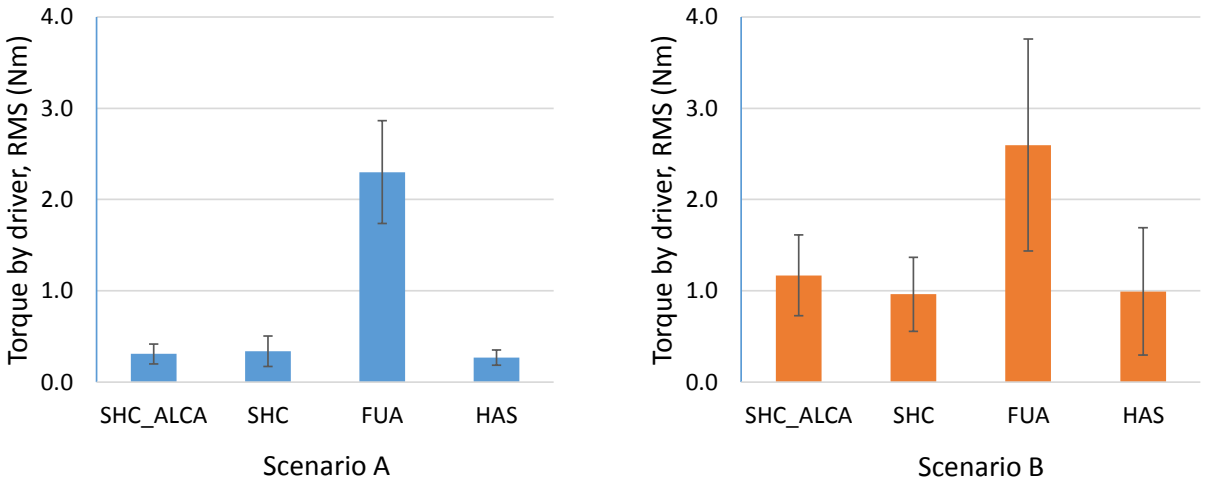


Figure 6.26 RMS of driver's steering torque in the different configurations in Scenario A and B

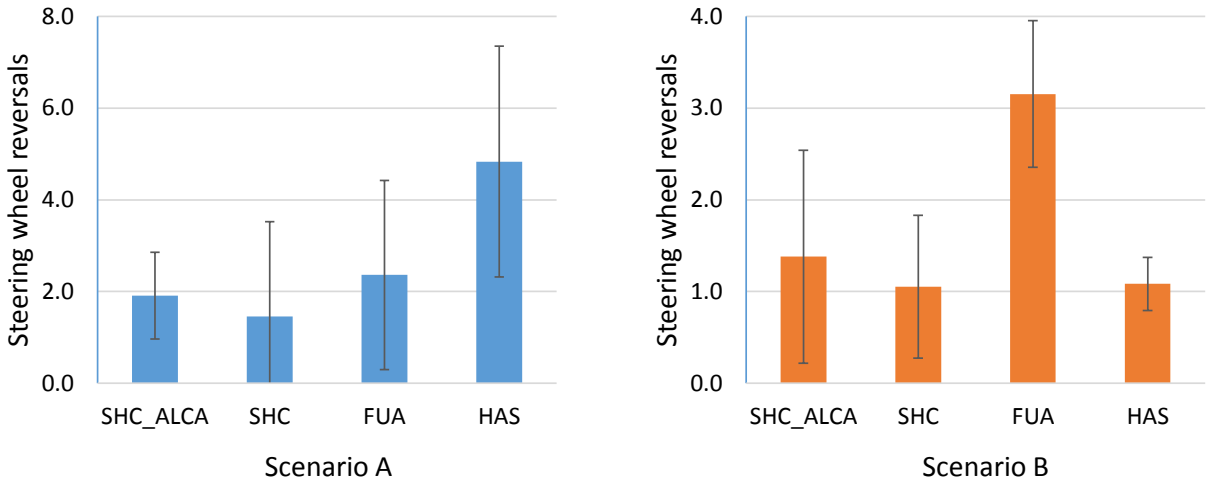


Figure 6.27 Steering wheel reversals in the different configurations in Scenario A and B

Fig. 6.28 shows the mean values of the responses on each item in the questionnaire (standard deviations on each item are omitted for figure clarity). There were significant effects of the configurations in comfort ($F(3,44) = 6.33$, $p < 0.001$) and sense of control ($F(3,44) = 6.33$, $p < 0.0006$). There were no significant differences between the four configurations in efficiency of regaining control and safety feeling. Except for the item efficiency, SHC received the highest rating in terms of comfort (M=3.42, SD=0.67), safety (M=3.42, SD=0.67) and control (M=3.75, SD=0.45).

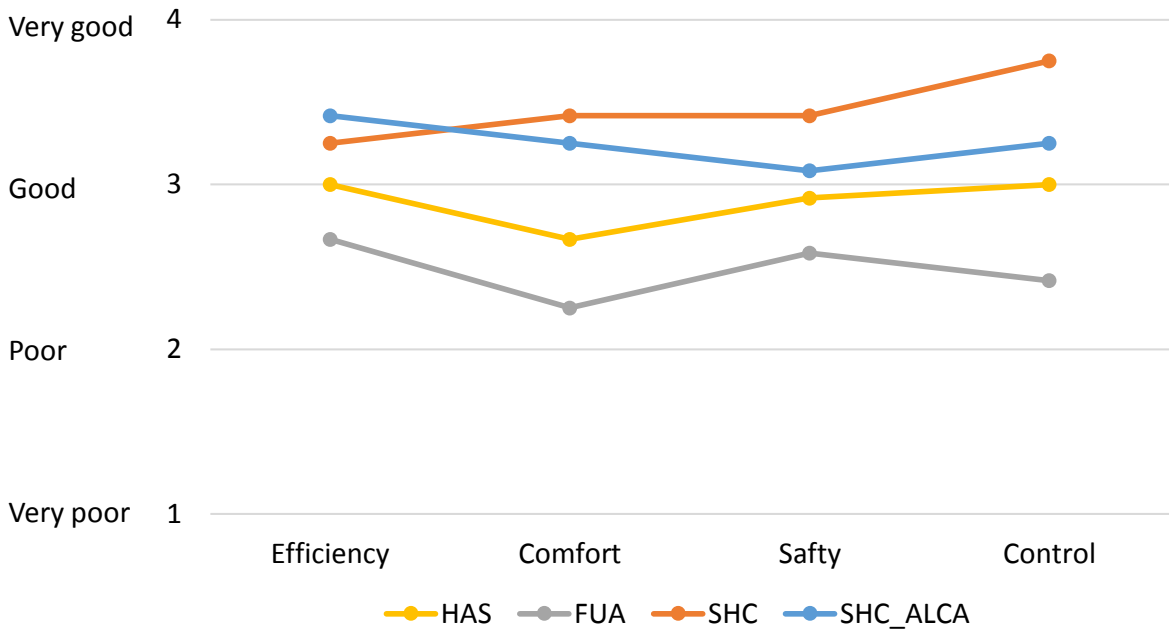


Figure 6.28 Subjective comparison between the four configurations in terms of efficiency, comfort, safety and control feeling

6.7.4 Discussion

Statistical analysis based on objective metrics suggested that with the predictive shared steering controller (SHC) users used much less steering efforts to adjust vehicle's lateral trajectory than with an automatic steering controller that did not adapt control to that of the driver (FUA). When the driver took over control to adjust lane positions in Scenario A, the predictive shared steering controller ensured a smoother control transition than a controller that can be deactivated by a small steering torque of the driver (HAS). These two conclusions are also confirmed by subjective ratings, as SHC received two highest ratings in terms of sense of control and comfort.

As for the active lane change assist, some subjects benefited from the support of the automation during lane change maneuvers, as illustrated by an example in Section 6.7.3.1. However, the average steering effort of subjects to perform lane change maneuvers was not reduced with the active lane change assist. Meanwhile, the SWR number had large variance between subjects. According to our interviews during the test, some subjects reported that they did not have confidence on the automation as new users. This may explain why they continued to exert forces after the predictive shared steering controller had begun to perform lane change. The second plausible cause may be related to the interference on the trajectory of a lane change maneuver between subjects and the controller. The trajectories of lane change maneuvers realized by the steering controller were strongly influenced by the weights in the cost function in the MPC. Subjects with different driving styles may have different accustomed lane change trajectories, e.g., a smoother one or a more aggressive one. The deviation between the subject's expected trajectory and that of the steering controller could incite subject's steering corrections which increased SWRs.

6.8 Conclusions

This chapter has described the design of a cooperative steering control system that allows the driver to override the system's steering control to adjust lane positions or make lane changes in highway driving scenarios. This system consists of two main components. The first one is a predictive shared steering controller which was implemented in the MPC framework. By adapting the weight on the stage cost and implementing dynamic constraints online, the shared steering controller ensures seamless control transfer between the system and the driver while conveying potential hazards through haptic feedback. The second function, namely active lane change assist, can detect the driver's lane change intention and assist the driver during the lane change maneuver.

The capability of the cooperative steering control system to perform lane keeping/lane change was first demonstrated in a simulation study. In a second experiment, we simulated a scenario in which the system rendered haptic resistance on the steering wheel to prevent a potential collision caused by driver override. During a preliminary user test, we compared four different steering interaction concepts. The test results suggest that the predictive shared steering control developed in this thesis allowed the driver to take over the control with ease and at the same time ensured a smooth transition. The test results also exposed interaction issues between subjects and the active lane change assist. We made hypotheses on the causes and these hypotheses will be studied in the future work.

7 GENERAL CONCLUSION AND PERSPECTIVE

7.1 Conclusions

This thesis addresses the design of cooperation between the human driver and the automated driving (AD) system. We applied the theoretical framework of human-machine cooperation and implemented the user-centered design (UCD) process to design cooperative control systems. Given the multidisciplinary objectives of this research, we present the conclusions from two aspects: the aspect of human-machine interaction and the technical aspect.

7.1.1 Aspect of human-machine interaction

We firstly decomposed the shared authority involved in driver-vehicle cooperation into three levels within a common hierarchy between Michon's model of the driving task and a layered functional architecture for the AD system. We proposed, implemented and evaluated two cooperation principles at the tactical level and the operational level respectively.

