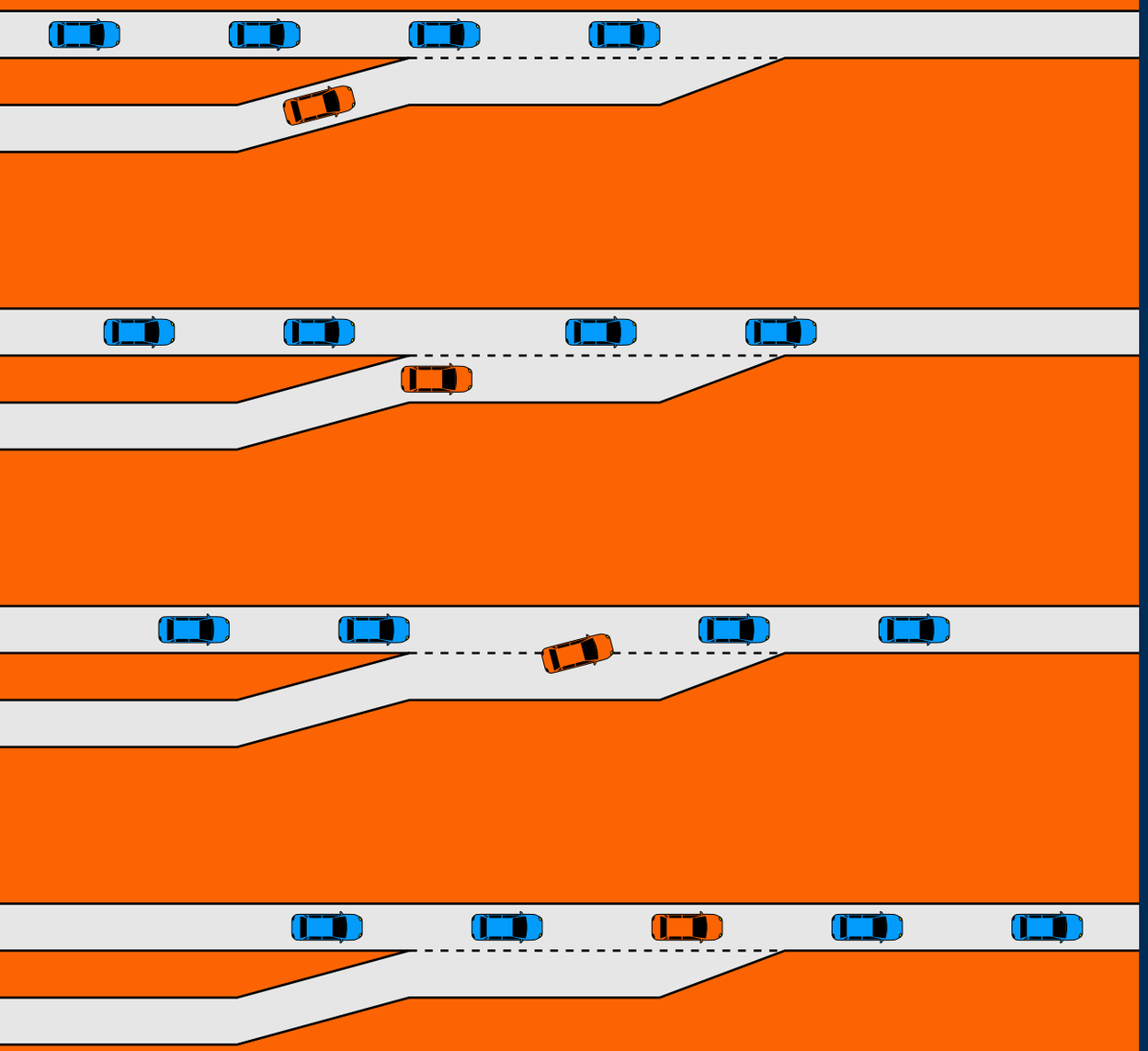
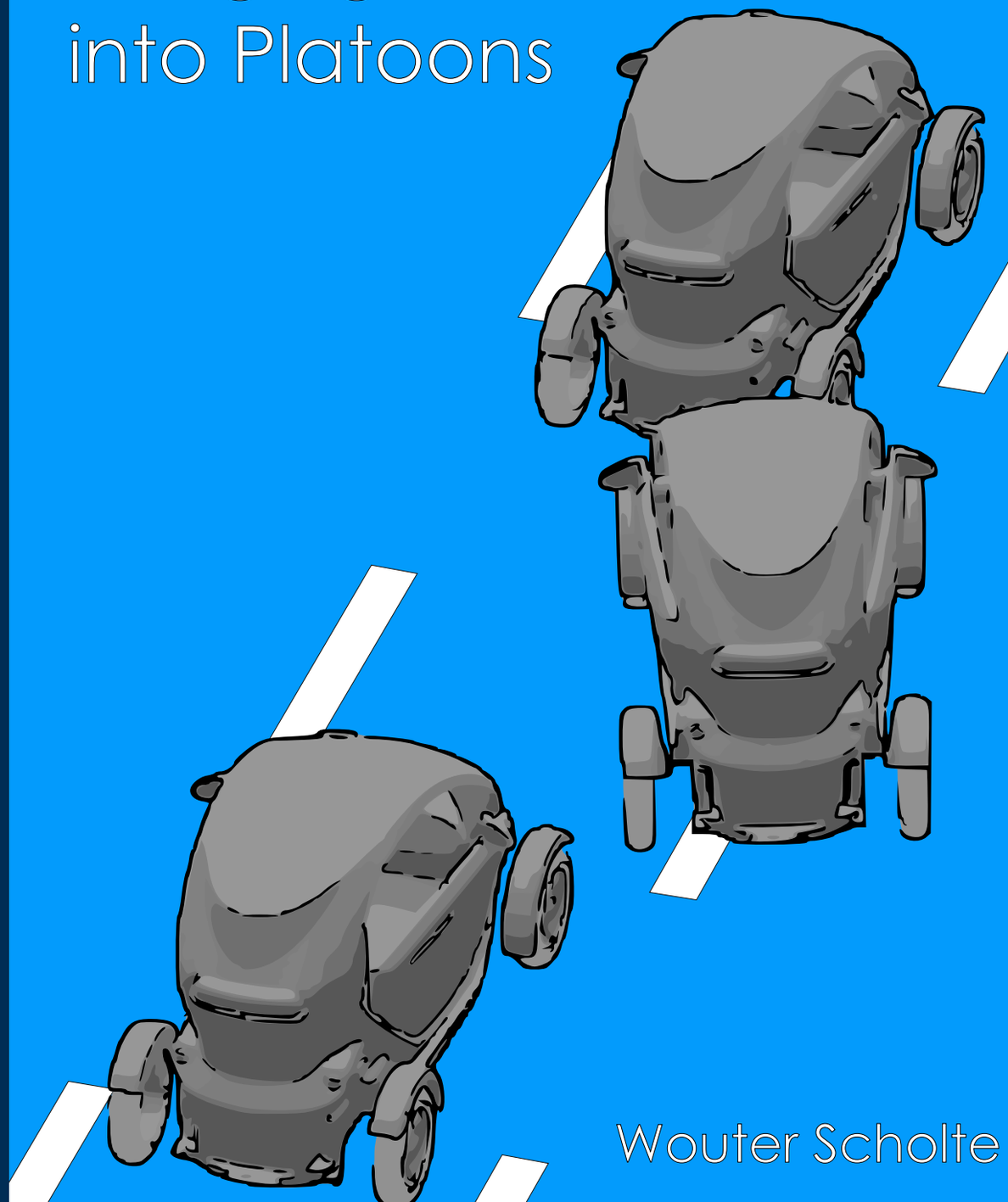
Control for Cooperative Merging Maneuvers into Platoons



Control for Cooperative Merging Maneuvers into Platoons

Wouter Scholte

Control for Cooperative Merging Maneuvers into Platoons



Wouter Scholte

Control for Cooperative Merging Maneuvers into Platoons

Wouter Scholte

The author has successfully completed the educational program of the Graduate School of the Dutch Institute of Systems and Control (DISC).

The work described in this thesis is carried out in the Dynamics & Control group at the Department of Mechanical Engineering of the Eindhoven University of Technology. The work is supported by the Dutch Organization for Scientific Research (NWO). This work has been carried out NWO as part of the i-CAVE project with project number 14893.

A catalogue record is available from the Eindhoven University of Technology Library.
ISBN: 978-90-386-5552-9

Reproduction: Ipskamp Printing, Enschede, the Netherlands

Control for Cooperative Merging Maneuvers into Platoons

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
rector magnificus prof.dr.ir. F.P.T. Baaijens, voor een
commissie aangewezen door het College voor
Promoties, in het openbaar te verdedigen
op dinsdag 13 september 2022 om 16.00 uur

door

Wouter Scholte

geboren te Nieuwegein

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

Voorzitter:	prof.dr.ir. M. Steinbuch
Promotoren:	prof.dr. H. Nijmeijer
	dr.ir. P.W.A. Zegelaar
Co-promotor:	dr.ir. T.P.J. van der Sande
Promotiecommissieleden:	prof.dr.ir. B. van Arem (Technische Universiteit Delft)
	prof.dr.ir. F.P.T Willems
	dr. P. Falcone (Chalmers University of Technology)

Het onderzoek dat in dit proefschrift wordt beschreven is uitgevoerd in overeenstemming met de TU/e Gedragscode Wetenschapsbeoefening.

Summary

Control for Cooperative Merging Maneuvers into Platoons

The interest in automated vehicles has increased in recent years. The potential benefits of automated driving include increased road safety, traffic throughput, and energy-efficient driving. In itself, automated vehicles may be comparable with human-driven vehicles in terms of behavior. However, automated vehicles can be enhanced by using inter-vehicle communication to provide additional environmental information. Such vehicles are typically referred to as Connected Automated Vehicles (CAVs).

A possible additional feature of CAVs, obtained by using inter-vehicle communication, is Cooperative Adaptive Cruise Control (CACC). This technique allows a string of vehicles to maintain small inter-vehicle distances by considering communicated inputs from preceding vehicles. The small inter-vehicle distances improve traffic throughput and potentially energy efficiency, while the controller aims to guarantee road safety. An ideal scenario for CACC is highway driving, due to the long roads with limited curvatures and relatively few entrances and exits. The string of vehicles, which is also referred to as a platoon, will thus experience relatively few external disturbances while driving. Nevertheless, new vehicles may enter the highway and will need to merge with the platoon. An easy solution would be by joining at the back of the platoon. However, if the new vehicle joins at a highway entrance, the length of the on-ramp may restrict this possibility. Due to the spatial constraint, the new vehicle is then forced to join between two platoon vehicles. The short inter-vehicle distances typically do not allow this. This thesis presents a control strategy for the merging of a single CAV into a platoon.

To address the merging problem, a controller for the individual vehicles is needed. In this research, a variable gap controller is designed based on a conventional CACC controller. This can be used to create a gap in the platoon to accommodate the merging vehicle. Furthermore, this controller can easily be adapted to initialize the conventional CACC controller. The desired inter-vehicle gap can be initialized as the measured gap and then be brought to that of a conventional CACC controller to obtain a steady-state platoon.

Using the variable gap controller, a control strategy for the highway on-ramp merging can be designed. The control strategy mainly focuses on the behavior of two vehicles; the vehicle that creates a gap in the platoon and the merging vehicle that merges into the platoon. The gap-creating vehicle initially moves using a conventional CACC algorithm. It then switches to the variable gap controller to open a gap in the platoon. Eventually, it transitions to a conventional CACC controller behind the newly merged vehicle. The merging vehicle must align itself with the newly created gap in the platoon. This is done with an individual controller in the sense that the control actions of other vehicles are not directly included in the controller. Of course, some platoon information (like position, velocity, and acceleration) is required to execute the alignment. At a certain point the controller changes to a CACC algorithm to become part of the platoon. One of the main challenges in this maneuver is the spatial constraint proposed by the environment, the maneuver needs to be finished before the end of the on-ramp is reached.

The control of the merging vehicle is validated through experiments with two modified Renault Twizys. One of the vehicles represents the preceding platoon vehicle, it is driven manually and communicates with a merging vehicle. The longitudinal control of the merging vehicle is automated. During the experiments, the merging vehicle aims to establish a two-vehicle platoon using the CACC algorithm before reaching a certain point. However, at the start of the experiment, the inter-vehicle gap is not correct, moreover, the vehicles can drive at different velocities. The proposed control strategy is utilized to obtain a steady-state platoon of two vehicles before the predefined point is reached.

The developed merging control strategy assumes the roles of vehicles during the maneuver are known. To dictate these roles, a merging sequence manager is designed. The merging sequence is the sequence of vehicles after the merging maneuver. This sequence thus dictates which platoon vehicle is required to create a gap and allow the merging vehicle to drive in front of it. Furthermore, the sequence affects the longitudinal trajectories required by the merging, and gap-creating vehicles. The selection of an adequate sequence is thus crucial in the merging process. This topic has been the subject of existing research. This thesis proposes a sequence manager specialized in highway platoons. Specific properties of such platoons are considered when the sequence is determined. Using this knowledge, the disturbances experienced by downstream vehicles may be minimized. The proposed algorithm is compared to multiple benchmark strategies, showing the benefits and challenges of different approaches.