At the tactical level, we proposed a principle for maneuver cooperation that allows the driver to change system's maneuver plan. This principle was implemented through the design of a cooperative longitudinal control system in a use case of highway merging management. In addition to the technical system, we designed a set of interfaces based on head-up display (HUD) and augmented reality. User test results suggest that the proposed principle has the potential to enhance the performance of the AD system in terms of handling the merging of a traffic vehicle in fluid and congested traffics. Results also show that the users tended to confirm system's maneuver intention during the cooperation if they shared the same intention with the

system's. At last, test results imply that the system needs to initiate the cooperation with a sufficient time before the engagement of maneuver plan so that the driver has time to assess the situation and to participate into the cooperation. Through this conclusion, we emphasized the requirement on situation assessment function of the system and the role of HMI in sharing situational awareness (SA) with the driver.

The second principle was proposed for the control cooperation at the operational level. Following this principle, the driver can easily override the system's control while benefiting from the support of the system in terms of task assistance and hazard warning. This principle was implemented through the design of a cooperative steering control system in a use case of highway lane positioning and lane changing. In the framework of haptic shared control, steering torques from the driver and the system served as a communication channel of their control activities. The results from a preliminary user test show that with the designed system the users could easily regain the steering control and enjoyed a smooth control transition.

7.1.2 Technical aspect

In this thesis, we also addressed the technical aspects to achieve driver-vehicle cooperation. We highlighted the role of the system's SA in both maneuver planning function and maneuver cooperation. We proposed an approach for highway driving scene representation based on a digital map. In this scene model, the states of traffic vehicles are represented in the Frenet frame on the road curve. A qualitative mapping based on vehicle's Frenet coordinates creates the spatial awareness of the AD system. We further proposed a method for the prediction of vehicle's longitudinal trajectory for maneuver planning. This method uses quintic polynomials to model the longitudinal dynamics of a vehicle that is maneuvering. The decision to switch to it from the constant acceleration model is formulated as a hypothesis testing problem. Incorporating the estimated jerk and context information, this model yields better prediction accuracy than the constant acceleration model for dynamic longitudinal maneuvers.

In the use case of highway merging management, we exploited the behavior-based paradigm for maneuver cooperation. To this end, we designed a hierarchical finite state machine (HFSM) specific to the highway merging management. A main feature of this HFSM consists of a state called intention phase. In this state, the system assesses the actual or future scene configuration and thus actively interacts with the merging vehicle by manifesting its intention (pass or yield). Moreover, we employed the model predictive control (MPC) framework to generate a smooth trajectory of a virtual leader for a low-level controller. In this way, the controller can perform a maneuver generated from the HFSM while ensuring the driving comfort.

In the use case of highway lane positioning and lane changing, we adapted MPC for haptic shared steering control. By changing the relative weight in the cost function online, the steering controller minimizes its control when the driver intervenes in the control-loop. By enforcing vehicle position constraints at lane borders, the steering controller forces the vehicle's trajectory to remain within a constraint-bounded navigational zone to prevent lane departure or potential collisions with traffic vehicles in the adjacent lanes. In this way, when the driver intends to steer the vehicle out of this zone, he receives the resistance from the system as a warning signal. We also developed an active lane-change assist function at the tactical level. This function actively assists the driver during a lane change maneuver if it detects driver's lane change intention.

7.2 Perspective

The following sections provide a summary of perspectives for further work. In Sections 7.2.1-7.2.3, we discuss specific open questions and directions suggested by the current research contributions. In section 7.2.4, we give more general prospects regarding to the system integration and the extension of application scenarios.

7.2.1 Situational awareness of the AD system

In the proposed framework for vehicle longitudinal trajectory prediction, we formalized the maneuver detection problem as a χ^2 test based on accumulated measurement residuals over a sliding time window. The threshold λ is determined to achieve a small probability of false alarm, however, it provides no information on the probability of detection. Furthermore, the window size was tuned by hand in the simulation based on a trade-off between the detection time delay and the randomness of χ^2 variable. Evidently, the time window size also influences the probability of detection and the jerk estimation. In the future work, it would be of interest to implement the *generalized likelihood ratio test* (GLRT, Kay 1998, 200) which is a composite hypothesis test based on the maximum likelihood estimates (MLE) of the unknown parameters—in our case, the unknown parameter is the jerk. The GLRT offers an appealing analytical framework for maneuver detection performance evaluation, since it is based on the Neyman-Pearson theorem (Kay 1998, 61). Moreover, as the estimate on jerk by the least square method is the *best linear unbiased estimate* which is equivalent to MLE under the Gaussian assumption, the statistical information for jerk estimation can be directly used for the maneuver detection.

Another concern for the maneuver detection and jerk estimation is the coupling of the longitudinal and lateral motions as revealed by the measurement model (4.14) in Section 4.4.

Consequently, the jerk estimation could be influenced by strong lateral motion, e.g., during a lane change maneuver. This scenario needs to be studied more deeply in the future work. Finally, the final velocity as an end state of the quintic polynomial motion model is determined by a heuristic based on the driving context in the current work. Future research also includes using formal methods, such as hidden Markov Model and dynamic Bayesian network to infer the end state of a maneuver from the driving context.

The proposed scene representation was mainly used to support system's functions in the current work. How to share this representation to the driver through HMI could be a future research direction from the perspective of human-machine interaction. Inversely, it would be useful to enrich the scene representation by considering what information a driver may need to know from a certain driving scene.

7.2.2 Maneuver cooperation

In the current work, the cooperative maneuver planning was modelled as a HFSM. The rule-based state transitions lack the ability to deal with the uncertainty of the inputs. Especially some transition rules in the intention phase directly use the predicted trajectory of the merging vehicle, thus making the system prone to the prediction errors. To propose a probabilistic decision-making framework that deals with the uncertainty of traffic vehicle states and the uncertainty propagation in the predicted trajectory is an important step towards an application for real vehicles. Moreover, the system simulates the ego vehicle's future motion by the constant acceleration model in the intention phase to evaluate "pass" or "yield". It would be desirable to simulate different trajectories representing different maneuver plans (e.g., using different accelerations or speed profiles). In this way, new maneuver alternatives, e.g., lane change maneuver can be added and be evaluated compared to other alternatives.

Based on the feedback from the user test in Section 5.8, current interaction logic and HMI could be improved in a new design cycle in the UCD process. The current cooperation principle could be extended so that the driver cannot only change but also confirm the system's current intention. If the driver confirms the system's intention, the system can directly engage the intended maneuver without showing alternatives. This modification is consistent with the test results indicating that subjects tended to confirm system's intention. Regarding to HMI configurations, test results suggest that HMIs dispatched into different zones increased attentional demands of users. The HUD and augmented reality techniques constitute efficient solutions to address this problem, because they allow overlaying HMI contents (in our case,

system's intention and alternatives) on the part of the real driving scene which the driver pays attention to.

7.2.3 Control cooperation

The MPC-based haptic shared control framework opens several avenues for future research. First, future research should investigate the haptic resistance caused by the state constraint (lane-based constraints). When the system's lateral offset state approaches the constraint under the driver's control, the current controller reacts in an open-loop maneuver, because under the shared control policy the single objective of the controller is to minimize its control (as a minimum-energy controller). It would be interesting to add some control objectives, e.g., penalizing the control changing rate to avoid the brutal torque rise or penalizing the large steering angle variation in a sense to increase the steering wheel stiffness. Most significantly, a user test is necessary in the future to study how users react to these haptic resistances rendered on the steering wheel. The effects of haptic feedbacks on driver's SA and user's acceptance should be examined in scrutiny.

Second, concerning the shared control policy, the proposed solution to detect the driver's steering intention consists of using a specific sensor for hands-on detection and placing a threshold on driver's steering torque. In the future study, it is worthy of studying a less costly solution (without using hands-on detection sensor and torque sensor), e.g., by estimating the driver's steering intention from the steering angle and the angle rate measurements. One may borrow the concept of maneuver detection used in the longitudinal trajectory prediction framework. The general idea is to set a reference model of steering system dynamics to characterize "hands-off" situation and to monitor the deviation between the measurements and the prediction of this model.

Lastly, from the control perspective, the MPC framework used for haptic shared control could be improved. Particularly, the assumption on the perfect knowledge of the state measurements limits the application of the framework. Moreover, the uncertainty on model parameters and the modelling error also need to be considered to ensure the robustness of the framework. As such, it is important to formulate a robust MPC problem (Bemporad and Morari 1999) in the future work.

The user test in Section 6.7 exposed the issues on user's interaction with the active lane change assist system. To investigate the hypothesis on user trust, a user study could be conducted in the future to verify whether the user's trust and the performance will increase as they get used to the system (similar to assess the learnability of a system by generating a learning curve).

Another hypothesis concerns that the user may want to correct the vehicle's trajectory during a lane change maneuver. A future direction is to develop a motion planner that can adapt the planned path to the driver's control. This direction has already been explored by Benloucif et al. (2017) and the simulator study showed the positive effect of a such adaptive motion planner in reducing the driver's effort to perform lane change maneuvers.

7.2.4 General prospects

We have proposed two cooperative control frameworks separately, the one for maneuver cooperation in the longitudinal dimension and the other for control cooperation in the lateral dimension. A future direction is to integrate these two frameworks into the proposed hierarchical cooperative control architecture. At the tactical level, more maneuver alternatives and more driving contexts could enrich the actual HFSM. At the operational level, the MPC-based shared control framework can be applied for vehicle longitudinal control, and active force feedback pedals could serve as a haptic interface. The key to combine control functions in longitudinal and lateral dimensions constitutes the motion planning function which was not formally addressed in this thesis. Especially, recent works (Bender et al. 2015; Park, Karumanchi, and Iagnemma 2015; Gu, Dolan, and Lee 2016) introduced the "maneuver aspect" into the motion planning, which may bring new ideas for driver-vehicle cooperation. These works decomposed a feasible trajectory space into discrete subspaces each of which can be interpreted as a maneuver alternative. In this framework of the trajectory space decomposition, the driver can either directly select a maneuver alternative—a trajectory subspace following the maneuver cooperation principle. He can also override the system's control to navigate freely in each trajectory subspace using the control cooperation principle. Furthermore, if he controls the vehicle's trajectory into another subspace, a maneuver transition can be triggered at the tactical level.

Another future research direction is to address critical situations in the scope of driver-vehicle cooperation. An important use case in this direction is the take-over scenario for a level 3 AD system (refer to Section 1.1). When the driver takes over the control following the request of the system, instead of immediately transferring all the control to the driver, the system could "accompany" the driver via the control interface, e.g., providing a minimum guidance to help the driver quickly rebuild the SA.

REFERENCES

AASHTO. 2010. *A Policy on Geometric Design of Highways and Streets*. 6th ed. American Association of State Highway and Transportation Officials. Accessed April 05, 2017. https://bookstore.transportation.org/collection_detail.aspx?ID=110

Abbink, David A.. 2006. "Neuromuscular Analysis of Haptic Gas Pedal Feedback during Car Following." PhD diss., Delft University of Technology.

Abbink, David A., and Mark Mulder. 2010. "Neuromuscular Analysis as a Guideline in Designing Shared Control." In *Advances in Haptics*, edited by Mehrdad Hosseini. InTech. Accessed April 05, 2017. http://www.intechopen.com/books/howtoreference/advances-in-haptics/neuromuscular-analysis-as-a-guideline-in-designing-shared-control

Abbink, David A., Mark Mulder, and Erwin R. Boer. 2012. "Haptic Shared Control: Smoothly Shifting Control Authority?" *Cognition, Technology & Work* 14 (1): 19–28.

ADAPTIVE. 2016. "Automated Driving Applications and Technologies for Intelligent Vehicles." Accessed April 05, 2017. https://www.adaptive-ip.eu/

Albert, Martin, Alexander Lange, Annika Schmidt, Martin Wimmer, and Klaus Bengler. 2015. "Assessment of Interaction Concepts Under Real Driving Conditions." *Procedia Manufacturing* 3 (January): 2832–39.

Albus, James S., Hui-Min Huang, Elena R. Messina, and Karl Murphy. 2002. "4D/RCS Version 2.0: A Reference Model Architecture for Unmanned Vehicle Systems." NIST Interagency/Internal Report (NISTIR)-6910. National Institute of Standards and Technology (NIST). Accessed April 05, 2017. http://www.nist.gov/manuscript-publication-search.cfm?pub_id=821823

Alessandretti, Giancarlo, Angelos Amditis, Sarah Metzner, Emma Johannson, and Felix Fahrenkrog. 2014. "Interactive Deliverable D1.9 - Final Report." Accessed April 05, 2017. http://www.interactive-ip.eu/index.dhtml/docs/interactIVe_SP1_20140506v1.2-D19-Final_Report.pdf

Alessandrini, Adriano, Alessio Cattivera, Carlos Holguin, and Daniele Stam. 2014. "CityMobil2: Challenges and Opportunities of Fully Automated Mobility." In *Road Vehicle Automation*, edited by Gereon Meyer and Sven Beiker, 169–84. Lecture Notes in Mobility. Springer International Publishing.

Althoff, Matthias. 2010. "Reachability Analysis and Its Application to the Safety Assessment of Autonomous Cars." PhD diss., Technische Universität München. Accessed April 05, 2017. https://mediatum.ub.tum.de/node?id=963752

Anderson, Sterling J, Sisir B Karumanchi, Karl Iagnemma, and James M Walker. 2013. "The Intelligent Copilot: A Constraint-Based Approach to Shared-Adaptive Control of Ground Vehicles." *IEEE Intelligent Transportation Systems Magazine* 5 (2): 45–54.

Anderson, Sterling J., Steven C. Peters, Tom E. Pilutti, and Karl Iagnemma. 2010. "An Optimal-Control-Based Framework for Trajectory Planning, Threat Assessment, and

Semi-Autonomous Control of Passenger Vehicles in Hazard Avoidance Scenarios." *International Journal of Vehicle Autonomous Systems* 8 (2–4): 190–216.

Ardelt, Michael, Constantin Coester, and Nico Kaempchen. 2012. "Highly Automated Driving on Freeways in Real Traffic Using a Probabilistic Framework." *IEEE Transactions on Intelligent Transportation Systems* 13 (4): 1576–85.

Bageshwar, Vibhor L., William L. Garrard, and Rajesh Rajamani. 2004 "Model Predictive Control of Transitional Maneuvers for Adaptive Cruise Control Vehicles." *IEEE Transactions on Vehicular Technology* 53 (5): 1573–85.

Bainbridge, Lisanne. 1983. "Ironies of Automation." *Automatica* 19 (6): 775–779.

Bar-Shalom, Yaakov, X. Rong Li, and Thiagalingam Kirubarajan. 2001. *Estimation with Applications to Tracking and Navigation*. 1st ed. New York: Wiley-Interscience.

Bartels, Arne, Christian Berger, Holger Krahn, and Bernhard Rumpe. 2009. "Qualitätsgesicherte Fahrentscheidungsunterstützung Für Automatisches Fahren Auf Schnellstrassen Und Autobahnen." In *Proceedings Des 10. Braunschweiger Symposiums "Automatisierungssysteme, Assistenzsysteme Und Eingebettete Systeme Für Transportmittel,"* 10:341–53. Braunschweig.

Bauer, Eric, Felix Lotz, Matthias Pfromm, Matthias Schreier, Stephan Cieler, Alfred Eckert, A. Hohm, et al. 2012. "PRORETA 3: An Integrated Approach to Collision Avoidance and Vehicle Automation." *At - Automatisierungstechnik* 60 (12): 755–765.

Behrmann, Elisabeth. 2016. "Volvo Plans Self-Driving Car by 2021 to Challenge BMW." *Automotive News*. July. Accessed April 05, 2017. http://www.autonews.com/article/20160722/COPY01/307229945/volvo-plans-self-driving-car-by-2021-to-challenge-bmw.

Bemporad, Alberto, and Manfred Morari. 1999. "Robust Model Predictive Control: A Survey." In *Robustness in identification and control*, edited by Andrea Garulli, Alberto Tesi and Antonio Vicino, 207–26. Lecture Notes in Control and Information Sciences. Springer, London.

Bemporad, Alberto, Manfred Morari, Vivek Dua, and Efstratios N. Pistikopoulos. 2002. "The Explicit Linear Quadratic Regulator for Constrained Systems." *Automatica* 38 (1): 3–20.

Bender, Philipp., Ömer Şahin Taş, Julius Ziegler, and Christoph Stiller. 2015. "The Combinatorial Aspect of Motion Planning: Maneuver Variants in Structured Environments." In *2015 IEEE Intelligent Vehicles Symposium (IV)*, 1386–92.

Benloucif, Mohamed Amir, Anh-Tu Nguyen, Chouki Sentouh, and Jean-Christophe Popieul. 2017. "A New Scheme for Haptic Shared Lateral Control in Highway Driving Using Trajectory Planning." In 2017 IFAC World Congress. Toulouse, France. (accepted)

Berndt, Holger, Jorg Emmert, and Klaus Dietmayer. 2008. "Continuous Driver Intention Recognition with Hidden Markov Models." In *2008 11th International IEEE Conference on Intelligent Transportation Systems*, 1189–94.

Besselmann, Thomas, and Manfred Morari. 2009. "Autonomous Vehicle Steering Using Explicit LPV-MPC." In *2009 European Control Conference (ECC)*, 2628–33.

Billings, Charles E. 1997. *Aviation Automation: The Search for a Human-Centered Approach*. Mahwah, N.J.: Lawrence Erlbaum Associates Publishers.

Blanco, Myra, Jon Atwood, Holland Vasquez, Tammy Trimble, Vikki Fitchett, Josh Radlbeck, Gregory Fitch, and Russell Sheldon. 2015. "Human Factors Evaluation of Level 2 and

Level 3 Automated Driving Concepts." Report (No. DOT HS 812 182). Washington, DC: National Highway Traffic Safety Administration. Accessed April 05, 2017. http://www.nhtsa.gov/DOT/NHTSA/NVS/CrashAvoidance/TechnicalPublications/2015/812182_HumanFactorsEval-L2L3-AutomDrivingConcepts.pdf

BMW. 2017. "BMW 7 Series Sedan : Driver Assistance." Accessed February 25, 2017. http://www.bmw.com/com/en/newvehicles/7series/sedan/2015/showroom/driver_assistance.html#drivingassistant.

Boer, Erwin R., Mauro Della Penna, Hans Utz, Liam Pedersen, and Maarten Sierhuis. 2015. "The Role of Driving Simulators in Developing and Evaluating Autonomous Vehicles." In *Proceedings of DSC 2015 Europe Driving Simulation Conference & Exhibition*, 3–10. Tübingen, Germany.

Boink, Rolf, Marinus M. van Paassen, Mark Mulder, and David A. Abbink. 2014. "Understanding and Reducing Conflicts between Driver and Haptic Shared Control." In *2013 IEEE International Conference on Systems, Man, and Cybernetics*, 1510–15.

Boyd, Stephen. 2008. "Linear Quadratic Regulator: Discrete-Time Finite Horizon." Lecture Slides presented at the EE363: Linear Dynamical Systems, Stanford, CA, US, September. Accessed April 05, 2017. http://stanford.edu/class/ee363/lectures/dlqr.pdf.

Boyd, Stephen, and Lieven Vandenberghe. 2004. *Convex Optimization*. Cambridge, UK ; New York: Cambridge University Press.

Brannstrom, Mattias, Erik Coelingh, and Jonas Sjöberg. 2010. "Model-Based Threat Assessment for Avoiding Arbitrary Vehicle Collisions." *IEEE Transactions on Intelligent Transportation Systems* 11 (3): 658–69.

Brännström, Mattias, Erik Coelingh, and Jonas Sjöberg. 2013. "Decision-Making on When to Brake and When to Steer to Avoid a Collision." *International Journal of Vehicle Safety* 7 (1): 87–106.

Broggi, Alberto, Pietro Cerri, Stefano Debattisti, Maria C. Laghi, Paolo Medici, Daniele Molinari, Matteo Panciroli, and Antonio Prioletti. 2015. "PROUD--Public Road Urban Driverless-Car Test." *IEEE Transactions on Intelligent Transportation Systems* 16 (6): 3508–19.

Buehler, Martin, Karl Iagnemma, and Sanjiv Singh, eds. 2009. *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*. 1st ed. Springer Tracts in Advanced Robotics. Berlin: Springer-Verlag.

Camacho, Eduardo F., and Carlos Bordons. 2007. *Model Predictive Control*. 2nd ed. Advanced Textbooks in Control and Signal Processing. London: Springer-Verlag.

Carvalho, Ashwin, Stéphanie Lefévre, Georg Schildbach, Jason Kong, and Francesco Borrelli. 2015. "Automated Driving: The Role of Forecasts and uncertainty—A Control Perspective." *European Journal of Control* 24 (July): 14–32.

Cassidy, Michael, and Soyoung Ahn. 2005. "Driver Turn-Taking Behavior in Congested Freeway Merges." *Transportation Research Record: Journal of the Transportation Research Board* 1934 (January): 140–47.

Cerone, Vito, Mario Milanese, and Diego Regruto. 2009. "Combined Automatic Lane-Keeping and Driver's Steering Through a 2-DOF Control Strategy." *IEEE Transactions on Control Systems Technology* 17 (1): 135–42.