To conclude, this thesis proposes a control strategy for automated merging maneuvers into cooperative platoons. The key vehicles in such a maneuver are the merging, and gap-creating platoon vehicles. Experiments validated the proposed controllers and overarching strategy. Lastly, an algorithm for merging sequence management was developed.

Contents

Nomenclature

Acronyms

ACC Adaptive Cruise Control
ADS Automated Driving System
APF Artificial Potential Field
AV Automated Vehicle
CACC Cooperative Adaptive Cruise Control
CAN Controller Area Network
CAV Connected Automated Vehicle
C-DMAT Cooperative Dual Mode Automated Transport
CZ Cooperation Zone
DDT Dynamic Driving Task
ETA Estimated Time of Arrival
FIFO First-In-First-Out
GCDC Grand Cooperative Driving Challenge
GNSS Global Navigation Satellite System
HMI Human-Machine Interface
i-CAVE Integrated Cooperative Automated VEhicles
IMU Inertial Measurement Unit
MPC Model Predictive Control
NED North, East, Down coordinate system
NWO Netherlands Organisation for Scientific Research
ODD Operational Design Domain
OEM Original Equipment Manufacturer
PD Proportional-Derivative
RMS Root Mean Square
RSU Road Side Unit
SAE Society for Automotive Engineers
V2I Vehicle-to-Infrastructure communication
V2V Vehicle-to-Vehicle communication
V2X Vehicle-to-Everything communication

Chapter 1

Introduction

The interest in automated vehicles has increased in recent years. The potential benefits of automated driving include increased road safety, traffic throughput, and energy-efficient driving. In itself, automated vehicles may be comparable with human-driven vehicles in terms of behavior. However, automated vehicles can be enhanced by using inter-vehicle communication to provide additional environmental information. An additional application obtained by using inter-vehicle communication is cooperative platooning. This technique allows a string of vehicles to maintain small inter-vehicle distances by considering communicated inputs from preceding vehicles. The small inter-vehicle distances improve traffic throughput and potentially energy efficiency, while the controller aims to guarantee road safety. An ideal scenario for platooning is highway driving, due to the long roads with limited curvatures and relatively few entrances and exits. The platoon experiences relatively few external disturbances while driving. Nevertheless, new vehicles may enter the highway and will need to merge with the platoon. An easy solution would be by joining at the end of the platoon. However, if the new vehicle joins at a highway entrance, the length of the on-ramp may restrict this possibility. Due to the spatial constraint, the new vehicle is then forced to join between two platoon vehicles. The short inter-vehicle distances typically do not allow this. This thesis presents a control strategy for the merging of a single automated vehicle into a platoon using inter-vehicle communication. To further introduce and motivate the research in this thesis, this chapter elaborates on automated vehicles and platoons, analysis the current challenges, formulates the research objectives, and outlines the contributions in this thesis.

I

Road safety, traffic congestion, and emissions are amongst the main challenges in current transportation systems. Road safety, traffic throughput, and fuel efficiency can be improved using Cooperative Adaptive Cruise Control. This is a technique in which Connected Automated Vehicles drive closely behind each other using their on-board sensors and Vehicle-to-Vehicle. Driving in such a string is sometimes referred to as platooning. In this thesis, the merging of a new vehicle in such a platoon in a highway on-ramp environment is investigated. Background regarding Automated Vehicles, Connected Automated Vehicles, platooning, and the highway on-ramp environment can be found in Section 1.1. In Section 1.2 some of the existing challenges in this field of research are identified. Based on these challenges, the objectives for this thesis are proposed in Section 1.3. Using these objectives, the section also summarizes the contributions of this thesis. An outline of the thesis is presented in Section 1.4.

1.1 Introduction to Cooperative Automated Merging

This section introduces the concept of cooperative automated merging. This is a topic within the field of automated driving. Its introduction therefore starts with information regarding Automated Vehicles (AVs) in Section 1.1.1. Next, a subclass of AVs, namely Connected Automated Vehicles (CAVs) will be discussed in Section 1.1.2. These are vehicles that use communication to collaboratively perform tasks, such as highway driving or non-signalized intersection crossing. One of the applications is cooperative platooning which will be discussed in Section 1.1.3. Then, the concept of Cooperative Dual Mode Automated Transport (C-DMAT) is discussed in Section 1.1.4. These are systems that have multiple control modes and can be used as AVs or CAVs. An important when researching these systems is the transition between control modes. One of such transitions is present during the cooperative merging maneuver, which is discussed in Section 1.1.5.

1.1.1 Automated Vehicles

Vehicle automation is a compelling field of research and development due to its potential to improve road safety, lower fuel consumption, and increase driver experience. Commercially available applications range from Anti-lock Braking Systems and cruise control, to automatic parallel parking and lane-keeping systems. To aid the development of more advanced systems, a proper taxonomy of these systems is required. Often the taxonomy of the Society for Automotive Engineers (SAE) is used (Society for Automotive Engineers, 2021). This taxonomy defines different levels of Automated Driving Systems (ADSs), from level 0 (*no driving automation*) to level 5 (*full driving automation*). The levels furthermore specify if the system or driver is responsible for the *vehicle motion control* and the *object and event detection and response*. The combination of these two

Table 1.1: SAE levels of driving automation adapted from Society for Automotive Engineers (2021)

Level	Name	DDT		DDT Fallback	ODD
		Sustained lateral and longitudinal vehicle motion control	Object and event detection and response		
Driver Support: Driver performs part or all of the DDT					
0	No driving automation	Driver	Driver	Driver	Not applicable
1	Driver assistance	Driver and System	Driver	Driver	Limited
2	Partial driving automation	System	Driver	Driver	Limited
Automated Driving: ADS performs the entire DDT (while engaged)					
3	Conditional driving automation	System	System	Fallback-ready user	Limited
4	High driving automation	System	System	System	Limited
5	Full driving automation	System	System	System	Unlimited

tasks is referred to as the Dynamic Driving Task (DDT). Furthermore, the levels specify under which conditions the system can be employed. This is called the Operational Design Domain (ODD). The full taxonomy is summarized in Table 1.1.

Many AV manufacturers are conducting experiments for automated driving on public roads (Boggs et al., 2020). While all systems are different, some statements regarding the general design of automated vehicles can be made.

To facilitate automated driving features, AVs require additional sensors and actuators compared to conventional vehicles. The actuators must be such that the DDT can be performed by the vehicle. Due to the level of technology in current production vehicles, the actuation side of the hardware design is relatively advanced. However, for high levels of automation, redundancy in actuators is required. Additionally, advanced sensors are necessary for automated driving. Information from these sensors, such as an Inertial Measurement Unit (IMU) and a high-accuracy Global Navigation Satellite System (GNSS) is fused to accurately

determine the vehicle's position and motion (Hoogeboom, 2020). Furthermore, perception sensors, such as camera, radar, lidar, and ultrasonic sensors, are required (Wang et al., 2020b). Lastly, additional processing power, such as a real-time computer, and a Human-Machine Interface (HMI) are required. An HMI is often composed of multiple objects, such as buttons, touchscreens, and audio devices.

To fully utilize the additional hardware, an appropriate software architecture is required. There are three main modules in the software. The first model abstracts and fuses sensor data to create a world model in which the ego vehicle and environmental features (including possible target vehicles) can be tracked (Hoogeboom, 2020; Serban et al., 2018). Then there is a decision-making module where the different layers of control are performed. This ranges from high-level decisions, such as route planning, to low-level vehicle control. The last module in the software architecture is used to translate the output of the vehicle controller to appropriate commands and transmit these to the actuators. In this thesis, the main interest lies in the decision-making process, since this is where the desired control actions are computed.

Apart from information processing, the architecture should ensure the performance regarding vehicle control. The technique that determines the vehicle actuation based on the processed information is referred to as the decision-making process. The decision-making process is often partitioned into a hierarchical structure (Paden et al., 2016). At the highest level, the *Route Planning* layer selects a route through the road network from the current position to the user-specified destination. Next, the *Behavioral Decision-Making* layer generates a desired vehicle (e.g., cruise in lane, stop at the intersection, or turn right). This behavior is based on the selected route, environment, traffic rules, and other traffic participants. Then, the *Motion Planning* layer uses the desired vehicle behavior to construct a reference path or trajectory. Lastly, a *Vehicle Control* layer translates the path or trajectory into low-level commands. These commands can be followed by the vehicle. An overview is shown in Figure 1.1, which also includes the influence of the vehicle on these processes.

1.1.2 Connected Automated Vehicles

CAVs are a class of AVs that use inter-vehicle communication to increase the available information for each vehicle. Information obtained from other vehicles can include their environment, states, and intention. This additional information is beyond that detectable by on-board sensors and provides a unique insight. To develop CAVs, the architecture of the vehicle is extended with a communication module. This enables the vehicle to communicate with other vehicles, infrastructure, or a combination of the two. This is referred to as Vehicle-to-Vehicle (V2V) communication, Vehicle-to-Infrastructure (V2I) communication, and Vehicle-to-Everything (V2X) communication respectively. The additional

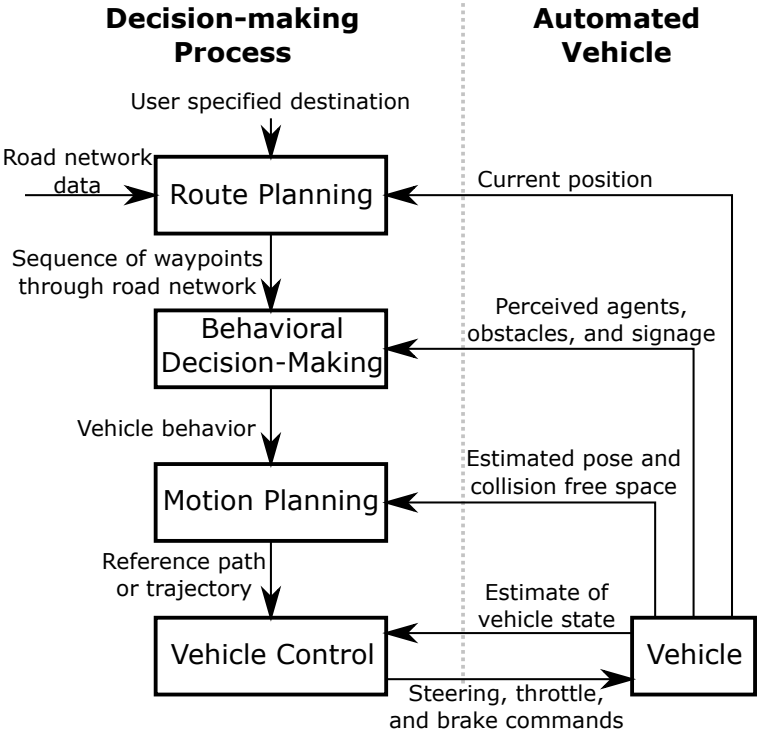


Figure 1.1: An overview of the typical hierarchical structure for decision-making processes (Paden et al., 2016).

information received through communication increases the applications of CAVs over that of conventional automated vehicles. In Wang et al. (2020a) five applications for CAV technology are identified.

The first application is speed harmonization on highways, a detailed review on this topic is provided in Ma et al. (2016). Speed harmonization is a technology that aims at reducing variations in traffic velocity over time and location. In other words, the objective is to reduce oscillations in the traffic velocity. Tools include variable speed limits and ramp metering. This technology has the potential to increase traffic throughput and safety, and reduce the environmental impact of the traffic.

The second application is cooperative driving at signalized intersections. In this application, CAVs are provided with additional information through a Road Side Unit (RSU). A typical scenario for this application is for CAVs to avoid a full stop during the red phase by reducing their velocity and slowly approaching the intersection. Another possible scenario is the coordinated start of multiple CAVs to allow more vehicles to pass through the intersection during the green

phase (Shladover et al., 2015).

The third application is automated coordination at non-signalized intersections. Here inter-vehicle communication is used to control the trajectories of vehicles crossing. The methods often assume a 100% market penetration of highly automated CAVs. Generally, the main aim is to ensure collision-free crossing of the intersection. Furthermore, this approach may increase traffic throughput compared to intersections with controlled traffic lights (Morales Medina et al., 2018).

The fourth application is platooning, this technology aims at controlling CAVs driving in a string. The vehicles utilize communication to maintain small inter-vehicle distances (Ploeg et al., 2011; Semsar-Kazerooni et al., 2016). This can improve road safety and traffic throughput, while reducing energy consumption. Most research aims at the longitudinal control of platoons, but some work extends to the lateral control (e.g., Bayuwindra (2019); Huang et al. (2019); Kianfar et al. (2014); Wei et al. (2019)). Platooning is at the heart of this research and will be further investigated in Section 1.1.3.

The last application is presented in Wang et al. (2020a) is cooperative merging at highway on-ramps. This topic has similar control challenges as the non-signalized intersection problem. An extensive survey regarding both these topics is provided in Rios-Torres and Malikopoulos (2017a). Section 1.1.5 provides a more detailed explanation of this application.

The following two sections will go into detail regarding the last two applications. Namely, platooning and cooperative merging at highway on-ramps. These applications are directly related to the work presented in this thesis and therefore will receive additional attention.