Chaplier, Julien, Thomas Nguyen, Marcus Hewatt, and Gilles Gallée. 2012. "Toward a Standard: RoadXML, the Road Network Database Format." In *Actes INRETS*, 211–20. Paris: Institut national de recherche sur les transports et leur sécurité.

Chen, Yi-Liang, Venkataraman Sundareswaran, Craig Anderson, Alberto Broggi, Paolo Grisleri, Pier Paolo Porta, Paolo Zani, and John Beck. 2009. "TerraMax: Team Oshkosh Urban Robot." In *The DARPA Urban Challenge*, edited by Martin Buehler, Karl Iagnemma, and Sanjiv Singh, 595–622. Springer Tracts in Advanced Robotics 56. Springer Berlin Heidelberg.

Chipalkatty, Rahul, and Magnus Egerstedt. 2010. "Human-in-the-Loop: Terminal Constraint Receding Horizon Control with Human Inputs." In *2010 IEEE International Conference on Robotics and Automation (ICRA)*, 2712–17.

Choudhury, Charisma, Varun Ramanujam, and Moshe Ben-Akiva. 2009. "Modeling Acceleration Decisions for Freeway Merges." *Transportation Research Record: Journal of the Transportation Research Board* 2124 (December): 45–57.

Clementini, Eliseo, Paolino Di Felice, and Daniel Hernández. 1997. "Qualitative Representation of Positional Information." *Artificial Intelligence* 95 (2): 317–56.

COCOVEA. 2013. "CoCoVeA: Coopération Conducteur-Véhicule Automatisé." Accessed April 05, 2017. http://www.univ-valenciennes.fr/cocovea/en/presentation.

Constantine, Larry L., and Lucy A. D. Lockwood. 2001. "Structure and Style in Use Cases for User Interface Design." In *Object Modeling and User Interface Design*, edited by Mark Van Harmelen, 245–279. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.

Dickmann, Jürgen, Nils Appenrodt, and Carsten Brenk. 2014. "How We Gave Sight to the Mercedes Robotic Car." *IEEE Spectrum: Technology, Engineering, and Science News*. July 24. Accessed April 05, 2017. http://spectrum.ieee.org/transportation/self-driving/how-we-gave-sight-to-the-mercedes-robotic-car.

Dickmanns, Ernst D. 1988. "Dynamic Computer Vision for Mobile Robot Control." In *19. International Symposium on Industrial Robots*, 314–27.

———. 2007. *Dynamic Vision for Perception and Control of Motion*. Secaucus, NJ, USA: Springer-Verlag New York, Inc.

Dijke, Jan van, and Margriet van Schijndel. 2012. "CityMobil, Advanced Transport for the Urban Environment: Update." *Transportation Research Record: Journal of the Transportation Research Board* 2324 (December): 29–36.

Dokic, Jadranka, Beate Müller, and Gereon Meyer. 2015. "European Roadmap Smart Systems for Automated Driving." European Technology Platform on Smart Systems Integration (EPoSS). Accessed April 05, 2017. http://www.smart-systems-integration.org/public/documents/publications/EPoSS%20Roadmap_Smart%20Systems%20for%20Automated%20Driving_V2_April%202015.pdf.

"Dr SiHMI Platform." 2016. Accessed December 18, 2016. http://www.irt-systemx.fr/en/valorisation/plateformes/plateforme-dr-sihmi/.

Dragan, Anca D, and Siddhartha S Srinivasa. 2013. "A Policy-Blending Formalism for Shared Control." *International Journal of Robotics Research*. 32 (7): 790–805.

Enache, N. Minoiu, Saïd Mammar, Mariana Netto, and Benoit Lusetti. 2010. "Driver Steering Assistance for Lane-Departure Avoidance Based on Hybrid Automata and Composite Lyapunov Function." *IEEE Transactions on Intelligent Transportation Systems* 11 (1): 28–39.

Endsley, Mica R. 1988. "Design and Evaluation for Situation Awareness Enhancement." *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 32 (2): 97–101.

Endsley, Mica R., and David B Kaber. 1999. "Level of Automation Effects on Performance, Situation Awareness and Workload in a Dynamic Control Task." *Ergonomics* 42 (3): 462–92.

Endsley, Mica R., and Esin O. Kiris. 1995. "The Out-of-the-Loop Performance Problem and Level of Control in Automation." *Human Factors: The Journal of the Human Factors and Ergonomics Society* 37 (2): 381–94.

Erlien, Stephen M., Susumu Fujita, and Joseph C. Gerdes. 2016. "Shared Steering Control Using Safe Envelopes for Obstacle Avoidance and Vehicle Stability." *IEEE Transactions on Intelligent Transportation Systems* 17 (2): 441–51.

Falcone, Paolo, Francesco Borrelli, Jahan Asgari, Hongtei E. Tseng, and Davor Hrovat. 2007. "Predictive Active Steering Control for Autonomous Vehicle Systems." *IEEE Transactions on Control Systems Technology* 15 (3): 566–80. doi:10.1109/TCST.2007.894653.

Falcone, Paolo., Manuela Tufo, Francesco Borrelli, Jahan. Asgari, and Hongtei E. Tseng. 2007. "A Linear Time Varying Model Predictive Control Approach to the Integrated Vehicle Dynamics Control Problem in Autonomous Systems." In *2007 46th IEEE Conference on Decision and Control*, 2980–85. doi:10.1109/CDC.2007.4434137.

Färber, Berthold. 2016. "Communication and Communication Problems between Autonomous Vehicles and Human Drivers." In *Autonomous Driving*, edited by Markus Maurer, J. Christian Gerdes, Barbara Lenz, and Hermann Winner, 125–44. Springer Berlin Heidelberg.

Flemisch, Frank, Matthias Heesen, Tobias Hesse, Johann Kelsch, Anna Schieben, and Johannes Beller. 2012. "Towards a Dynamic Balance between Humans and Automation: Authority, Ability, Responsibility and Control in Shared and Cooperative Control Situations." *Cognition, Technology & Work* 14 (1): 3–18.

Flemisch, Frank, Fawzi Nashashibi, Nadja Rauch, Anna Schieben, Sebastien Glaser, Gerald Temme, Paulo Resende, et al. 2010. "Towards Highly Automated Driving: Intermediate Report on the HAVEit-Joint System." In *3rd European Road Transport Research Arena (TRA 2010)*. Accessed April 05, 2017. https://hal.inria.fr/inria-00533483/document.

Flemisch, Frank O., Catherine A. Adams, Sheila R. Conway, Ken H. Goodrich, Michael T. Palmer, and Paul C. Schutte. 2003. "The H-Metaphor as a Guideline for Vehicle Automation and Interaction." Report (No. NASA/TM—2003-212672). Hampton, VA, USA : NASA. Accessed April 05, 2017. http://ntrs.nasa.gov/search.jsp?R=20040031835.

Franz, Benjamin, Michaela Kauer, Anton Blanke, Michael Schreiber, Ralph Bruder, and Sebastian Geyer. 2012. "Comparison of Two Human-Machine Interfaces for Cooperative Maneuver-Based Driving." *Work* 41 (Supplement 1): 4192–99.

Fu, Xiaoxin, Yongheng Jiang, Geng Lu, Jingchun Wang, Dexian Huang, and Danya Yao. 2014. "Probabilistic Trajectory Prediction in Intelligent Driving." *IFAC Proceedings Volumes*, 19th IFAC World Congress, 47 (3): 2664–72.

Gasser, Tom M, Clemens Arzt, Mihiar Ayoubi, Arne Bartels, Lutz Bürkle, Jana Eier, Frank Flemisch, et al. 2009. "Legal Consequences of an Increase in Vehicle Automation." Report (No. BASt-Report F83). Bergisch Gladbach, Germany: Die Bundesanstalt Für Strassenwesen (BASt).

Glaser, Sébastien. 2013. "ABV - Automatisation Basse Vitesse: Compte-Rendu de Fin de Projet." Report (No. ANR-09-VTT-01). Accessed April 05, 2017. http://serres.ifsttar.fr/fileadmin/contributeurs/serres/Action4/Compte-rendu_final_ABV_v10.pdf.

Glaser, Sébastien, Benoit Vanholme, Saïd Mammar, Dominique Gruyer, and Lydie Nouvelière. 2010. "Maneuver-Based Trajectory Planning for Highly Autonomous Vehicles on Real Road With Traffic and Driver Interaction." *IEEE Transactions on Intelligent Transportation Systems* 11 (3): 589–606.

Gold, Christian, Daniel Damböck, Lutz Lorenz, and Klaus Bengler. 2013. "'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): 1938–42.

Goodwin, Antuan. 2016. "Nissan CEO Carlos Ghosn Talks Self-Driving Cars, EVs at New York Show." *Roadshow*. Accessed April 05, 2017. https://www.cnet.com/roadshow/news/nissan-ceo-carlos-ghosn-talks-self-driving-cars-evs/.

Gu, Tanyu, John M. Dolan, and Jin-Woo Lee. 2016. "Automated Tactical Maneuver Discovery, Reasoning and Trajectory Planning for Autonomous Driving." In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5474–80.

Guo, Chunshi, Chouki Sentouh, Jean-Christophe Popieul, Jean-Baptiste Haué, Sabine Langlois, Jean-Jacques Loeillet, Boussaad Soualmi, and Thomas Nguyen. 2016. "Cooperation between Driver and Automated Driving System: Implementation and Evaluation." In *DSC 2016 Europe Driving Simulation Conference & Exhibition*.

Guo, Chunshi, Chouki Sentouh, Boussaad Soualmi, Jean-Baptiste Haué, and Jean-Christophe Popieul. 2015. "A Hierarchical Cooperative Control Architecture for Automated Driving Systems." In *ITS World Congress 2015*. Bordeaux, France.

———. 2016. "Adaptive Vehicle Longitudinal Trajectory Prediction for Automated Highway Driving." In *2016 IEEE Intelligent Vehicles Symposium (IV)*, 1279–84.

Gurobi Optimization, Inc. 2017. "Gurobi Optimization - The Best Mathematical Programming Solver." Accessed January 13, 2017. http://www.gurobi.com/home?utm_expid=11945996-34.HBhfjyzcSu6g184-mX_hxQ.1&utm_referrer=https%3A%2F%2Fwww.google.fr%2F.

Handelsblatt Global. 2015. "The New E-Class – in the Fast Lane with Active Lane Change Assist." Handelsblatt Global Edition. Accessed December 15, 2016. https://global.handelsblatt.com//the-new-e-class-in-the-fast-lane-with-active-lane-change-assist-396283.

Hidas, Peter. 2005. "Modelling Vehicle Interactions in Microscopic Simulation of Merging and Weaving." *Transportation Research Part C: Emerging Technologies* 13 (1): 37–62.

Highway Loss Data Institute. 2016. “Predicted Availability of Safety Features on Registered Vehicles: A 2016 Update.” HLDI Bulletin. Arlington, VA.: Highway Loss Data Institute. Accessed April 05, 2017. http://www.iihs.org/media/31d3dcc6-79d5-48a8-bafb-1e93df1fb16f/pVFLIw/HLDI%20Research/Bulletins/hldi_bulletin_31_15.pdf.

Hoc, Jean-Michel. 1996. *SUPERVISION ET CONTROLE DE PROCESSUS. La cognition en situation dynamique*. Grenoble, France: Presses Universitaires de Grenoble.

———. 2001. “Towards a Cognitive Approach to Human–machine Cooperation in Dynamic Situations.” *International Journal of Human-Computer Studies* 54 (4): 509–40.

———. 2012. “PARTAGE: Contrôle Partagé Entre Conducteur et Assistance À La Conduite Automobile Pour Une Trajectoire Sécurisée.” Report (No. ANR-08-VTT-012-01). Accessed April 05, 2017. http://www.predit.prd.fr/predit4/documentFo.fo?cmd=visualize&inCde=43999.

Hoc, Jean-Michel, Pietro C. Cacciabue, Erik Hollnagel, and P. Carlo Cacciabue, eds. 1994. *Expertise and Technology: Cognition & Human-Computer Cooperation*. Hillsdale, N.J: Psychology Press.

Hoc, Jean-Michel, Mark S. Young, and Jean-Marc Blosseville. 2009. “Cooperation between Drivers and Automation: Implications for Safety.” *Theoretical Issues in Ergonomics Science* 10 (2): 135–60. doi:10.1080/14639220802368856.

Hoeger, Reiner, Angelos Amditis, Martin Kunert, Alfred Hoess, F Flemisch, Hans-Peter Krueger, Arne Bartels, Achim Beutner, and K Pagle. 2008. “Highly Automated Vehicles for Intelligent Transport: HAVEit Approach.” In *ITS World Congress 2008, NY, USA*.

Hogan, Neville. 1985. “Impedance Control: An Approach to Manipulation: Part I—Theory.” *Journal of Dynamic Systems, Measurement, and Control* 107 (1): 1–7.

Hollnagel, Erik, and David Woods. 1999. “Cognitive Systems Engineering: New Wine in New Bottles.” *International Journal of Human-Computer Studies* 51 (2): 339–56.

Horrell, Paul. 2017. “VW to Build Self-Driving Cars ‘Faster than Competition.’” *Top Gear*. Accessed February 25, 2017. http://www.topgear.com/car-news/geneva-motor-show/vw-build-self-driving-cars-faster-competition.

Hörwick, Markus, and Karl-Heinz Siedersberger. 2010. “Strategy and Architecture of a Safety Concept for Fully Automatic and Autonomous Driving Assistance Systems.” In *2010 IEEE Intelligent Vehicles Symposium*, 955–60.

Houenou, Aadam, Philippe Bonnifait, Veronique Cherfaoui, and Wen Yao. 2013. “Vehicle Trajectory Prediction Based on Motion Model and Maneuver Recognition.” In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4363–69.

Houska, Boris, Hans Joachim Ferreau, and Moritz Diehl. 2011. “An Auto-Generated Real-Time Iteration Algorithm for Nonlinear MPC in the Microsecond Range.” *Automatica* 47 (10): 2279–85.

Howard, Thomas M., Colin J. Green, and Alonzo Kelly. 2010. “Receding Horizon Model-Predictive Control for Mobile Robot Navigation of Intricate Paths.” In *Field and Service Robotics*, edited by Andrew Howard, Karl Iagnemma, and Alonzo Kelly, 69–78. Springer Tracts in Advanced Robotics 62. Springer Berlin Heidelberg.

Huang, Jihua, and Han-Shue Tan. 2006. "Vehicle Future Trajectory Prediction with a DGPS/INS-Based Positioning System." In *2006 American Control Conference*, 5831–36.

Iagnemma, Karl, and Martin Buehler. 2006. "Editorial for Journal of Field Robotics—Special Issue on the DARPA Grand Challenge: Editorial." *Journal of Intelligent & Robotic Systems*. 23 (9): 655–656.

Ibañez-Guzmán, Javier, Christian Laugier, John-David Yoder, and Sebastian Thrun. 2012. "Autonomous Driving: Context and State-of-the-Art." In *Handbook of Intelligent Vehicles*, edited by Azim Eskandarian, 1271–1310. Springer London.

Inagaki, Toshiyuki. 2003. "Adaptive Automation: Sharing and Trading of Control." *Handbook of Cognitive Task Design* 8: 147–169.

Ioannou, Petros A, and Cheng-Chih Chien. 1993. "Autonomous Intelligent Cruise Control." *IEEE Transactions on Vehicular Technology* 42 (4): 657–72.

ISO 9241-210:2010. "ISO 9241-210, Ergonomics of Human-System Interaction—Part 210: Human-Centred Design for Interactive Systems." Geneva, Switzerland: ISO. Accessed April 05, 2017. http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=16883.

ISO 17387:2008. "ISO 17387:2008, Intelligent Transport Systems—Lane Change Decision Aid Systems (LCDAS)—Performance Requirements and Test Procedures." Geneva, Switzerland: ISO. Accessed April 05, 2017. http://www.iso.org/iso/catalogue_detail.htm?csnumber=43654.

Itoh, Makoto, and Toshiyuki Inagaki. 2014. "Design and Evaluation of Steering Protection for Avoiding Collisions during a Lane Change." *Ergonomics* 57 (3): 361–73.

Kaber, David B., and Mica R. Endsley. 2004. "The Effects of Level of Automation and Adaptive Automation on Human Performance, Situation Awareness and Workload in a Dynamic Control Task." *Theoretical Issues in Ergonomics Science* 5 (2): 113–53.

Kammel, Sören, Julius Ziegler, Benjamin Pitzer, Moritz Werling, Tobias Gindele, Daniel Jagzent, Joachim Schöder, et al. 2009. "Team AnnieWAY's Autonomous System for the DARPA Urban Challenge 2007." In *The DARPA Urban Challenge*, edited by Martin Buehler, Karl Iagnemma, and Sanjiv Singh, 359–91. Springer Tracts in Advanced Robotics 56. Springer Berlin Heidelberg.

Kay, Steven. 1998. *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory*. 1st ed. Englewood Cliffs, N.J: Prentice Hall.

Kim, Beomjun, and Kyongsu Yi. 2014. "Probabilistic and Holistic Prediction of Vehicle States Using Sensor Fusion for Application to Integrated Vehicle Safety Systems." *IEEE Transactions on Intelligent Transportation Systems* 15 (5): 2178–90.

Kim, Sanggyum. 2012. "Design of the Adaptive Cruise Control Systems: An Optimal Control Approach." PhD diss. University of California, Berkeley. Accessed April 05, 2017. http://escholarship.org/uc/item/0v5399z9.

Klančar, Gregor, and Igor Škrjanc. 2007. "Tracking-Error Model-Based Predictive Control for Mobile Robots in Real Time." *Robotics and Autonomous Systems* 55 (6): 460–69.

KPMG. 2015. "Connected and Autonomous Vehicles – The UK Economic Opportunity." KPMG. Accessed April 05, 2017. https://www.smmt.co.uk/wp-content/uploads/sites/2/CRT036586F-Connected-and-Autonomous-Vehicles-%E2%80%93-The-UK-Economic-Opportu...1.pdf.

Kuge, Nobuyuki, Tomohiro Yamamura, Osamu Shimoyama, and Andrew Liu. 2000. "A Driver Behavior Recognition Method Based on a Driver Model Framework." *SAE Technical Paper 2000-01-0349*.

Kumar, Puneet, Mathias Perrollaz, Stéphanie Lefèvre, and Christian Laugier. 2013. "Learning-Based Approach for Online Lane Change Intention Prediction." In *2013 IEEE Intelligent Vehicles Symposium (IV)*, 797–802.

Kuwata, Yoshiaki, Justin Teo, Sertac Karaman, Gaston Fiore, Emilio Frazzoli, and Jonathan P. How. 2008. "Motion Planning in Complex Environments Using Closed-Loop Prediction." In *AIAA Guidance, Navigation and Control Conference and Exhibit*.

Laugier, Christian, Igor E. Paromtchik, Mathias Perrollaz, Mao Yong, John-David Yoder, Christopher Tay, Kamel Mekhnacha, and Amaury Nègre. 2011. "Probabilistic Analysis of Dynamic Scenes and Collision Risks Assessment to Improve Driving Safety." *IEEE Intelligent Transportation Systems Magazine* 3 (4): 4–19.

Lawitzky, Andreas, Dirk Wollherr, and Martin Buss. 2012. "Maneuver-Based Risk Assessment for High-Speed Automotive Scenarios." In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1186–91.

Lee, Suzanne E., Erik C. B. Olsen, and Walter W. Wierwille. 2004. "A Comprehensive Examination of Naturalistic Lane-Changes." Report (No. DOT HS 809 702). Washington, DC: National Highway Traffic Safety Administration. Accessed April 05, 2017. https://www.nhtsa.gov/DOT/NHTSA/NRD/Multimedia/PDFs/Crash%20Avoidance/2004/Lane%20Change%20Final.pdf.

Lefèvre, Stéphanie., Chao Sun, Ruzena Bajcsy, and Christian Laugier. 2014. "Comparison of Parametric and Non-Parametric Approaches for Vehicle Speed Prediction." In *2014 American Control Conference*, 3494–99.

Lefèvre, Stéphanie, Dizan Vasquez, and Christian Laugier. 2014. "A Survey on Motion Prediction and Risk Assessment for Intelligent Vehicles." *ROBOMECH Journal* 1 (1). doi:10.1186/s40648-014-0001-z.

Li, X. Rong., and Vesselin P. Jilkov. 2002. "Survey of Maneuvering Target Tracking: Decision-Based Methods." *Proceedings SPIE, Signal and Data Processing of Small Targets 2002*, 4728:511–34.

Da Lio, Maurio, Francesco Biral, Enrico Bertolazzi, Marco Galvani, Paolo Bosetti, David Windridge, Andrea Saroldi, and Fabio Tango. 2015. "Artificial Co-Drivers as a Universal Enabling Technology for Future Intelligent Vehicles and Transportation Systems." *IEEE Transactions on Intelligent Transportation Systems* 16 (1): 244–63.