1.1.3 Platooning

Platooning is a concept where vehicles drive closely together in a string with automated longitudinal control. Platooning has the potential to improve traffic throughput, enhance driver comfort and convenience, and increase safety (Rajamani, 2012). The longitudinal controller can be based solely on on-board perception sensors (e.g., radar, lidar, and camera) using technologies such as Adaptive Cruise Control (ACC). One potential danger for platoons is the amplification of perturbations in the upstream direction of the vehicle flow. This is demonstrated with experiments using a string of vehicles with ACC in Milanés et al. (2014). A braking action of 1 $\mathrm{m/s^2}$ of the leader is amplified to 3 $\mathrm{m/s^2}$ in the fourth vehicle. It is likely that if the platoon were longer emergency braking or even a rear-end collision could occur. Therefore, an important consideration in the design of a platooning algorithm is preventing the braking actions from amplifying when they propagate down the string. The attenuation of excitations in the upstream direction of the vehicle flow is known as string-stability (Chu, 1974; Ploeg et al., 2011). There exist some ACC systems that achieve string-stability (Liang and

Peng, 1999). However, this often requires a sufficiently long headway time which is not always implemented for practical reasons. In practice, commercial ACC systems are therefore often not string-stable (Gunter et al., 2021; Li et al., 2021). When considering the traffic flow, lack of string-stability may lead to *phantom traffic jams* (Calvert et al., 2011). Therefore, other solutions for platooning are investigated.

One solution is cooperative platooning, where CAVs use communicated data to maintain string-stability while driving at small inter-vehicle distances. Therefore, this technology has the additional potential to reduce aerodynamic drag (Al Alam et al., 2010; Kim et al., 2021). Due to the combination of small inter-vehicle distances and string stability, cooperative platoons are the standard in platooning technology. Therefore, platooning and cooperative platooning are often used synonymously. From this point on, the term platooning will be used to indicate cooperative platoons. There are multiple control strategies for cooperative platooning, the most common categories are Cooperative Adaptive Cruise Control (CACC) and Artificial Potential Field (APF) approaches.

CACC is an extension of ACC that uses additional communicated information to better follow the preceding vehicle. Commonly, the communicated information includes the control action of the preceding vehicle. Ploeg et al. (2011) proposes a Proportional-Derivative (PD) type controller with an additional feedforward term using the communicated data. In Milanés et al. (2014), the previously mentioned ACC platooning experiments are repeated with a CACC controller. The deceleration of the fourth vehicle is then lower than that of the platoon leader. These results demonstrate the string-stable behavior of the CACC algorithm.

The APF approach is an alternative to the CACC approach (Semsar-Kazerooni et al., 2016). The main advantage of the APF approach is the possibility of gentle gap closing and heavy emergency braking with the same control strategy. In such an approach, the subsequent vehicle experiences a repulsive artificial force when it is too close to its predecessor. An attractive artificial force is experienced when the vehicle is too far from its predecessor. These artificial forces are then translated to longitudinal control inputs for the vehicle. The repulsive and attractive artificial forces can be tuned differently to allow for the desired gap closing and emergency braking behavior. Another advantage of the APF is its suitability for multiple-objectives problems. For example, in Semsar-Kazerooni et al. (2017), an APF approach is used in a highway merging problem. During merging, the gap opening platoon vehicle considers the vehicle on another lane for its control input. However, the repulsive force of the preceding vehicle in the same lane is considered when the inter-vehicle distance gets too small. This prevents head-on collisions in the platoon during the maneuver. The disadvantage of an APF approach is that it may be more difficult to prove string-stability than with a CACC approach. This is due to the nonlinearities that are often in the error definition of the APF algorithm.

I

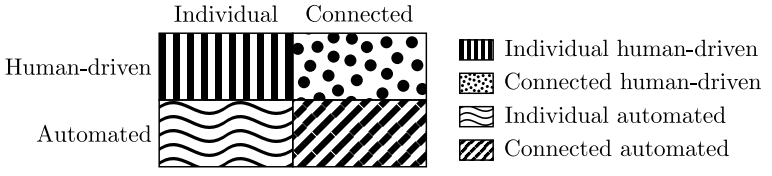


Figure 1.2: An overview of the possible control modes of C-DMAT systems.

Well-known experiments for platooning were conducted during the 2011 Grand Cooperative Driving Challenge (GCDC) (Van Nunen et al., 2012). During the event, nine international teams were challenged to demonstrate their cooperative driving technologies. This is done through two scenarios. First, there is an *urban scenario*, in which a platoon of vehicles must start driving at a traffic light and eventually join another platoon. Then there is a *highway scenario* where a steady-state platoon must be maintained behind an organization lead vehicle. The lead vehicle introduces disturbances and eventually comes to a full stop. These experiments show that string-stability is achievable with an inhomogeneous platoon, meaning the vehicles in the platoon are not identical in important aspects such as the drivetrain (Nieuwenhuijze et al., 2012).

1.1.4 Cooperative Dual Mode Automated Transport Systems

In practice, a vehicle can be categorized differently depending on its environment. For example, a level 4 AV can be a human-driven vehicle when outside of its ODD, or a CAV can operate as an AV when no other CAVs are within communication range. Systems designed specifically to consider the multiple control modes are referred to as C-DMAT systems. An overview of the possible control modes is given in Figure 1.2. The individual human-driven control mode is that of conventional human-driven vehicles. This can be extended by introducing V2X communication to provide the driver with additional information. An example is the eight-generation Volkswagen Golf, where the driver can receive warnings based on other vehicles' experiences. For example, the driver can be warned for a slippery road surface or sudden braking of traffic ahead Abuelsamid (2019). The two automated control modes have been discussed in previous sections. When designing a C-DMAT system, it is important to not just consider these separate modes, but also the transitions between them.

One research program in which a C-DMAT system is researched and developed is the *Integrated Cooperative Automated VEhicles* (i-CAVE) program (I-Cave, 2016; Nijmeijer et al., 2021). To investigate different aspects of C-DMAT systems, this program is divided into seven projects. Six of these projects are research-focused, namely: Sensing and Mapping (Lu and Dubbelman, 2020), Cooperative Vehicle Control (Van Hoek, 2021; Schinkel, 2021), Dynamic Fleet Management (Alves Beirigo, 2021; Los et al., 2020), Communication (Lampel

et al., 2020), Human Factors (Boelhouwer, 2021; Dey, 2020; Walker, 2021), and Architecture Functional Safety (Serban et al., 2018). The seventh project is the development of a full-scale demonstrator platform (Hoogeboom, 2020). The research of the first six projects can be integrated on this platform for experimentation, validation, and demonstration. The work in this thesis is part of the Cooperative Vehicle Control project in the i-CAVE program with project number 14893, which is partly financed by the Netherlands Organisation for Scientific Research (NWO).

1.1.5 Cooperative Merging Maneuver at Highway On-ramps

An example of C-DMAT employment is found in the context of platooning on highways. It is desirable that other CAVs can merge into an existing platoon while driving. The new vehicles then become a CAV, while they are driven individually before the merge. Highway on-ramps are a logical environment to merge and join a platoon. Due to the small inter-vehicle distances in platoons, long platoons may block the highway entrance for a substantial amount of time. It is then not possible for new vehicles to join in front of or behind the existing platoon. Therefore, cooperative merging maneuvers are required for practical utilization of platoons.

During the 2016 GCDC, the merging of cooperative platoons is investigated using a *highway lane-reduction scenario* as described in Ploeg et al. (2018). Like the 2011 GCDC, the 2016 GCDC is a competition between multiple teams from academia and industry. In the *highway lane-reduction scenario*, two platoons that are driving in adjacent lanes must merge into one lane because the other lane is blocked. Since two platoons merge, the vehicles are considered CAVs before and after the merging maneuver. This environment is furthermore differentiated from a typical on-ramp environment by the fact that the initial positions and velocities of the vehicles are relatively close. In an on-ramp environment, the platoon and new vehicle may have vastly different initial conditions (Cao et al., 2015), which creates an additional challenge. Therefore, the new vehicle likely requires higher excitations to ensure its position and velocity are both correct and a steady-state platoon is achieved.

Since the importance of the on-ramp environment has been established, it is important to formulate the details of the environment. An example of a typical environment is shown in Figure 1.3. The figure shows a one-lane highway with an on-ramp. The existing highway lane is referred to as the main lane, a cooperative platoon is driving on this lane. Parallel to the main lane is the acceleration lane, which is connected to the on-ramp. Initially, the on-ramp is occupied by a vehicle that does not incorporate communicated information in its controller. This vehicle is referred to as an individually driven vehicle and aims to enter the main lane via the acceleration lane. It needs to merge into the main lane before the end of the acceleration lane, which is indicated by the merging point. It

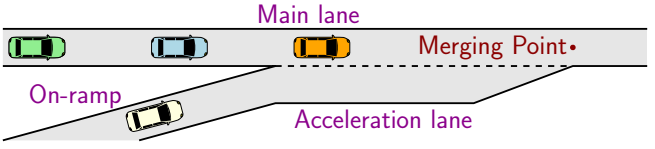


Figure 1.3: A typical on-ramp merging environment with one merging vehicle on the on-ramp and a cooperative platoon on the main lane.

should be noted that some literature defines the environment slightly different. For example, the acceleration lane may be omitted with the on-ramp feeding directly into the main lane. Furthermore, a merging zone may be used instead of a merging point (e.g., Rios-Torres and Malikopoulos (2017b)). These are just minor differences since the approaches often convert the environment to a one-dimensional representation.

In existing research, one of the main variating properties in the environmental description is the considered communication network. Some research assumes V2I communication through an RSU. In that case, often all vehicles within a certain range of the merging point can communicate with each other (e.g., Milanés et al. (2011)). Other research only considers V2V communication to be present (e.g., Raboy et al. (2021)). A description of the established communication network is then more difficult to formulate. The advantage of the latter communication strategy is that no additional infrastructure is required to deploy this technology.

The consideration of V2I or V2V communication is also connected to the control approach. An important split between the control approaches is that of *centralized* and *decentralized* approaches (Rios-Torres and Malikopoulos, 2017a). In *centralized* approaches, at least one task is globally decided for all vehicles by a single controller. Commonly an RSU is used for this controller, but a *master* vehicle may be used. In *decentralized* approaches, each vehicle determines its own control policy for which it can use communicated data from other vehicles. The usage of a V2V based network is more logical for such a scenario because additional infrastructure can be avoided.

As previously mentioned, there are many similarities between cooperative merging and non-signalized intersection control. This is especially true for non-signalized intersection controllers that use the virtual platooning concept (e.g., Morales Medina et al. (2018) and Vaio et al. (2019)). A virtual platoon is a platoon that is formed according to a one-dimensional coordinate system, even when not all vehicles follow the same path in a two-dimensional space. For example, the one-dimensional coordinate system can be defined as the distance to the middle of an intersection or the end of an on-ramp. The virtual platoon ensures there is sufficient inter-vehicle distance when the two-dimensional paths intersect at this point.

Intuitively, an important aspect of the non-signalized intersection problem is the order in which vehicles cross the intersection. This order needs to be established such that collisions can be avoided. In the cooperative merging scenario, this order is just as important. Due to the small inter-vehicle distances in the platoon, some platoon vehicles are required to take action to accommodate the new vehicle. The relevant platoon vehicles are selected by defining a merging sequence, which is the sequence of the vehicles in the platoon after the maneuver. In essence, the sequence dictates the two platoon vehicles between which the new vehicle will join. In literature, multiple approaches for sequence management have been investigated. The merging sequencing approaches can be divided into three categories. First, is a distance-based approach, often referred to as First-In-First-Out (FIFO) (e.g., Rios-Torres and Malikopoulos (2017b)). In essence, when communication is established, vehicles are sequenced based on the distance to the merging point. The distance-based method is easy to implement and gives good results when the initial velocities of all vehicles are similar. However, if there are large variations in the velocity such as in on-ramp environments (Cao et al., 2015), a distance-based approach may be suboptimal. Implicitly, the distance-based approach assumes an equal average velocity from initialization to the merging point. Large differences in initial velocity will thus result in large excitations to align the vehicles. Next, there are time-based approaches, here the Estimated Time of Arrival (ETA) at the merging point is considered rather than the distance (e.g., Wang et al. (2018)). Using the ETA accounts for possible differences in average velocity between the establishment of communication and the merging point. The time-based approach is thus more suitable for an on-ramp environment than the distance-based approach. Lastly, optimization approaches are used (e.g., Jing et al. (2019)). In essence, these approaches consider the predicted trajectories of the vehicles involved to find an optimal merging sequence. They may result in a more desirable sequence than the time-based approaches but are also more computationally expensive.

1.2 Challenges for Cooperative Merging Maneuvers into Platoons at Highway On-ramp Environments

The advantages of cooperative merging into a platoon are evident. For this reason, much research regarding this problem has previously been performed. However, some challenges remain, especially in the context of an on-ramp environment. To understand the open challenges, the on-ramps environment is first investigated. The two main factors of such environments that increase the difficulty of the merging problem are:

1. **A spatial constraint**; in an on-ramp environment, it is important that the maneuver is completed before the new vehicle is forced onto the main lane.

2. **Differences in initial conditions**; typically, vehicles drive slower on the on-ramp than on the highway. This difference in initial velocity also results in a difference in initial relative position and desired terminal relative position. This is because the average velocity of the new vehicle during the maneuver is likely lower than that of the platoon.