Lofberg, Johan. 2004. "YALMIP: A Toolbox for Modeling and Optimization in MATLAB." In *2004 IEEE International Conference on Robotics and Automation (ICRA)*, 284–89.

Lorenz, Lutz, Philipp Kerschbaum, and Josef Schumann. 2014. "Designing Take over Scenarios for Automated Driving How Does Augmented Reality Support the Driver to Get Back into the Loop?" *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 58 (1): 1681–85.

"LRA | IRT SystemX." 2017. Accessed February 21, 2017. http://www.irt-systemx.fr/en/project/lra/.

Luca, A. De, G. Oriolo, and C. Samson. 1998. "Feedback Control of a Nonholonomic Car-like Robot." In *Robot Motion Planning and Control*, edited by Dr J.-P. Laumond, 171–253. Lecture Notes in Control and Information Sciences 229. Springer Berlin Heidelberg.

Lundquist, Christian, Thomas B. Schön, and Fredrik Gustafsson. 2012. "Situational Awareness and Road Prediction for Trajectory Control Applications." In *Handbook of Intelligent Vehicles*, edited by Azim Eskandarian, 365–96. Springer London.

Mandalia, Hiren M., and Mandalia Dario D. Salvucci. 2005. "Using Support Vector Machines for Lane-Change Detection." *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 49 (22): 1965–69.

Markoff, John. 2010. "Google Cars Drive Themselves, in Traffic." *The New York Times*, October 9. Accessed April 05, 2017. http://www.nytimes.com/2010/10/10/science/10google.html.

Martinez, John-Jairo, and Carlos Canudas-de-Wit. 2007. "A Safe Longitudinal Control for Adaptive Cruise Control and Stop-and-Go Scenarios." *IEEE Transactions on Control Systems Technology* 15 (2): 246–58.

Matarić, Maja J., and François Michaud. 2008. "Behavior-Based Systems." In *Springer Handbook of Robotics*, edited by Bruno Siciliano Prof and Oussama Khatib Prof, 891–909. Springer Berlin Heidelberg.

MathWorks. 2016. "What Is an S-Function? - MATLAB & Simulink." Accessed October 24, 2016. https://fr.mathworks.com/help/simulink/sfg/what-is-an-s-function.html.

Maturana, Humberto R., and Francisco J. Varela. 1992. *The Tree of Knowledge: The Biological Roots of Human Understanding*. Revised edition. Boston : New York: Shambhala.

Mercedes Benz. 2015. "Intelligent Drive Next Level as Part of Driving Assistance Package." *Mercedes-Benz.com*. July 16. Accessed April 05, 2017. https://www.mercedes-benz.com/en/mercedes-benz/innovation/with-intelligent-drive-more-comfort-in-road-traffic/.

Michon, John A. 1985. "A Critical View of Driver Behavior Models: What Do We Know, What Should We Do?" In *Human Behavior and Traffic Safety*, edited by Leonard Evans and Richard C. Schwing, 485–524. Springer US.

Miller, Isaac, Mark Campbell, Dan Huttenlocher, Aaron Nathan, Frank-Robert Kline, Pete Moran, Noah Zych, et al. 2009. "Team Cornell's Skynet: Robust Perception and Planning in an Urban Environment." In *The DARPA Urban Challenge*, edited by Martin Buehler, Karl Iagnemma, and Sanjiv Singh, 257–304. Springer Tracts in Advanced Robotics 56. Springer Berlin Heidelberg.

Millot, P, and A Kamoun. 1988. "An Implicit Method for Dynamic Task Allocation between Man and Computer in Supervision Posts of Automated Processes." In *3rd IFAC/IFIP/IEA/IFORS Conference on Analysis Design and Evaluation of Man-Machine Systems*, 77–82. Oulu, Finland.

Mitschke, Manfred, and Henning Wallentowitz. 2004. *Dynamik der Kraftfahrzeuge*. 4th ed. Berlin: Springer.

Montemerlo, Michael, Jan Becker, Suhrid Bhat, Hendrik Dahlkamp, Dmitri Dolgov, Scott Ettinger, Dirk Haehnel, et al. 2009. "Junior: The Stanford Entry in the Urban Challenge." In *The DARPA Urban Challenge*, edited by Martin Buehler, Karl Iagnemma, and Sanjiv Singh, 91–123. Springer Tracts in Advanced Robotics 56. Springer Berlin Heidelberg.

Mörtl, Alexander, Martin Lawitzky, Ayse Kucukyilmaz, Metin Sezgin, Cagatay Basdogan, and Sandra Hirche. 2012. "The Role of Roles: Physical Cooperation between Humans and Robots." *The International Journal of Robotics Research* 31 (13): 1656–74.

Mulder, Mark, David A. Abbink, Marinus M. van Paassen, and Max Mulder. 2011. "Design of a Haptic Gas Pedal for Active Car-Following Support." *IEEE Transactions on Intelligent Transportation Systems* 12 (1): 268–79.

Nardi, Daniele, and Ronald J. Brachman. 2003. "An Introduction to Description Logics." In *The Description Logic Handbook*, edited by Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, 5–44. New York, NY, USA: Cambridge University Press.

Nass, Clifford, B. J. Fogg, and Youngme Moon. 1996. "Can Computers Be Teammates?" *International Journal of Human-Computer Studies.* 45 (6): 669–678.

National Highway Traffic Safety Administration. 2013. "Preliminary Statement of Policy Concerning Automated Vehicles." May 30. Accessed April 05, 2017. https://www.nhtsa.gov/staticfiles/rulemaking/pdf/Automated_Vehicles_Policy.pdf.

Navarro, J., F. Mars, and M. S. Young. 2011. "Lateral Control Assistance in Car Driving: Classification, Review and Future Prospects." *IET Intelligent Transport Systems* 5 (3): 207–20.

Nguyen, Anh-Tu, Chouki Sentouh, and Jean-Christophe Popieul. 2016. "Driver-Automation Cooperative Approach for Shared Steering Control under Multiple System Constraints: Design and Experiments." *IEEE Transactions on Industrial Electronics* PP (99): 1–1.

Nielsen, Jakob. 1994. *Usability Engineering*. New edition. Boston: Morgan Kaufmann.

Nilsson, Julia, Paolo Falcone, Mohammad Ali, and Jonas Sjöberg. 2015. "Receding Horizon Maneuver Generation for Automated Highway Driving." *Control Engineering Practice* 41 (August): 124–33.

Norman, Donald A., and Stephen W. Draper. 1986. *User Centered System Design; New Perspectives on Human-Computer Interaction*. Hillsdale, NJ, USA: L. Erlbaum Associates Inc.

Nouvelière, Lydie. 2002. "Commandes Robustes Appliquées Au Contrôle Assisté D'un Véhicule À Basse Vitesse." PhD diss. Université de Versailles-Saint-Quentin-en-Yvelines. Accessed April 05, 2017. http://www.theses.fr/2002VERS0023.

Nunen, Ellen van, M. R. J. A. E. Kwakkernaat, Jeroen Ploeg, and Bart D. Netten. 2012. "Cooperative Competition for Future Mobility." *IEEE Transactions on Intelligent Transportation Systems* 13 (3): 1018–25.

OKTAL. 2015. *SCANeR Studio User Manual--Simulation Mode*. OKTAL SAS.

Östlund, Joakim, Björn Peters, Birgitta Thorslund, Johan Engström, Gustav Markkula, Andreas Keinath, Dorit Horst, Susann Juch, Stefan Mattes, and Uli Foehl. 2005. "Driving Performance Assessment - Methods and Metrics." Report (No. Project IST-1-507674-IP, Deliverable 2.2.5). EU project AIDE. Accessed April 05, 2017. http://www.aide-eu.org/res_sp2.html.

Paden, Brian, Michal Čáp, Sze Zheng Yong, Dmitry Yershov, and Emilio Frazzoli. 2016. "A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles." *IEEE Transactions on Intelligent Vehicles* 1 (1): 33–55.

Pandey, Mohinder, and Yosef Dalbah. 2014. "A Method for Reducing False Warnings in Collision Warning System (CWS) during Turning Maneuvers at Road Intersection." In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, 310–16.

Papp, Zoltán. 2012. "Situational Awareness in Intelligent Vehicles." In *Handbook of Intelligent Vehicles*, edited by Azim Eskandarian, 61–80. Springer London.

Papp, Zoltán, Christopher Brown, and Christine Bartels. 2008. "World Modeling for Cooperative Intelligent Vehicles." In *2008 IEEE Intelligent Vehicles Symposium*, 1050–55.

Parasuraman, R., T. B. Sheridan, and C. D. Wickens. 2000. "A Model for Types and Levels of Human Interaction with Automation." *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 30 (3): 286–97.

Parasuraman, Raja, and Victor Riley. 1997. "Humans and Automation: Use, Misuse, Disuse, Abuse." *Human Factors: The Journal of the Human Factors and Ergonomics Society* 39 (2): 230–53.

Park, J., S. Karumanchi, and Karl Iagnemma. 2015. "Homotopy-Based Divide-and-Conquer Strategy for Optimal Trajectory Planning via Mixed-Integer Programming." *IEEE Transactions on Robotics* 31 (5): 1101–15.

Patz, Benjamin J., Yiannis Papelis, Remo Pillat, Gary Stein, and Don Harper. 2009. "A Practical Approach to Robotic Design for the DARPA Urban Challenge." In *The DARPA Urban Challenge*, edited by Martin Buehler, Karl Iagnemma, and Sanjiv Singh, 305–58. Springer Tracts in Advanced Robotics 56. Springer Berlin Heidelberg.

Pirjanian, Paolo. 1999. "Behavior Coordination Mechanisms—State-of-the-Art." Report (No. IRIS-99–375). University of Southern California, Institute for Robotics and Intelligent Systems.

Polychronopoulos, A., M. Tsogas, A. Amditis, U. Scheunert, L. Andreone, and F. Tango. 2004. "Dynamic Situation and Threat Assessment for Collision Warning Systems: The EUCLIDE Approach." In *2004 IEEE Intelligent Vehicles Symposium*, 636–41.

Pritchard, Justin, and Tom Krisher. 2016. "Self-Driving Cars Go Public; Uber Offers Rides in Pittsburgh." *The Big Story*. August 2016. Accessed April 05, 2017. http://bigstory.ap.org/article/e92ea7b2f13745f1b6f0c7bc96da9c47/uber-use-autonomous-cars-haul-people-next-few-weeks.

Qian, Xiangjun, Florent Altché, Philipp Bender, Christoph Stiller, and Arnaud de La Fortelle. 2016. "Optimal Trajectory Planning for Autonomous Driving Integrating Logical Constraints: An MIQP Perspective." In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 205–10.

Quain, John R. 2016. "How to Use Tesla's Autopilot (and How Not to)." *Tom's Guide*. September 9. Accessed April 05, 2017. http://www.tomsguide.com/us/how-to-use-tesla-autopilot,review-3870.html.

Rajamani, Rajesh. 2006. *Vehicle Dynamics and Control*. Mechanical Engineering Series. New York: Springer-Verlag.

Rasmussen, Jens. 1983. "Skills, Rules, and Knowledge; Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models." *IEEE Transactions on Systems, Man, and Cybernetics* SMC-13 (3): 257–66.

Reichel, Michael, Michael Botsch, Robert Rauschecker, Karl-Heinz Siedersberger, and Markus Maurer. 2010. "Situation Aspect Modelling and Classification Using the Scenario Based Random Forest Algorithm for Convoy Merging Situations." In *13th International IEEE Conference on Intelligent Transportation Systems*, 360–66.

Reinholtz, Charles, Dennis Hong, Al Wicks, Andrew Bacha, Cheryl Bauman, Ruel Faruque, Michael Fleming, et al. 2009. "Odin: Team VictorTango's Entry in the DARPA Urban

Challenge." In *The DARPA Urban Challenge*, edited by Martin Buehler, Karl Iagnemma, and Sanjiv Singh, 125–62. Springer Tracts in Advanced Robotics 56. Springer Berlin Heidelberg.

Riekert, P., and T. E. Schunck. 1940. "Zur Fahrmechanik des gummibereiften Kraftfahrzeugs." *Ingenieur-Archiv* 11 (3): 210–24.

Rios-Torres, Jackeline, and Andreas. A. Malikopoulos. 2016. "A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps." *IEEE Transactions on Intelligent Transportation Systems* PP (99): 1–12.

Russell, Stuart, and Peter Norvig. 2009. *Artificial Intelligence: A Modern Approach*. 3 edition. Upper Saddle River, NJ: Pearson.

SAE. 2015. "Operational Definitions of Driving Performance Measures and Statistics." Standard J2944_201506. SAE International.

———. 2016. "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles." Standard J3016_201609. SAE International.

Saleh, Louay, Philippe Chevrel, Fabien Claveau, Jean-Francois Lafay, and Franck Mars. 2013. "Shared Steering Control Between a Driver and an Automation: Stability in the Presence of Driver Behavior Uncertainty." *IEEE Transactions on Intelligent Transportation Systems* 14 (2): 974–83.

SARTRE. 2012. "SARTRE: Safe Road Trains for the Environment." Accessed April 05, 2017. http://www.sartre-project.eu/en/Sidor/default.aspx.

Scarinci, Riccardo, and Benjamin Heydecker. 2014. "Control Concepts for Facilitating Motorway On-Ramp Merging Using Intelligent Vehicles." *Transport Reviews* 34 (6): 775–97.

Schmidt, K. 1991. "Cooperative Work: A Conceptual Framework." *Distributed Decision Making. Cognitive Models for Cooperative Work*, 75–110.

Schubert, R., E. Richter, and G. Wanielik. 2008. "Comparison and Evaluation of Advanced Motion Models for Vehicle Tracking." In *2008 11th International Conference on Information Fusion*, 1–6.

SENSODRIVE. 2017. "Senso-Wheel SD-LC - the Perfect One." Accessed February 14, 2017. https://www.sensodrive.de/EN/products/Force-Feedback-Wheels/Senso-Wheel-SD-LC.php.

Sentouh, Chouki, Philippe. Chevrel, Frank Mars, and Fabien Claveau. 2009. "A Sensorimotor Driver Model for Steering Control." In *2009 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2462–67.

Sentouh, Chouki., Boussaad. Soualmi, Jean-Christophe Popieul, and S. Debernard. 2013. "Cooperative Steering Assist Control System." In *2013 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 941–46.

Sentouh, Chouki, Jean-Christophe Popieul, Serge Debernard, and Serge Boverie. 2014. "Human-Machine Interaction in Automated Vehicle: The ABV Project." *IFAC Proceedings Volumes*, 19th IFAC World Congress, 47 (3): 6344–49.

Sheridan, Thomas B. 1992. *Telerobotics, Automation, and Human Supervisory Control*. Cambridge, MA, USA: MIT Press.

Sheridan, Thomas B., and William L. Verplank. 1978. "Human and Computer Control of Undersea Teleoperators." Report. Cambridge, MA: Massachusetts Institute of Technology.

Shladover, Steven E., Charles A. Desoer, J. Karl Hedrick, Masayoshi Tomizuka, Jean Walrand, Wei Bin Zhang, Donn H. McMahon, Huei Peng, Shahab Sheikholeslam, and Nick McKeown. 1991. "Automated Vehicle Control Developments in the PATH Program." *IEEE Transactions on Vehicular Technology* 40 (1): 114–30.

Simon, Herbert A. 1996. *The Sciences of the Artificial*. 3rd ed. Cambridge, MA: The MIT Press.

Soualmi, Boussaad., Chouki Sentouh, Jean-Christophe Popieul, and Serge Debernard. 2014. "Automation-Driver Cooperative Driving in Presence of Undetected Obstacles." *Control Engineering Practice* 24 (March): 106–19.

Sowa, John F. 1999. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Pacific Grove: Course Technology Inc.

Suchman, Lucy A. 1987. *Plans and Situated Actions: The Problem of Human-Machine Communication*. Cambridge university press.

Suh, Jongsang, Kyongsu Yi, Jiyeol Jung, Kyungjun Lee, Hyokjin Chong, and Bongchul Ko. 2016. "Design and Evaluation of a Model Predictive Vehicle Control Algorithm for Automated Driving Using a Vehicle Traffic Simulator." *Control Engineering Practice* 51 (June): 92–107.

Takada, Yuji, Erwin R Boer, and Tetsuo Sawaragi. 2013. "Driving Assist System: Shared Haptic Human System Interaction." *IFAC Proceedings Volumes* 46 (15): 203–210.

Takahashi, A., T. Hongo, Y. Ninomiya, and G. Sugimoto. 1989. "Local Path Planning And Motion Control For Agv In Positioning." In *IEEE/RSJ International Workshop on Intelligent Robots and Systems '89. The Autonomous Mobile Robots and Its Applications (IROS '89)*, 392–97.

Taş, Ömer Ş, Florian Kuhnt, J. Mairus Zöllner, and Christoph Stiller. 2016. "Functional System Architectures towards Fully Automated Driving." In *2016 IEEE Intelligent Vehicles Symposium (IV)*, 304–9.

That, Thomas Nguen, and Jordi Casas. 2011. "An Integrated Framework Combining a Traffic Simulator and a Driving Simulator." *Procedia - Social and Behavioral Sciences*, The State of the Art in the European Quantitative Oriented Transportation and Logistics Research – 14th Euro Working Group on Transportation & 26th Mini Euro Conference & 1st European Scientific Conference on Air Transport, 20 (January): 648–55.

The MathWorks, Inc. 2016a. "Generate Random Numbers That Are Repeatable - MATLAB & Simulink." Accessed December 19, 2016. https://www.mathworks.com/help/matlab/math/generate-random-numbers-that-are-repeatable.html.

———. 2017b. "Stateflow - MATLAB & Simulink." Accessed January 13, 2017. https://www.mathworks.com/products/stateflow.html.

Theureau, Jacques, and François Jeffroy. 1994. *Ergonomie des situations informatisées. La conception centrée sur le cours d'action de l'utilisateurs*. Premiere edition. Toulouse: Octarès Editions.

Thrun, Sebastian, Wolfram Burgard, and Dieter Fox. 2005. *Probabilistic Robotics*. Intelligent Robotics and Autonomous Agents series edition. Cambridge, Mass: The MIT Press.

Ulbrich, Simon, and Markus Maurer. 2015. "Situation Assessment in Tactical Lane Change Behavior Planning for Automated Vehicles." In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 975–81.

Ulbrich, Simon, Till Menzel, Andreas Reschka, Fabian Schuldt, and Markus Maurer. 2015. "Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving." In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 982–88. doi:10.1109/ITSC.2015.164.

Urdiales, Cristina, Manuel Fernandez-Carmona, Jose M. Peula, R. Annicchiaricco, F. Sandoval, and Carlo Caltagirone. 2010. "Efficiency Based Modulation for Wheelchair Driving Collaborative Control." In *2010 IEEE International Conference on Robotics and Automation (ICRA)*, 199–204.

Urmson, Chris, Joshua Anhalt, Drew Bagnell, Christopher Baker, Robert Bittner, M. N. Clark, John Dolan, et al. 2009. "Autonomous Driving in Urban Environments: Boss and the Urban Challenge." In *The DARPA Urban Challenge*, edited by Martin Buehler, Karl Iagnemma, and Sanjiv Singh, 1–59. Springer Tracts in Advanced Robotics 56. Springer Berlin Heidelberg.

Vanholme, Benoit, Dominique Gruyer, Benoit Lusetti, Sébastien Glaser, and Saïd Mammar. 2013. "Highly Automated Driving on Highways Based on Legal Safety." *IEEE Transactions on Intelligent Transportation Systems* 14 (1): pp 333-347.

Vermersch, Pierre. 1994. *L'entretien D'explicitation*. Paris: Editions ESF.

Wakasugi, Takashi. 2005. "A Study on Warning Timing for Lane Change Decision Aid Systems Based on Driver's Lane Change Maneuver." In *Proc. 19th International Technical Conference on the Enhanced Safety of Vehicles*.

Walch, Marcel, Kristin Lange, Martin Baumann, and Michael Weber. 2015. "Autonomous Driving: Investigating the Feasibility of Car-Driver Handover Assistance." In *Proceedings of the 7th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, 11–18. AutomotiveUI '15. New York, NY, USA: ACM.

Wegschweider, M., and G. Prokop. 2005. "Modellbasierte Komfortbewertung von Fahrerassistenzsystemen / Model-Based Comfort Evaluation of Driver Assistance Systems." *VDI-Berichte*, no. 1900.

Wei, Junqing, John M. Dolan, and Bakhitar Litkouhi. 2013. "Autonomous Vehicle Social Behavior for Highway Entrance Ramp Management." In *2013 IEEE Intelligent Vehicles Symposium (IV)*, 201–7. doi:10.1109/IVS.2013.6629471.