To tackle this problem, an appropriate definition of a platoon control strategy is required. There are many approaches for platooning available in literature. Generally, the aim of the platooning strategy is to maintain small inter-vehicle distances, making it difficult for new vehicles to merge into the platoon. Therefore, a strategy to accommodate the merging vehicle is required for the on-ramp merging problem. One solution is the implementation of a gap-opening strategy. The platoon can then prepare for the maneuver before the acceleration lane is reached. Some research has been done regarding gap closing (Van Hoek et al., 2020; Milanés et al., 2014; Semsar-Kazerooni et al., 2016). It is unsure whether these gap-closing algorithms can be adapted easily for the gap-opening problem. Some research does consider gap creation in a platoon (Pueboobpaphan et al., 2010; Uno et al., 1999). However, often these approaches do not ensure that the desired gap is created before a predefined time or location. Ensuring when the maneuver is completed is important because of the spatial constraint of the on-ramp environment. Other research calculates the desired trajectories for main lane vehicles individually during the merge (Ntousakis et al., 2016; Rios-Torres and Malikopoulos, 2017b). Such approaches can ensure sufficient space for merging considering the spatial constraint. However, in the context of platooning, additional switching of control systems is introduced which can lead to undesired behavior. A variable gap platooning controller that avoids some switching may thus be desirable.

Besides the design of a variable gap platooning controller, a suitable design for the vehicle merging control strategy is required. To understand the possible strategies, it is important to distinguish between the main lane platoon vehicles and the new vehicle. One platoon vehicle is required to create a gap in the platoon for which the variable gap platooning controller can be used. There is a vast amount of research regarding the merging of cooperative platoons specifically which can be used as inspiration for the exact implementation (Hult et al., 2018; Pueboobpaphan et al., 2010; Uno et al., 1999). The new vehicle is initially individually controlled because no communication with other vehicles has been established yet. It is then required to align itself with the gap and become part of the platoon. There exists a wide range of literature where optimized trajectories for all vehicles are calculated and executed (Ntousakis et al., 2016; Rios-Torres and Malikopoulos, 2017b; Zhou et al., 2018). Such strategies compute the trajectories of platoon vehicles as well as that of the new vehicle. However, when the problem is investigated as a whole, it is important that the new vehicle transitions from the individual to the cooperative controller. The transition is required to obtain a steady-state platoon and complete the maneuver. Ntousakis

et al. (2016) considers the transition to an ACC controller, but most other work does not specify the transition. Switching to a cooperative controller outside after the maneuver is therefore not specifically considered in the above-mentioned work. The switches may lead to additional transient behavior of the vehicle. A strategy to handle these controller switches can be used to avoid transient behavior to establish a steady-state platoon in a timely fashion.

The controller transition during the merging maneuver is an important challenge. Some literature does not specify the control strategy before and after the maneuver (Rios-Torres and Malikopoulos, 2017b; Zhou et al., 2018). In applications where the controllers outside of the maneuver are specified, they may not be specialized for cooperative platooning. For example, a Model Predictive Control (MPC) Cao et al. (2015) or Ntousakis et al. (2016) ACC algorithm may be applied. In the latter example, an MPC approach is used to determine the vehicle trajectories during the merging maneuver. Simultaneously, an ACC controller is run in the background. Then, the most restrictive control command is applied. The ACC controller directly computes the desired acceleration, but if a controller with an integrator action is used (e.g., Ploeg et al. (2011)) this approach may be more difficult to apply. Other applications use one controller throughout the maneuver but switch the error definition linearly over position or time (e.g., Hult et al. (2018); Milanés et al. (2011)). An analysis regarding the convergence to a steady-state platoon, which is required due to the spatial constraint, is often not included for this approach. Vaio et al. (2019) used an APF platooning approach for which the convergence of the average velocity of all vehicles to a desired velocity can be bound by a tuning parameter. The reasoning for this is that if the average velocity has converged, the position error has also converged. The bound on the average velocity is exponential with respect to time. Therefore, it may not be desirable for on-ramp environments in which large initial differences may be present. The tuning parameter to resolve these large differences may result in excessively large initial excitations which become smaller towards the end of the maneuver. A controller transition method that more evenly distributes these excitations is thus desired for the on-ramp environment.

In literature, multiple approaches for platoon sequencing are proposed, these approaches have varying levels of complexity. As previously discussed, merging sequence strategies include distance-based, time-based, and optimal control-based strategies. Here optimal control-based strategies are the most advanced but also the most computationally expensive. To the author's knowledge, the previously proposed optimal merging sequence managers do not use platoon-specific knowledge. In essence, the approaches regard all vehicles within a certain range (e.g., cooperation areas). However, in the case of platooning it may be possible to consider the impact of the maneuver on subsequent platoon vehicles outside of this range. Current research does not investigate this possibility. The alternative time-based strategies often use simplified vehicle models. For

example, the approach of Wang et al. (2018) allows for instantaneous changes in acceleration in the planned trajectory. Time-based methods that more accurately consider the expected or desired trajectories of the vehicles pose a possible alternative to the optimal control method.

1.3 Research Objectives and Contributions

Based on the aforementioned challenges, this thesis aims to contribute to the available control strategies for the automated merging into a cooperative platoon. Specifically, a highway on-ramp environment is considered which introduces additional spatial constraints to this problem. Moreover, the transition from an individually driven to a cooperatively driven automated vehicle is investigated. To achieve this aim, the following research objectives are defined:

- Design of a control strategy for the cooperative merging of a new vehicle between two platoon vehicles in highway on-ramp environments. This design includes the required inter-vehicle information flow and longitudinal controllers for the involved vehicles. To ensure a steady-state platoon is obtained at the end of the maneuver, the transition to a CACC controller required by some vehicles is given special attention.

- Design of a sequence manager that decides on the desired position of a new vehicle in a platoon. This manager can handle large differences between the initial states of the platoon and new vehicles.

- Experimental testing of the controller transition in spatially constrained environments with full-scale vehicles.

This thesis addresses the objectives through the following contributions:

1. **Variable Gap CACC Controller Design**

   The first contribution is the design of a CACC controller with a variable inter-vehicle distance. The design is based on a PD CACC controller with an additional gap term in the desired inter-vehicle distance. The value of this gap term can be changed to accommodate a new vehicle in the platoon. When the additional term remains constant the benefits of the PD CACC algorithm are obtained, such as string-stability. Feed-forward terms in the control law are based on the derivatives of the gap term. These feed-forward terms ensure that the desired trajectory is followed. Furthermore, this controller can also be used to close a residual gap during the transition to the desired CACC controller in the last phase of the merging maneuver. The controller design and its demonstration with a test setup can be found in Chapter 3. This chapter is based on previous research published in Scholte et al. (2020).

2. **Control Strategy Design for the Merging Maneuver at Highway On-ramps**

   The second contribution considers the control strategy for the entire merging maneuver. This is everything from the moment that the new vehicle knows where to join to the completion of the new steady-state platoon. The main vehicles of interest are the gap opening platoon vehicle and the new vehicle. Both vehicles are required to perform a controller transition. The gap opening vehicle transitions between two CACC controllers to select a new predecessor. The new vehicle transitions from an individual controller to the CACC controller. The designed strategy is provided in Chapter 4 which also demonstrates its behavior using simulations. This chapter is based on previous research published in Scholte et al. (2022).

3. **Design of a Merging Sequence Manager for the Merging into Platoons**

   An important aspect of the merging maneuver is the merging sequence manager. The merging sequence dictates the two platoon vehicles between which the merging vehicle should enter. The position of the merge affects the excitations of not just the new and gap-opening vehicle, but also all subsequent vehicles in the platoon. Therefore, the third contribution is a merging sequence manager which aims to limit the longitudinal excitations of the subsequent platoon vehicles, even if they do not have communication established with the new vehicle. In Chapter 5, the proposed design is presented and analyzed using multiple benchmark methods. This results in recommendations regarding the current implementation of a merging sequence manager and the direction of future research.

4. **Experimental testing of the Longitudinal Control Strategy of the New Vehicle**

   The last contribution is the experimental testing of the longitudinal control strategy of the new vehicle using a full-scale setup. The control strategy exists of an individual controller, a controller transition, and a CACC controller. Therefore, it has all the components of the gap opening control strategy and more. The successful experiments show the potential of the proposed merging maneuver control strategy. Furthermore, spatial constraints are considered in the experiments to simulate the finite length of an on-ramp. Chapter 6 presents the experiments and their results.

1.4 Outline

Additional background, literature, and preliminaries regarding some important topics discussed in this introduction can be found in Chapter 2. Then, the proposed design of the cooperative on-ramp merging control strategy will be

discussed in subsequent chapters. The different aspects are discussed from the lower to the higher control levels of the hierarchy shown in Figure 1.1. This reversed order is used because lower-level controllers are required for the analysis of higher-level controls. However, the analysis of lower-level controllers can be performed using specific scenarios such that the required outcome of higher-level controllers can be determined manually. The proposed variable gap CACC algorithm will thus first be discussed in Chapter 3. This chapter includes an analysis and experiments with small mobile robots. Next, Chapter 4 discusses the proposed strategy for cooperative highway on-ramp merging when the desired merging sequence is known. This requires the *Behavioral Decision-Making* layer to determine when vehicles should change their control strategy, the *Motion Planning* layer to ensure the vehicles are spaced correctly, and the *Vehicle Control* layer to execute the trajectory accordingly. Simulations are used to validate the proposed control strategy. Then, Chapter 5 discusses the merging sequence management. This is part of the *Behavioral Decision-Making* layer, lower-level control layers of the previously proposed solution are used to demonstrate the effectiveness of this manager through simulations. The applicability of the controllers is then shown in Chapter 6 where experiments on a full-scale demonstrator platform are analyzed. Lastly, Chapter 7 gives conclusions on the presented work and suggests future research directions.

Chapter 2

Background and Preliminaries

This thesis proposes a control algorithm for cooperative merging maneuvers into platoons. Two important concepts used throughout this thesis are platooning and cooperative merging maneuvers. This section presents some preliminaries regarding these topics intended to help introduce the work presented in this thesis. An overview of the available techniques for platooning is provided. Moreover, the specific technique that is used as a basis for the work presented in this thesis is explained in more detail. This is followed by a definition of string stability and a criterion of string stability used for platooning. Next, the topic of cooperative merging maneuvers is covered. First, an overview of existing research is provided. This overview introduces important design possibilities such as the network structure and road geometry. In addition, the motivation for the decisions on these topics made in this thesis is provided.

This chapter introduces the two main concepts discussed in this thesis. First, the concept of platooning is introduced, with a literature review, the definition of one controller technique, and a definition of string stability. Secondly, the problem of automated merging maneuvers is explained. An overview of the literature concerning this problem is given, followed by a clarification of the general design choices made in this project.

2

2.1 Platooning

Adequate longitudinal vehicle following is essential for the development of Automated Vehicles (AVs). This is recognized in Chandler et al. (1958), where vehicle following behavior is modeled and the propagation of perturbations down a string of vehicles is analyzed. When perturbations are amplified as they propagate down the string, *phantom traffic jams* can occur (Calvert et al., 2011). This is a phenomenon where a braking action of a vehicle in a sequence of vehicles amplifies as it moves down the string such that subsequent vehicles experience severe braking actions and congestion occurs. Therefore, it is important that perturbations are not amplified, which is indicated with the term *string stability* (Chu, 1974; Ploeg et al., 2011). To achieve *string stability*, the usage of inter-vehicle communication is beneficial as it can provide additional information to the following vehicle (Gunter et al., 2021; Milanés et al., 2014). AVs with wireless communication are referred to as Connected Automated Vehicles (CAVs).

Vehicles driving closely behind each other using inter-vehicle communication and longitudinal automation are referred to as a cooperative platoon. The potential benefits of this technique include increased traffic throughput (Van Arem et al., 2006), safety (Alam et al., 2015), and efficiency (Kim et al., 2021). There are several control strategies available for cooperative platooning. This section will briefly go over the available techniques, then one of the control approaches is highlighted and covered in detail. This approach will be the basis of the controller proposed in this thesis. Finally, the notion of string stability is briefly defined.

2.1.1 Literature Overview

There are multiple approaches for platooning, an overview of the coordinates often used in these approaches is given in Figure 2.1. The vehicles are driving in a string where the index of a subsequent vehicle is one higher than that of its predecessor. This environment enables their position to be expressed in a one-dimensional coordinate system which is denoted as q in the figure. In the figure, position q_i is at the rear bumper of vehicle i. Therefore, the distance between position q_i and q_{i-1} is the sum of vehicle length L_i and inter-vehicle distance d_i. Sometimes, the position q_i is placed in the center of gravity or the front bumper. Then, L_{i-1} is required to express the d_i in terms of q_i and q_{i-1}. Therefore,

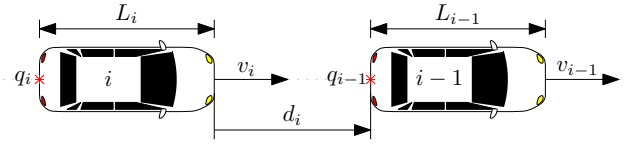


Figure 2.1: The general coordinate system used in platooning.

the preferred method for lower-level control strategies is placing q_i on the rear bumper. The velocity v of the vehicles is expressed along the one-dimensional coordinate q. For different strategies, the available and required information may differ. Generally, it is assumed that the vehicles can measure their own position, velocity, and acceleration. Furthermore, vehicle i can measure the inter-vehicle distance d_i and relative velocity $v_{i-1} - v_i$ using perception sensors such as radar. Any missing information can be transmitted using wireless communication which introduces a time-delay in the system.

One approach to solving the platooning problem is by using a linear feedback controller as proposed in VanderWerf et al. (2001). The proposed Cooperative Adaptive Cruise Control (CACC) controller computes a commanded acceleration, based on the acceleration of the preceding vehicle, the relative velocity between the two vehicles, and the position error based on a desired distance between vehicles. This desired distance is the maximum among the safe following distance, a constant time following distance, and a minimum allowed distance. The user can tune the reaction of the controller on the relative velocity and position error. Simulations demonstrate the increased traffic throughput using the proposed algorithm compared to manually driven or Adaptive Cruise Control (ACC) vehicles, assuming 100 percent market penetration. Van Arem et al. (2006) investigates the performance of this controller in a highway lane drop environment from four to three lanes. The results indicate an increase in traffic stability and throughput even with partial market penetration rates.

Another linear feedback control solution for the platooning problem is the CACC algorithm presented in Ploeg et al. (2011). This algorithm defines a desired inter-vehicle distance d_r that consists of a headway time and a standstill distance. Then a Proportional-Derivative (PD) controller is used to ensure the desired inter-vehicle distance is maintained. The control input of the vehicles is a desired acceleration that their predecessor broadcasts. The desired acceleration of the preceding vehicle is used in the control algorithm to better react to perturbations. For homogeneous platoons, this algorithm can achieve string stability despite communication delays by properly choosing the PD gains.

The CACC controller is extended in Lefeber et al. (2020). Instead of broadcasting the desired acceleration, each vehicle broadcasts its current acceleration. Therefore, knowledge of a preceding vehicle's driveline dynamics is no longer necessary, and the controller is therefore applicable to heterogeneous platoons

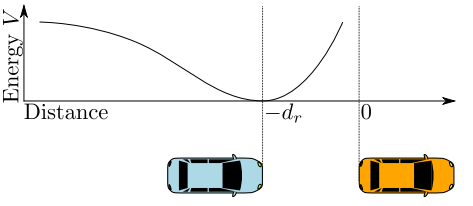


Figure 2.2: The virtual energy as a function of inter-vehicle distance for an APF in platooning, based on Semsar-Kazerooni et al. (2016).

for suitably chosen gains. The resulting platoon behavior is string stable. Using the correct tuning, this controller can mimic the behavior of the previous CACC controller.

The effectiveness and applicability of the linear feedback controller for the platooning problem are demonstrated in multiple experiments. Nieuwenhuijze et al. (2012) describes experiments using a heterogeneous platoon where a heavy-duty truck is following a much lighter passenger vehicle. For a large enough headway time, string stability can be achieved. However, the limited performance of the truck influences the string stability. In Milanés et al. (2014), the behavior of this type of controller during real traffic scenarios is investigated. These scenarios include changing the desired headway time, and cut-in and cut-out maneuvers. During these experiments, the vehicles and underlying control algorithms perform well.

A disadvantage of a linear feedback controller is that vehicle may close the gap too aggressively because no distinction is made between the inter-vehicle distance being too large and it being too small. However, when the distance is too small aggressive deceleration may be required to avoid a collision whereas the vehicle may behave more comfortably when the distance is too large. One solution that takes this difference into account is a controller based on Artificial Potential Fields (APFs) (Semsar-Kazerooni et al., 2016). This technique creates an artificial force that either attracts or repulses the subsequent vehicle. These forces can be tuned independently. Therefore, heavy braking can be used for collision avoidance when the subsequent vehicle is too close. Meanwhile, the acceleration during gap closing when the vehicle is too far remains reasonable. In essence, a virtual energy function V is defined as a function of inter-vehicle distance which is zero at the desired inter-vehicle distance d_r. The partial derivative of this energy to inter-vehicle distance is used in the control law which aims to minimize the energy. The energy function is roughly shaped as shown in Figure 2.2. Therefore, the resulting control law can perform emergency braking or gradual gap-closing depending on the inter-vehicle distance.

Another solution to the platooning problem that allows for more freedom in designing the desired behavior is using a Model Predictive Control (MPC) ap-

proach as described in Kianfar et al. (2015). This approach allows for constraints on the position and velocity errors as well as on states such as acceleration and velocity. These bounds can be chosen by the user, which helps to achieve safe and comfortable vehicle behavior. Moreover, the MPC controller is combined with a frequency domain controller to ensure string stability.

In Chen et al. (2018), a min-max MPC approach is proposed that takes feedback delays and parametric uncertainties into account. The proposed algorithm is demonstrated using simulations with a heterogeneous platoon. In all simulations, the proposed min-max MPC controller outperforms a nominal MPC controller. For obvious reasons, the robustness and applicability improve the applicability of this controller in real-world applications.

Platooning can also be performed by running a trajectory planning algorithm that considers the trajectory of the preceding vehicle. As shown in Van Hoek et al. (2021), this technology can merge the framework of AVs and CAVs to potentially achieve a string stable platoon. In the algorithm, vehicles are required to communicate their expected trajectories. Often, such solutions can cause problems due to the amount of data required to communicate a trajectory and the limited amount of bandwidth available. However, in the proposed algorithm the trajectories are designed using B-splines. Therefore, they can be expressed using only a few parameters, and communicated with limited bandwidth.

In the above examples of platooning algorithms, vehicles consider communicated information from their predecessor. However, Wilmink et al. (2007) proposes a linear feedback algorithm where information from multiple preceding platoon vehicles is used. Depending on the settings, the proposed controller can maintain the distance to the predecessors safer, more comfortable, or with higher throughput. In Zegers et al. (2016), a bi-directional predecessor-follower network topology is proposed. Experiments using a three-vehicle platoon are used to demonstrate the feasibility of this design. Similarly, Di Bernardo et al. (2015) proposes a controller design for switching and fixed network topologies that consider vehicles further in the platoon. The network topology may be switching due to vehicles joining or leaving the platoon, or due to the loss of communication links. The controller is demonstrated using simulations and experiments with a three-vehicle platoon. The behavior under different network topologies is still an open topic in research, especially regarding performance metrics such as string stability. Therefore, a traditional unidirectional predecessor-follower network topology is assumed for the remainder of this thesis.

In this thesis, the linear feedback CACC algorithm of Ploeg et al. (2011) is used as a basis. In Chapter 3 the definition of the desired inter-vehicle distance is altered to allow for gradual gap-closing and gap-opening. This lowers the need for an MPC or APF approach and the properties of the PD controller make the effect of changing the desired inter-vehicle distance while driving easily understandable. However, the proposed method can likely be extended to the other approaches because they also use a desired inter-vehicle distance.

2.1.2 Conventional CACC Controller

This section discusses the CACC control strategy of Ploeg et al. (2011). This controller is used as the basis for the control strategies presented in this thesis and hence will be referred to as the conventional CACC controller from here on. The longitudinal vehicle model used for this controller is defined as

$$\dot{q}(t) = v(t) \tag{2.1}$$
$$\dot{v}(t) = a(t) \tag{2.2}$$
$$\dot{a}(t) = \frac{1}{\tau}u(t) - \frac{1}{\tau}a(t). \tag{2.3}$$

Where q, v, a, and u denote the one-dimensional position, velocity, acceleration, and desired acceleration respectively. Furthermore, τ is a time constant that represents the driveline dynamics. It is assumed the value of τ is equal for all vehicles in the platoon, this is often referred to as a homogeneous platoon. The desired acceleration u is a control input determined by the controller.

The inter-vehicle distance is defined as the distance from the front bumper of vehicle i to the rear bumper of its preceding vehicle $i-1$. Positions q_i and q_{i-1} are defined as the position of the rear bumper. The inter-vehicle distance and its derivative can thus be denoted as

$$d_i(t) = q_{i-1}(t) - q_i(t) - L_i \tag{2.4}$$
$$\dot{d}_i(t) = v_{i-1}(t) - v_i(t). \tag{2.5}$$

Here L_i denotes the length of vehicle i. Index $i \in \mathcal{P}_m$ denotes the index of the controlled vehicle in the set of m subsequent platoon vehicles $\mathcal{P}_m = \{1, \ldots, m\}$.

The CACC controller aims to maintain a desired inter-vehicle distance $d_{r,i}$ defined as

$$d_{r,i}(t) = hv_i(t) + r. \tag{2.6}$$

Where constants h and r denote the headway time and standstill distance. The desired inter-vehicle distance thus increases with the velocity. From this point onward, time argument t will be omitted for readability.

Now, the position error is defined as the difference between the desired and actual inter-vehicle distance. Using the vehicle dynamics, the error dynamics can be defined as

$$e_i = d_i - d_{r,i} = q_{i-1} - q_i - L_i - hv_i - r \tag{2.7}$$
$$\dot{e}_i = v_{i-1} - v_i - ha_i \tag{2.8}$$
$$\ddot{e}_i = a_{i-1} - a_i\left(1 - \frac{h}{\tau}\right) - \frac{h}{\tau}u_i. \tag{2.9}$$

The error states are defined as $e_{1,i} := e_i$, $e_{2,i} := \dot{e}_i$, $e_{3,i} := \ddot{e}_i$. Assuming the driveline constant τ is equal for vehicles i and $i-1$, the derivative of $e_{3,i}$ yields

$$\dot{e}_{3,i} = -\frac{1}{\tau}e_{3,i} - \frac{1}{\tau}\xi_i + \frac{1}{\tau}u_{i-1}, \tag{2.10}$$

with

$$\xi_i = h\dot{u}_i + u_i. \tag{2.11}$$

Based on (2.10) a function of ξ is designed that controls the error dynamics and compensates for u_{i-1}, such that

$$\xi_i = k_{\mathrm{p}} e_{1,i} + k_{\mathrm{d}} e_{2,i} + u_{i-1}. \tag{2.12}$$

Here scalars k_{p} and k_{d} are control parameters. Now (2.11) and (2.12) yield the control law

$$\dot{u}_i = \frac{1}{h}\left(k_{\mathrm{p}} e_{1,i} + k_{\mathrm{d}} e_{2,i} + u_{i-1} - u_i\right). \tag{2.13}$$

The stability of the individual vehicle's error dynamics is investigated. These dynamics are investigated by writing them in the form

$$\begin{bmatrix} \dot{e}_{1,i} \\ \dot{e}_{2,i} \\ \dot{e}_{3,i} \\ \dot{u}_i \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{-k_p}{\tau} & \frac{-k_d}{\tau} & -\frac{1}{\tau} & 0 \\ \frac{k_p}{h} & \frac{k_d}{h} & 0 & -\frac{1}{h} \end{bmatrix} \begin{bmatrix} e_{1,i} \\ e_{2,i} \\ e_{3,i} \\ u_i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{h} \end{bmatrix} u_{i-1}. \tag{2.14}$$

This system has an equilibrium in the origin for $u_{i-1} = 0$. The Routh-Hurwitz stability criterion can be applied to the state matrix. It follows that the error dynamics are stabilized for $h > 0$ and any $k_p > 0$ and $k_d > 0$ that satisfy $k_d > k_p \tau$.

Using the vehicle model of (2.1)-(2.3), the spacing error (2.7), and control law (2.13), the following homogeneous platoon model is obtained

$$\begin{bmatrix} \dot{e}_i \\ \dot{v}_i \\ \dot{a}_i \\ \dot{u}_i \end{bmatrix} = \begin{bmatrix} 0 & -1 & -h & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} \\ \frac{k_p}{h} & \frac{k_d}{h} & -k_d & -\frac{1}{h} \end{bmatrix} \begin{bmatrix} e_i \\ v_i \\ a_i \\ u_i \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{k_d}{h} & 0 & \frac{1}{h} \end{bmatrix} \begin{bmatrix} e_{i-1} \\ v_{i-1} \\ a_{i-1} \\ u_{i-1} \end{bmatrix}, \quad i \in \mathcal{P}_m \tag{2.15}$$

or, in short,

$$\dot{x}_i = A_0 x_i + A_1 x_{i-1}, \quad i \in \mathcal{P}_m \tag{2.16}$$

with state vector $x_i = \begin{bmatrix} e_i & v_i & a_i & u_i \end{bmatrix}^\top$, and the matrices A_0 and A_1 are defined according to (2.15).

Based on the vehicle model of (2.1)-(2.3), and input dynamics (2.11), the dynamics of the platoon leader $i = 0$ can be formulated as

$$\begin{bmatrix} \dot{e}_0 \\ \dot{v}_0 \\ \dot{a}_0 \\ \dot{u}_0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} \\ 0 & 0 & 0 & -\frac{1}{h} \end{bmatrix} \begin{bmatrix} e_i \\ v_i \\ a_i \\ u_i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{h} \end{bmatrix} \xi_0 \tag{2.17}$$

or, in short,

$$\dot{x}_0 = A_r x_0 + B_r u_r \tag{2.18}$$

with state vector $x_0 = \begin{bmatrix} e_0 & v_0 & a_0 & u_0 \end{bmatrix}^\top$, input $u_r = \xi_0$ being the external input to the platoon, and the matrices A_r and B_r are defined according to (2.17). Note that e_0 is a dummy state for which holds that $e_0(t) = e_0(0)$. This dummy state has no further influence since the first columns of A_1 and A_r are equal to the zero column. In the remainder of this thesis, $e_0(0) = 0$ is chosen. For $u_r = 0$, the equilibrium state of (2.17) is equal to $\bar{x}_0 = \begin{bmatrix} 0 & \bar{v}_0 & 0 & 0 \end{bmatrix}^\top$. This equilibrium is only marginally stable because (2.17) is an uncontrolled vehicle model. The extension to the controlled vehicle model is deemed unnecessary at the moment.