Werling, M., J. Ziegler, S. Kammel, and S. Thrun. 2010. "Optimal Trajectory Generation for Dynamic Street Scenarios in a Frenet Frame." In *2010 IEEE International Conference on Robotics and Automation (ICRA)*, 987–93.

White, Christopher E, David Bernstein, and Alain L Kornhauser. 2000. "Some Map Matching Algorithms for Personal Navigation Assistants." *Transportation Research Part C: Emerging Technologies* 8 (1–6): 91–108.

Willmore, Thomas J. 2012. *An Introduction to Differential Geometry*. Reprint edition. New York: Dover Publications.

Winner, Hermann. 2012. "Adaptive Cruise Control." In *Handbook of Intelligent Vehicles*, edited by Azim Eskandarian, 613–56. Springer London.

Young, Mark S., Neville A. Stanton, and Don Harris. 2007. "Driving Automation: Learning from Aviation about Design Philosophies." *International Journal of Vehicle Design* 45 (3): 323.

Ziegler, Julius, Philipp Bender, Markus Schreiber, Henning Lategahn, Tobias Strauss, Christoph Stiller, Thao Dang, et al. 2014. "Making Bertha Drive—An Autonomous Journey on a Historic Route." *IEEE Intelligent Transportation Systems Magazine* 6 (2): 8–20.

APPENDICES

APPENDIX A TRANSFORMATIONS FROM GLOBAL CARTESIAN COORDINATES TO THE FRENET COORDINATES

We present the transformation of the state vector of a detected object from the Earth Cartesian frame to the Frenet frame, i.e.,

$$[x, y, \psi, v_x, a_x, \dot{\psi}]^{\mathrm{T}} \rightarrow [s, \dot{s}, \ddot{s}, d, \dot{d}, \ddot{d}]^{\mathrm{T}} \tag{A.1}$$

In the digital map we used, the road geometry is represented by three primitive curves: straight lines, circle arcs and clothoids. Thus, road curve $\mathbf{r}(s)$ in the map can be uniformly represented in the form of the Fresnel's integrals:

$$\mathbf{r}(s) = \left[\int_0^s \cos\big(\psi(s)\big)\, ds + x_0,\ \int_0^s \sin\big(\psi(s)\big)\, ds + y_0\right], \tag{A.2}$$

$$\psi(s) = \int_0^s \kappa(s) ds + \psi_0, \tag{A.3}$$

$$\kappa(s) = \kappa_0 + c_m s. \tag{A.4}$$

where x_0 and y_0 are initial coordinates of a curve, ψ_0 is the initial heading angle, κ_0 is the initial curvature and c_m is the curvature rate. To calculate the Frenet coordinates (*s,d*) of a point $\mathbf{x}_{2D}$, we need to find the projection of $\mathbf{x}_{2D}$ on the curve $\mathbf{r}(s)$. Let $\mathbf{r}(s^*)$ denote the projection. We can calculate s^* by inverting (A.2). For a straight line or a circular arc, a closed-form solution exists for (A.2). For a clothoid, (A.2) does not have an explicit form, therefore we need to resolve the minimization problem by numerical

$$s^* = \underset{\sigma}{\operatorname{argmin}} \ \|\mathbf{x}_{2D} - \mathbf{r}(\sigma)\|. \tag{A.5}$$

Then we can calculate the d-component of $\mathbf{x}_{2D}$, denoted by d^* via

$$d^* = \big(\mathbf{x}_{2D} - \mathbf{r}(s^*)\big)^T \mathbf{n}(s^*) \tag{A.6}$$

Time derivative of (A.6) yields

$$\begin{aligned} \dot{d} &= (\dot{\mathbf{x}}_{2D} - \dot{\mathbf{r}})^T \mathbf{n} + (\mathbf{x}_{2D} - \mathbf{r})^T \dot{\mathbf{n}} \\ &= v_x \mathbf{t}_x^T \mathbf{n} - \dot{s}\mathbf{t}\mathbf{n} - \kappa d \dot{s} \mathbf{n}^T \mathbf{t} = v_x sin\Delta\psi. \end{aligned} \tag{A.7}$$

where $\mathbf{t}_x$ is the unit tangent vector associated with the direction of v_x. We rearrange (A.6) as

$$\mathbf{x}_{2D}(s, d) = \mathbf{r}(s) + d\mathbf{n}(s). \tag{A.8}$$

Time derivative of (4.5) yields

$$\begin{aligned} \dot{\mathbf{x}}_{2D} &= \dot{\mathbf{r}} + \dot{d}\mathbf{n} + d\dot{\mathbf{n}} \\ &= \dot{s}\mathbf{t} + \dot{d}\mathbf{n} - \kappa d \dot{s}\mathbf{t} = (1 - \kappa d)\dot{s}\mathbf{t} + \dot{d}\mathbf{n}. \end{aligned} \tag{A.9}$$

Therefore, v_x can be calculated by

$$v_x = \|\dot{\mathbf{x}}_{2D}\|_2 = \sqrt{(1-\kappa d)^2\dot{s}^2 + \dot{d}^2}\,. \tag{A.10}$$

Substituting $\dot{d}$ by, we can get

$$\dot{s} = \frac{v_x cos\Delta\psi}{1-\kappa d}. \tag{A.11}$$

We calculate the second derivatives of *s* and *d* to be

$$\ddot{s} = \frac{a_x cos\Delta\psi}{1-\kappa d} + v_x \frac{-sin\Delta\psi(\dot{\psi}-\kappa\dot{s})(1-\kappa d)+cos\Delta\psi(c_m\dot{s}+\kappa\dot{d})}{(1-\kappa d)^2}\,, \tag{A.12}$$

$$\ddot{d} = a_x sin\Delta\psi + v_x cos\Delta\psi(\dot{\psi}-\kappa\dot{s}). \tag{A.13}$$

Given that κ is very small for highway roads and d is also small (since the vehicle is driving within a lane), we make the following approximation $1-\kappa d \approx 1$. Therefore, (A.11) and (A.12) become

$$\dot{s} = v_x cos\Delta\psi, \tag{A.14}$$

$$\ddot{s} = a_x cos\Delta\psi + v_x(-sin\Delta\psi(\dot{\psi}-\kappa\dot{s}) + cos\Delta\psi(c_m\dot{s}+\kappa\dot{d})). \tag{A.15}$$

At last, we rearrange (A.14) and (A.15) to give the measurement model (4.13) in Section 4.4.4.3

$$v_x = \frac{\dot{s}}{cos\Delta\psi}, \tag{A.16}$$

$$a_x = \frac{\ddot{s}}{cos\Delta\psi} + \frac{\tan\Delta\psi\dot{\psi}\dot{s}}{\cos\Delta\psi} - \frac{(2\kappa\tan\Delta\psi+c_m)\dot{s}^2}{\cos\Delta\psi}. \tag{A.17}$$

APPENDIX B STEERING SYSTEM MODELLING

In this appendix, we show how to derive the equivalent moment of inertia and damping of an EPAS system showed in Fig. B.1.

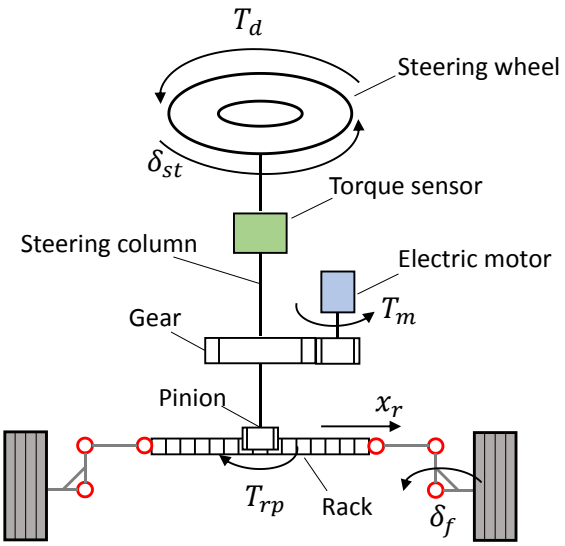


Figure B.1 Mechanical model of an EPAS system

The dynamics of the steering column is given by

$$(I_c + i_m^2 I_m)\ddot{\delta}_p + (c_c + i_m^2 c_m)\dot{\delta}_p = T_d + i_m T_m - T_{rp}, \tag{B.1}$$

where δ_p is the angle of the pinion, I_c and I_m are the moments of inertia of the steering column and the electric motor respectively, c_c and c_m are their dampings, i_m is the gear ratio, and T_{rp} is the torque at the rack-pinion. To simplify the notation, $i_m T_m$ is replaced by T_{ctrl} to represent the control of the automation.

At the rack-pinion, the force equilibrium yields

$$m_{eq}\ddot{x}_r + c_r\dot{x}_r = \frac{T_{rp}}{R_p} - F_r, \tag{B.2}$$

where m_{eq} denotes the equivalent mass of the rack and F_r is the force resulted by the self-aligning torque of the front wheels. x_r is the displacement of the rack has the following relationship with the pinion angle

$$x_r = \delta_p R_p, \tag{B.3}$$

Substituting (B.2) and (B.3) into (B.1), the equation of steering system dynamics is obtained as

$$\left(I_c + i_m^2 I_m + m_{eq}{R_p}^2\right)\ddot{\delta}_p + \left(c_c + i_m^2 c_m + c_r{R_p}^2\right)\dot{\delta}_p = T_d + T_{ctrl} - F_r R_p. \tag{B.4}$$

The relation between the steering angle of the front wheel δ_f and the pinion angle δ_p is determined by the geometry of the steering system. In practice, this relation is usually approximated by the steering ratio i_s such that

$$\delta_p = \delta_f i_s. \tag{B.5}$$

We simplify the electric motor's assisting torque as a coefficient k_p applied on the auto-alignment torque T_{al} from the tire. With this simplification, $F_r R_p$ can be obtained as

$$F_r R_p = \frac{T_{al}}{i_s k_p}. \tag{B.6}$$

Substituting (B.5) and (B.6) into (B.4),

$$i_s^2\left(I_c + i_m^2 I_m + m_{eq}{R_p}^2\right)\ddot{\delta}_f + i_s^2\left(c_c + i_m^2 c_m + c_r{R_p}^2\right)\dot{\delta}_f = i_s(T_d + T_{ctrl}) - \frac{T_{al}}{k_p} \tag{B.7}$$

Thus, the equivalent moment of inertia and damping of the steering system are

$$I_{eq} = i_s^2\left(I_c + i_m^2 I_m + m_{eq}{R_p}^2\right). \tag{B.8}$$

$$c_{eq} = i_s^2\left(c_c + i_m^2 c_m + c_r{R_p}^2\right). \tag{B.9}$$