Using (2.16) and (2.18) a state-space model describing the entire platoon can now be formulated as

$$\begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \vdots \\ \dot{x}_m \end{bmatrix} = \begin{bmatrix} A_r & & & \mathrm{O} \\ A_1 & A_0 & & \\ & \ddots & \ddots & \\ \mathrm{O} & & A_1 & A_0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_m \end{bmatrix} + \begin{bmatrix} B_r \\ \mathrm{O} \\ \vdots \\ \mathrm{O} \end{bmatrix} u_r \tag{2.19}$$

or, in short,

$$\dot{x} = Ax + Bu_r \tag{2.20}$$

with state vector $x = \begin{bmatrix} x_0^\top & x_1^\top & \dots & x_m^\top \end{bmatrix}^\top$, and the matrices A and B are defined according to (2.19).

2.1.3 String Stability

As previously mentioned, an important aspect of platoons is string stability. The notion of string stability is formalized in this section. In Ploeg (2014) a definition of $\mathcal{L}_p$ string stability is provided, which is as follows. Given a heterogeneous interconnected system

$$\dot{x}_0 = f_r(x_0, u_r) \tag{2.21}$$

$$\dot{x}_i = f_i(x_i, x_{i-1}), \qquad i \in \mathcal{P}_m \tag{2.22}$$

$$y_i = h(x_i), \qquad i \in \mathcal{P}_m. \tag{2.23}$$

Here $u_r \in \mathbb{R}^q$ denotes the external input, $x_i \in \mathbb{R}^n$, $i \in \mathcal{P}_m \cup \{0\}$ denotes the state vector, and $y_i \in \mathbb{R}^l$, $i \in \mathcal{P}_m$ denotes the output vector. Moreover, $\mathcal{P}_m = \{1, \dots, m\}$ is the set of m subsequent platoon vehicles, and $f_r : \mathbb{R}^n \times \mathbb{R}^q \mapsto \mathbb{R}^n$, $f_i : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}^n$, $i \in \mathcal{P}_m$, and $h : \mathbb{R}^n \mapsto \mathbb{R}^l$. In the scope of platooning the state is typically defined as $x_i = \begin{bmatrix} e_i, & v_i & a_i & \dots \end{bmatrix}^\top$, $i \in \mathcal{P}_m \cup \{0\}$ with a possible extension for additional states such as controller or spacing policy dynamics. It can be noted that external input u_r only influences states x_0 of the platoon leader directly and the states x_i, $i \in \mathcal{P}_m$ of subsequent vehicles are influenced by the states of their predecessor. This observation indeed matches with the control law (2.13).

Let $x = \begin{bmatrix} x_0^\top & x_1^\top & \dots & x_m^\top \end{bmatrix}^\top$ be the lumped state vector with equilibrium solution $\bar{x} = \begin{bmatrix} \bar{x}_0^\top & \bar{x}_0^\top & \dots & \bar{x}_0^\top \end{bmatrix}^\top$ for $u_r = 0$. The system is then $\mathcal{L}_p$ string stable if there exist class $\mathcal{K}$ functions α and β (i.e., functions that are continuous, strictly increasing and $\alpha(0) = 0$, $\beta(0) = 0$), such that, for any initial state $x(0) \in \mathbb{R}^{(m+1)n}$, any $u_r \in \mathcal{L}_p^q$, and $t \in [0, \infty)$,

$$\|y_i(t) - h(\bar{x}_0)\|_{\mathcal{L}_p} \leq \alpha\left(\|u_r(t)\|_{\mathcal{L}_p}\right) + \beta\left(\|x(0) - \bar{x}\|\right), \ \forall\, i \in \mathcal{P}_m. \tag{2.24}$$

If, in addition, with $x(0) = \bar{x}$ it also holds that

$$\|y_i(t) - h(\bar{x}_0)\|_{\mathcal{L}_p} \leq \|y_{i-1}(t) - h(\bar{x}_0)\|_{\mathcal{L}_p}, \ \forall\, i \in \mathcal{P}_m \backslash \{1\}, \tag{2.25}$$

the system is strictly $\mathcal{L}_p$ string stable with respect to its input $u_r(t)$. Here, $\|\cdot\|$ denotes any vector norm, and $\|\cdot\|_{\mathcal{L}_p}$ denotes the signal p-norm, which for signal $y_i(t)$ is defined as Desoer and Vidyasagar (2009)

$$\|y_i(t)\|_{\mathcal{L}_p} = \sqrt[p]{\int_0^\infty \left(\|y_i(t)\|_p^p\right) dt}. \tag{2.26}$$

$\mathcal{L}_p^q$ denotes the q-dimensional space of vector signals that are bounded in the $\mathcal{L}_p$ sense (i.e., have a bounded p-norm). In (2.24) the term $\alpha\left(\|u_r(t)\|_{\mathcal{L}_p}\right)$ accounts for input $u_r(t)$ and the term $\beta\left(\|x(0) - \bar{x}\|\right)$ accounts for perturbations in the initial condition.

The definition of string stability given above is relatively general as it applies to all systems of that can be expressed as (2.21)-(2.23). The platoon dynamics in (2.19) qualifies as such a system independent of the chosen matrices A_0, A_1, A_r, and B_r. For this special case of systems a more specific string stability criterion can be formulated (Ploeg, 2014). This criterion is valid for the presented conventional CACC controller, but also for a wide range of other platooning algorithms.

First, a linear output function is defined as

$$y_i = Cx_i = C_i x, \quad i \in \mathcal{P}_m \tag{2.27}$$

with output matrix C and $C_i = \begin{bmatrix} \mathrm{O}_{l \times n(i-1)} & C & \mathrm{O}_{l \times n(m-i)} \end{bmatrix}$. Using a coordinate transformation the equilibrium state is chosen as $\bar{x} = \begin{bmatrix} \bar{x}_0^\top & \dots & \bar{x}_0^\top \end{bmatrix}^\top = 0$, hence $h(\bar{x}_0) = C_i \bar{x}_0 = 0 \ \forall \ i \in \mathcal{P}_m$. Then the model (2.20), (2.27) can be formulated in the Laplace domain as

$$y_i(s) = P_i(s) u_r(s) + O_i(s) x(0), \quad i \in \mathcal{P}_m \tag{2.28}$$

where, with a slight abuse of mathematical notation, $y_i(s)$ and $u_r(s)$ with $s \in \mathbb{C}$ denote the Laplace transforms of $y_i(t)$ and $u_r(t)$ respectively. Furthermore, $P_i(s) = C_i \left(sI - A\right)^{-1} B$, $i \in \mathcal{P}_m$ and $O_i(s) = C_i \left(sI - A\right)^{-1}$, $i \in \mathcal{P}_m$ are the

complementary sensitivity transfer function and the initial condition transfer function respectively.

To investigate string stability the relation between $y_i(s)$ and $y_{i-1}(s)$ is investigated. To this end, a string stability complementary sensitivity function $\Gamma_i(s)$ is used, such that

$$y_i(s) = \Gamma_i(s) y_{i-1}(s). \tag{2.29}$$

From (2.28) it follows that, with $x(0) = 0$,

$$\Gamma_i(s) := P_i(s) P_{i-1}^{-1}(s), \tag{2.30}$$

assuming $P_i(s)$ is nonsingular for all i, thus guaranteeing the existence of $P_{i-1}^{-1}(s)$. In practice, $\Gamma_i(s)$ can be described as the transfer function from "input velocity" v_{i-1} to "output velocity" v_i, or equivalently, from "input acceleration" a_{i-1} to "output acceleration" a_i (Lefeber et al., 2020). Now the criterion for (strict) $\mathcal{L}_2$ string stability in Ploeg (2014) is stated as follows:

Let (2.20), (2.27) represent a linear unidirectional-interconnected system of which the input-output behavior is described by (2.28). Assume that the pair (C_i, A) is such that unstable and marginally stable modes are unobservable and that $P_i(s)$ is square and nonsingular for all $i \in \mathbb{N}$. Then the system (2.20), (2.27) is $\mathcal{L}_2$ string stable if:

1. $\|P_1(s)\|_{\mathcal{H}_\infty} < \infty$ and
2. $\|\Gamma_i(s)\|_{\mathcal{H}_\infty} \leq 1, \ \forall \ i \in \mathbb{N} \backslash \{1\}$

with $\Gamma_i(s)$ as in (2.30). Moreover, the system is $\mathcal{L}_2$ string stable if and only if conditions 1 and 2 hold.

Here $\|\Gamma_i(s)\|_{\mathcal{H}_\infty}$ denotes the $\mathcal{H}_\infty$ norm of $\Gamma_i(s)$ which can be defined as

$$\|\Gamma_i(s)\|_{\mathcal{H}_\infty} := \sup_{y_{i-1} \neq 0} \frac{\|y_i(t)\|_{\mathcal{L}_2}}{\|y_{i-1}(t)\|_{\mathcal{L}_2}}. \tag{2.31}$$

Similarly, $\|P_1(s)\|_{\mathcal{H}_\infty}$ is defined as the supremum of the $\mathcal{L}_2$ norm of $y_i(t)$ over the $\mathcal{L}_2$ norm of $u_r(t)$ for $u_r \neq 0$.

To analyze string stability for practical implementation delays in communication have to be taken into account. The communication delay is often expressed as a time delay on the input value u_{i-1} which has to be transmitted. The individual stability of the system (2.14) is around its equilibrium $u_{i-1} = 0$ and therefore not affected by the communication delay. However, for string stability u_{i-1} is not constant and therefore the role of the time delay becomes important. It is possible to include a time delay in $P_i(s)$ in (2.28). In the frequency domain, the transfer function $D(s) = e^{-\theta s}$ represents a time delay of θ seconds. In Ploeg (2014) a more detailed model including time delay is made and analyzed for string stability. It is found that the time delay dictates the minimum value of headway time h for which the system is string stable. For further reading on the topic of time delay and its effect on string stability, the reader is directed to Ploeg (2014).

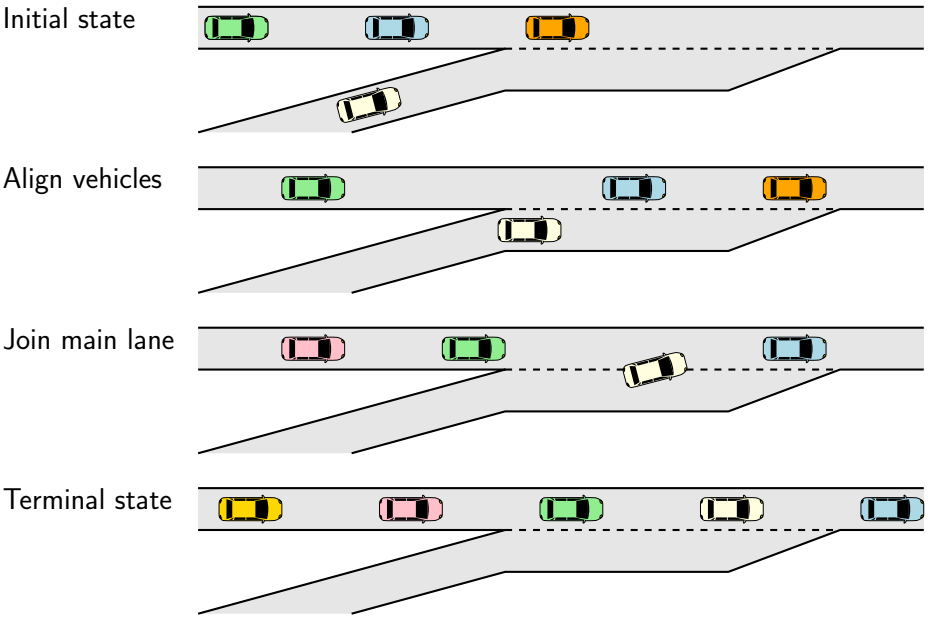


Figure 2.3: A visual summary of the merging maneuver.

2.2 Automated Merging Maneuvers

Automated merging maneuvers are an important research topic in the field of automated driving. When an AV enters a highway, it needs to interact with the existing traffic to perform a safe and comfortable merging maneuver. The difficult environment of an on-ramp provides little information to the vehicle. The usage of inter-vehicle communication can enhance available information for the new vehicle and thus its performance. Such automated merging maneuvers are also applicable to platooning situations and can assist in letting new vehicles join the platoon.

To iterate, the objective of this thesis is to design a control system for the merging of a single vehicle into a densely packed platoon at highway on-ramps. The general process is shown in Figure 2.3. Essentially, the maneuver starts with the densely packed platoon on a highway and a single vehicle on an on-ramp. Then the vehicles must align themselves, meaning a space must be created in the densely packed platoon and the single vehicle must align itself with this space. Then, the new vehicle must join the main lane to form the platoon. The terminal state of the maneuver is then a steady-state platoon that includes the new vehicle.

This section will introduce some of the existing literature regarding automated merging maneuvers. Then, it will specify the on-ramp environment as

considered in this thesis. This specification includes aspects such as the on-ramp geometry and communication structure.

2.2.1 Literature Overview

An important topic in the research of Automated Driving System (ADS) is that of automated lane change maneuvers. This topic is close to that of automated merging maneuvers, especially when an acceleration lane is considered. In Nilsson et al. (2016) an automated system is developed to determine whether a lane change is desirable, select an appropriate inter-vehicle traffic gap and time instance, and calculate the corresponding longitudinal and lateral trajectory. The proposed algorithm is demonstrated by simulation and experimental results. It should be noted that in the context of merging into a platoon, the small inter-vehicle distances in the platoon cause a small average time gap on the main lane. It is unclear how the algorithm proposed in Nilsson et al. (2016) will react to such a situation. Depending on the density of the platoon, a lane change may be deemed undesirable, and there may be no appropriate inter-vehicle traffic gap.

The problem of appropriate spacing requirements for automated lane changes is widely discussed in literature (Kanaris et al., 2001). However, there are relatively few solutions for automated highway merging in congested traffic situations where appropriate spacing may not exist. One of such solutions is proposed in Chae et al. (2018). Here an automated vehicle aims to merge between two manually driven vehicles. The intended scenario in this research is one where the vehicles in the main lane are driving closely together, such that there is insufficient space for the merging maneuver. The new vehicle then needs to communicate its intention to merge with the human drivers. Since no inter-vehicle communication is available the new vehicle signals its intention by driving close to the main lane. The idea behind this is that the rear driver will slow down and create space. The solution is validated using experiments with one automated vehicle and two manually driven vehicles. In the context of highway merging, platoons can also be categorized as congested traffic in the sense that they typically do not have sufficient inter-vehicle space to allow for a safe merging maneuver. Since the lateral position can be used to predict cut-ins in cooperative platoons (Remmen et al., 2018), this solution may be applicable for platooning scenarios. However, due to the availability of inter-vehicle communication when a CAV joins a platoon, the potential for better solutions exists.

There are multiple examples of research on merging maneuvers with CAVs that include experiments. In the proposed algorithms, the creation of space in the platoon is specifically investigated. A gap can be created by increasing the desired distance between two platoon vehicles (Milanés et al., 2011; Raboy et al., 2021) or by switching the target vehicle and creating a *virtual platoon* (Lu et al., 2004; Ploeg et al., 2018). *Virtual platooning* is a technique to relate the trajectories of vehicles in two different lanes by mapping a vehicle onto a

one-dimensional path (Uno et al., 1999).

When the possibility of creating a gap in the platoon is introduced, finding an appropriate gap on the main lane is no longer required. However, the problem of *platoon sequencing* becomes relevant. Essentially, *platoon sequencing* is the process of determining the sequence of vehicles in the platoon after the merging maneuver. In other words, *platoon sequencing* is the process of deciding between which two platoon vehicles the new vehicle will merge.

In literature, multiple approaches for sequence management have been proposed. The merging sequencing approaches can be divided into three categories. The first category is distance-based approaches, often referred to as First-In-First-Out (FIFO) (Chen et al., 2020; Rios-Torres and Malikopoulos, 2017b) or alternatively First-Come-First-Serve (Dresner and Stone, 2008). In essence, when communication is established, vehicles are sequenced based on their one-dimensional distance. The distance-based method is easy to implement and gives good results in simulations when the initial velocities of all vehicles are similar. However, in highway on-ramp environments, large variations in the velocity can typically be expected (Cao et al., 2015). Therefore, distance-based approaches may be suboptimal because they implicitly assume an equal average velocity throughout the merging maneuver. Large differences in initial velocity will therefore result in large excitations to align the vehicles.

The second category of *platoon sequencing* algorithms is time-based approaches. Here the Estimated Time of Arrival (ETA) at the merging point is considered for sequencing. Using the ETA accounts for possible differences in the average velocity throughout the merging maneuver. The time-based approach is therefore more suitable for an on-ramp environment than the distance-based approach. Determination of the ETA is something that can be considered when designing the sequencing algorithm. In some research a vehicle model that allows for instantaneous acceleration changes is considered and it is assumed that the maximum acceleration is used to reach the desired velocity as soon as possible (Eiermann et al., 2020; Wang et al., 2018). The proposed trajectory is then not achievable with the vehicle model of (2.1), (2.2), and (2.3) or in practice because of the required instantaneous acceleration changes. Other approaches of determining the ETA, such as Zhou et al. (2018, 2019), may therefore be interesting in combination with this category of *platoon sequencing* algorithms.

The third category of *platoon sequencing* algorithms is optimization-based approaches. In essence, these approaches are used to produce desired trajectories of the vehicles involved to find an optimal merging sequence. The optimality criterion is defined by the user and varies between approaches. Examples include minimizing the time required for all CAVs to pass through the merging environment (Awal et al., 2013; Fukuyama, 2020; Mahbub et al., 2020), and minimizing a cost function based on the acceleration and jerk of all vehicles (Athans, 1969; Cao et al., 2014; Jing et al., 2019). By selecting a desired optimality criterion, the result for this type of approach can result in more desirable sequences

than those of the time-based approaches. For optimization-based approaches, the optimization strategy needs to be established. Some possibilities are conventional optimization solvers (Cao et al., 2014), recursive pruning algorithms (Awal et al., 2013), and game theory approaches (Fukuyama, 2020; Jing et al., 2019). In general, the optimization-based approaches are computationally expensive and the effect of measurement noise on their result is unknown. To the best of the author's knowledge, a direct comparison between optimization-based and time-based approaches does not yet exist.

Independent of the *platoon sequencing* approach, the maneuver requires a form of negotiation, control, and supervision over the wireless network. This ensures there is a distributed consensus over the actions expected from each vehicle. One approach to this problem is utilizing the *Collaborative Maneuver Protocol* as developed by Sawade et al. (2018). In essence, the protocol can be seen as a distributed state machine. The protocol creates a *session* in which each vehicle is given a *role* that it is expected to follow. This provides participants with knowledge about the expected behavior of another vehicle without information about the intended trajectory. One benefit of a *role*-based communication approach is the additional robustness. The trust in other vehicles adhering to communicated intended trajectories is limited. Vehicles may transmit false information deliberately or unknowingly. The usage of vehicle roles and information that is verifiable using on-board sensors is therefore beneficial. The development and implementation of such a protocol are beyond the scope of this thesis. However, it is important to consider this aspect to some extent when designing the merging control algorithm.

2.2.2 On-ramp Environments

An important topic in the investigation of automated merging maneuvers is the on-ramp environment. This section will specify the environment and network structure used throughout the remainder of this thesis. These choices are motivated by the existing literature.

Some research investigates highway merging maneuvers at multi-lane roads (Hang et al., 2021; Hu and Sun, 2019; Liu et al., 2021), but the majority of the work is focused on a single-lane road feeding into another single-lane road. The scenario of two single-lane roads is most suitable for an investigation into the merging into a platoon. Apart from the number of lanes, two main aspects need to be considered, namely, the road geometry and the network structure.

There are two categories for road geometry. In the first category, the on-ramp feeds directly into the main lane. In the second category, an acceleration lane is used to connect the on-ramp with the main lane. An acceleration lane is a lane parallel to the main lane where the vehicle can perform the lane change. The two categories are visualized in Figure 2.4. Using an acceleration lane there are thus multiple points for the new vehicle to enter the main lane, whereas there

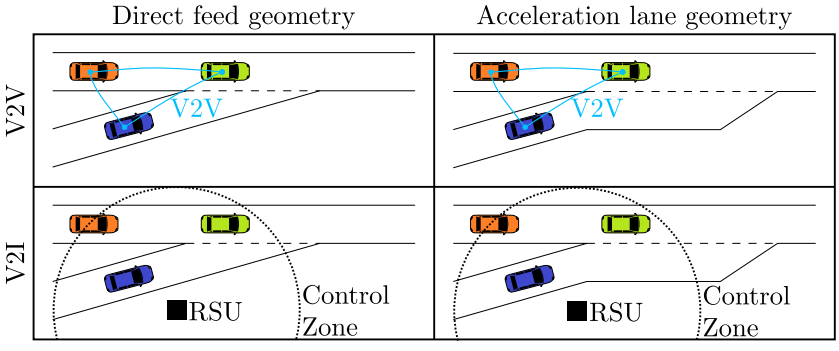


Figure 2.4: Examples of the geometry and communication network aspects of the on-ramp environment.

is only one point in the direct feed environment. Often the on-ramp environment is simplified to a one-dimensional path and only the longitudinal dynamics are investigated. The end of the acceleration lane can then be selected as the predetermined merging point and the geometry of the road is not important. However, some exceptions exist in which the lateral behavior is investigated, and an acceleration lane is considered (Cao et al., 2015; Eiermann et al., 2020).

There are two possible network structures for CAVs in on-ramp environments. The first is a network with only Vehicle-to-Vehicle (V2V) communication. Second, is a network with Vehicle-to-Infrastructure (V2I) communication where additional infrastructure, such as a Road Side Unit (RSU), assists in the maneuver. These two network structures are shown in Figure 2.4. Generally, when an RSU is used a control zone is defined in which the vehicles can communicate with the RSU. The communication network is related to the suitability of the control approach used in the solution. A centralized control approach is such that at least one task in the system is globally decided for all vehicles by a single controller (Rios-Torres and Malikopoulos, 2017a). A V2I network is very suitable for such an approach because the global controller can be run on an RSU. However, it is possible to run a centralized control approach using a V2V network by appointing a master vehicle that decides the tasks for the other vehicles (Hallé and Chaib-draa, 2005). A V2V network is more suitable for a decentralized control approach because all vehicles can directly obtain the required information from the other vehicles.

In this thesis, an acceleration lane is considered. The acceleration lane provides additional difficulty regarding spatial constraint because the longitudinal vehicle alignment and the lane change must be completed before the end of the acceleration lane. The distance required for a lane change is velocity dependent, which influences the end position of the alignment and two-dimensional path of the new vehicle. These problems are avoided when a direct feed environment

is used. The theory developed in this thesis can therefore easily be transferred to a direct feed environment. The communication network in this thesis is a V2V network because no additional infrastructure is required. The designed approach can therefore be implemented more easily. Furthermore, this network structure can easily be mimicked by a V2I network by passing all information directly through the RSU. The only additional challenge will then be a likely increase in communication delay. It is therefore expected that the developed control strategies can be applied to environments with a V2I network.

Chapter 3

Variable Gap Platooning Controller Design to Accommodate Merges in Cooperative Platoons

In this chapter, a cooperative variable gap platooning controller is developed. Its intended application is gap creation in cooperative platoons to accommodate merges with spatial restrictions. Due to the movement of the platoon, the main objective is to execute the maneuver in a predefined time. The controller design is based on a regular cooperative adaptive cruise control algorithm with an additional gap term in the desired inter-vehicle distance. The gap term can be varied over time and feedforward terms are used in the control law accurately track the desired gap. Experimental validation of the controller is performed with small mobile robots. The proposed control strategy is capable of opening the gap in a predefined time.

This chapter is based on Scholte et al. (2020)

3.1 Introduction

Connected Automated Vehicles (CAVs) can play an important role in tackling several current transportation challenges. For example, CAVs may reduce fuel consumption, emissions, and traffic congestion by using cooperative Cooperative Adaptive Cruise Control (CACC) (Rios-Torres and Malikopoulos, 2017a). Vehicles using CACC can drive close together in a string while transmitting their control input. Such a string of vehicles is often referred to as a platoon. The advantage of communication is that vehicles can drive closer together while *string stability* is maintained. *String stability* is a concept that describes the mitigation of perturbations down the string. If perturbations dampen, the platoon is said to be string stable. Without this property, longitudinal perturbations of the leader vehicle cause larger excitations at the back of the string. The growing excitations may cause emergency braking or even stopping. Therefore, it is dangerous and ineffective to employ platoons without *string stability*.

Loss of string stability may be caused by communication delays between the vehicles. String stability despite these delays can be achieved by using a velocity-dependent inter-vehicle distance. In essence, the desired inter-vehicle distance is a *headway time* multiplied by the vehicle's velocity. A constant *standstill distance* may be added, which essentially is a coordinate transformation and does not affect the dynamics. Experimental results using this control strategy have been presented in Ploeg et al. (2011). Due to the proven practical capabilities, this controller will be the basis of the controller design in this thesis.

For practical implementation of platooning, it should be possible to add new vehicles to a platoon while driving. Due to the small inter-vehicle distances, an obvious solution is to add vehicles at the front or back of the platoon. In an on-ramp environment, this can be impractical if the existing platoon is long as the new vehicle may need to wait. It is then preferable to have the new vehicle merge in between two platoon vehicles. Trivially, this requires an additional algorithm for the new and platoon vehicles. An overview of the current research on merging at highway on-ramps can be found in the survey of Rios-Torres and Malikopoulos (2017a). Platooning-specific techniques are discussed but are not the main focus of this survey. The examples of platooning in the survey do not employ a velocity-dependent inter-vehicle distance. Two platoons utilizing a velocity-dependent inter-vehicle distance are merged during the 2016 Grand Cooperative Driving Challenge (GCDC). The platoons employ a heuristic control strategy in which the vehicles linearly switch their target vehicle (Hult et al., 2018). At the end of this maneuver, all vehicles thus drive at a correct distance from their new target vehicle. During this alignment, the position error may grow, and timely execution of the maneuver is therefore not guaranteed. A similar problem in the context of nonsignalized intersections is identified in Vaio et al. (2019). A solution is proposed which has an exponential bound on the convergence of the average velocity of all vehicles to a desired velocity. However,

such an exponential bound may be undesired for the on-ramp merging scenario due to its large differences in initial velocity (Cao et al., 2015). An exponential bound may cause large accelerations at the start of the maneuver, and small accelerations at the end of the maneuver. Ideally, the acceleration trajectory is such that they are spread out evenly to lower the peak and average acceleration.

There is research regarding the optimal longitudinal trajectory during the merging maneuver (Ntousakis et al., 2016; Rios-Torres and Malikopoulos, 2017b; Wang et al., 2017). This research often does not specify lower-level control or the realization of these trajectories. An important aspect of this is switching to the desired platooning algorithm at the end of the maneuver. In Ntousakis et al. (2016) this is briefly discussed, and the controller is switched to an Adaptive Cruise Control (ACC) algorithm at the end of the maneuver. However, the possible initial error and its convergence are not investigated. This transient behavior is important to ensure the timely formation of a steady-state platoon.

This chapter proposes a variable gap platooning controller. This controller can be used for regular platooning and gap creation, which is a fundamental part of the merging maneuver. Since the same controller is used for platooning and gap opening, transient behavior is avoided. This helps ensure the timely execution of the maneuver as required due to the spatial constraints of the on-ramp environment. The vehicle model and controller design are discussed in Section 3.2, which is concluded with simulations. In Section 3.3 the basics of the trajectory design for the additional gap are explained. A combination of the controller and the trajectory design is analyzed in Section 3.4. The performance of the proposed controller is demonstrated with experiments using mobile robots in Section 3.5. The chapter is concluded in Section 3.6.

3.2 Control Strategy

The proposed control strategy is based on the vehicle model and controller of Ploeg et al. (2011). This strategy can be used for regular CACC driving and varying the inter-vehicle distance. The vehicle model and controller design are both discussed in this section.

3.2.1 CACC and Inter-vehicle Distance Control

The longitudinal dynamics of each vehicle are described using

$$\dot{q}_i(t) = v_i(t) \tag{3.1}$$

$$\dot{v}_i(t) = a_i(t) \tag{3.2}$$

$$\dot{a}_i(t) = \frac{1}{\tau}u_i(t) - \frac{1}{\tau}a_i(t). \tag{3.3}$$

Where q_i, v_i, and a_i are the 1-D position, velocity, and acceleration of the i^{th} vehicle. The control input is denoted with u_i and the driveline dynamics are

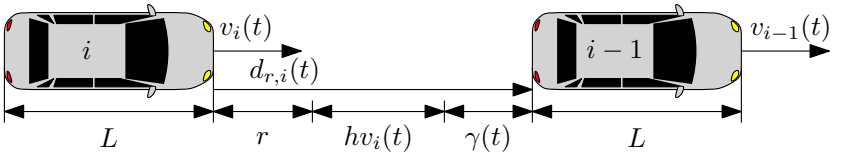


Figure 3.1: Definition of the platoon during steady state driving.

represented using time constant τ. The vehicle string is assumed homogeneous and therefore τ is equal for all vehicles.

An example of a platoon is shown in Figure 3.1 which consists of vehicles i and $i-1$. The distance between the front bumper of vehicle i and the rear bumper of vehicle $i-1$ is denoted with d_i. The vehicles are equipped with radar therefore vehicle i can measure d_i and its derivative $\dot{d}_i = v_{i-1} - v_i$. A controller aims to keep the vehicle at a desired inter-vehicle distance

$$d_{r,i}(t) = r + hv_i(t) + \gamma(t). \tag{3.4}$$

Where r and h denote the standstill distance and the headway time respectively. The purpose of r is to maintain a safe distance at low velocities. For the purpose of the merging maneuver, an additional gap γ is introduced, which is zero during normal driving. By choosing this notation, the additional gap does not influence the behavior of the controller during normal driving but can be used for gap opening and closing. The length of each vehicle is denoted as L. Wireless communication allows for data transfer between the vehicles. It should be noted that the quantity and quality of communicated data are limited by bandwidth, communication delay, and packet drops.

The error is defined as $e_i = d_i - d_{r,i}$, with the corresponding error states $\begin{bmatrix} e_{1,i} & e_{2,i} & e_{3,i} \end{bmatrix} = \begin{bmatrix} e_i & \dot{e}_i & \ddot{e}_i \end{bmatrix}$. This yields the error dynamics

$$e_{1,i} = d_i - d_{r,i} \tag{3.5}$$
$$e_{2,i} = v_{i-1} - v_i - \dot{\gamma} - ha_i \tag{3.6}$$
$$e_{3,i} = a_{i-1} - a_i \left(1 - \frac{h}{\tau}\right) - \ddot{\gamma} - \frac{h}{\tau} u_i \tag{3.7}$$
$$\dot{e}_{3,i} = -\frac{1}{\tau} e_{3,i} + \frac{1}{\tau} u_{i-1} - \frac{1}{\tau} \xi_i, \tag{3.8}$$

where

$$\xi_i = h\dot{u}_i + u_i + \ddot{\gamma} + \tau \dddot{\gamma}. \tag{3.9}$$

Based on (3.8) a definition of ξ_i is designed that controls the error dynamics and compensates for u_{i-1}, such that

$$\xi_i = k_p e_{1,i} + k_d e_{2,i} + u_{i-1}. \tag{3.10}$$

Where scalars k_p and k_d are control parameters. Now (3.9) and (3.10) yield the control law

$$\dot{u}_i = \frac{1}{h}\left(k_p e_{1,i} + k_d e_{2,i} + u_{i-1} - u_i - \ddot{\gamma} - \tau \dddot{\gamma}\right). \tag{3.11}$$

This control law can be used for gap opening. However, the designed trajectory of gap distance γ must have C^2 continuity such that $\dddot{\gamma}$ can be obtained at all times.

The stability of the individual vehicle's error dynamics is investigated. These dynamics are investigated by writing them in the form

$$\begin{bmatrix} \dot{e}_{1,i} \\ \dot{e}_{2,i} \\ \dot{e}_{3,i} \\ \dot{u}_i \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{-k_p}{\tau} & \frac{-k_d}{\tau} & -\frac{1}{\tau} & 0 \\ \frac{k_p}{h} & \frac{k_d}{h} & 0 & -\frac{1}{h} \end{bmatrix} \begin{bmatrix} e_{1,i} \\ e_{2,i} \\ e_{3,i} \\ u_i \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{h} & \frac{-1}{h} \end{bmatrix} \begin{bmatrix} u_{i-1} \\ \ddot{\gamma} + \tau \dddot{\gamma} \end{bmatrix}. \tag{3.12}$$

This system has an equilibrium in the origin for $u_i = 0$ and $\ddot{\gamma} + \tau \dddot{\gamma} = 0$. Similar to Ploeg et al. (2011), the Routh-Hurwitz stability criterion can be applied to the state matrix. It follows that the error dynamics are stabilized for $h > 0$ and any $k_p > 0$ and $k_d > 0$ that satisfy $k_d > k_p \tau$.

3.2.2 Controller Analysis

The proposed controller is compared to two feedback control strategies, which do not consider $\ddot{\gamma}$ and $\dddot{\gamma}$ in the computation of u_i. The feedback controllers differ in their computation error $e_{2,i}$. One controller uses the derivative $\dot{\gamma}$ in the error computation using (3.6). The other assumes γ to be constant and computes $e_{2,i}$ by only using measurements $v_{i-1} - v_i$ and a_i. The three controllers are referred to as *feedforward controller* (FF), *feedback controller assuming a differentiable* γ (FBD), and *feedback controller assuming a constant* γ (FBC) respectively.

Feedforward controller

The feedforward controller is designed in the previous section. It is subject to control law (3.11) and therefore requires a γ-trajectory with C^2 continuity. To isolate the influence of the gap opening maneuver, it is assumed that the preceding vehicle is driving at a constant velocity v_{nom}. In essence, $v_{i-1} = v_{nom}$, $a_{i-1} = 0$, and $u_{i-1} = 0$. Now, states x_i and outputs y_i are defined as

$$x_i = \begin{bmatrix} e_i, & v_i - v_{i-1}, & a_i, & u_i, & \dot{\gamma}, & \ddot{\gamma} \end{bmatrix}^\top, \tag{3.13}$$

$$y_i = \begin{bmatrix} e_i, & v_i - v_{i-1}, & a_i \end{bmatrix}^\top, \tag{3.14}$$

the system can be written in the form

$$\dot{x}_i = \begin{bmatrix} 0 & -1 & -h & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} & 0 & 0 \\ \frac{k_p}{h} & -\frac{k_d}{h} & -k_d & -\frac{1}{h} & -\frac{k_d}{h} & -\frac{1}{h} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\frac{\tau}{h} \\ 0 \\ 1 \end{bmatrix} \dddot{\gamma}, \tag{3.15}$$

$$y_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} x_i. \tag{3.16}$$

Using this linear time-invariant model, the system can be analyzed in the frequency domain. First, the frequency domain transfer function between the requested γ and position error e_i is investigated. The influence of γ rather than input $\dddot{\gamma}$ is examined because the units of γ and e_i are both meters. Based on the state-space model of (3.15) and (3.16), the relation between γ and e_i yields

$$\frac{e_i(s)}{\gamma(s)} = s^3 \frac{e_i(s)}{\dddot{\gamma}(s)} = 0. \tag{3.17}$$

Since γ is a C^2 continuous trajectory and $\dddot{\gamma}$ is its highest derivative used as control input in (3.11), e_i is unaffected by changes in γ when using this control law with the initial errors being zero. Therefore, the vehicle will maintain the requested inter-vehicle distance for any C^2 continuous γ-trajectory without creating an error.

The inter-vehicle distance of the vehicle in relation to the desired additional gap γ is another performance indicator. This shows the longitudinal behavior for changes in γ. It can be noted that $\frac{d_i(s)}{\gamma(s)} = \frac{-v_i(s)}{\dot{\gamma}(s)} = \frac{-a_i(s)}{\ddot{\gamma}(s)}$. The response in distance is thus strongly related to the response in velocity and acceleration. These responses can be obtained using input $\dddot{\gamma}$ and output v_i or a_i with the formulas $\frac{d_i(s)}{\gamma(s)} = -s^2 \frac{v_i(s)}{\dddot{\gamma}(s)} = -s \frac{a_i(s)}{\ddot{\gamma}(s)}$. This yields transfer function

$$G_{FF}(s) = \frac{d_i(s)}{\gamma(s)} = \frac{1}{1 + hs}. \tag{3.18}$$

The transfer functions can be seen as a low-pass filter. Since $h > 0$ the maximum gain of the functions is 1. In other words, the high frequency excitations of γ will cause excitations of d_i with smaller amplitude. Low frequency excitations of γ will be better tracked by the vehicle. This behavior can be explained using (3.4). If a gap is opened by increasing γ the vehicle slows down, this decreases the distance hv_i. Thus, at a time t_1 during a gap opening maneuver starting at time t_0, $\gamma(t_1) - \gamma(t_0) \geq d_{r,i}(t_1) - d_{r,i}(t_0)$. Therefore, d_i may still be changing at the end of the γ-trajectory as the vehicle is still adjusting its velocity. However, a suitable gap is available for the new vehicle when the γ-trajectory ends.

To obtain further insights in the behavior of the vehicle during the maneuver the transfer functions between $\dot{\gamma}$ and velocity, and $\ddot{\gamma}$ and acceleration are investigated. These functions yield

$$\frac{-v_i(s)}{\dot{\gamma}(s)} = -s^2 \frac{v_i(s)}{\dddot{\gamma}(s)} = \frac{1}{1+hs}, \tag{3.19}$$

$$\frac{-a_i(s)}{\ddot{\gamma}(s)} = -s \frac{a_i(s)}{\dddot{\gamma}(s)} = \frac{1}{1+hs}. \tag{3.20}$$

Both transfer functions can be seen as a low-pass filter. Since $h > 0$ the maximum gain of the functions is 1. In other words, the excitations of v_i and a_i are dampened versions of excitations in $\dot{\gamma}$ and $\ddot{\gamma}$. This behavior can be explained using (3.4). If a gap is opened by increasing γ the vehicle slows down, this decreases the distance hv_i. Thus, at a time t_1 in a gap opening maneuver starting at time t_0, $\gamma(t_1) - \gamma(t_0) > d_{r,i}(t_1) - d_{r,i}(t_0)$. Reversely, for a gap-closing maneuver hv_i increases while γ decreases. During a gap-opening maneuver, d_i may therefore still be change after the end of the γ-trajectory because the vehicle is not yet at the velocity of the preceding velocity. However, distance γ is available for the new vehicle to join the platoon.

Feedback controller assuming a differentiable γ

The feedback controller reacts to changes in the error and does not directly use the planned trajectory of γ. In essence, the control law (3.11) is replaced by

$$\dot{u}_i = \frac{1}{h}\left(k_p e_{1,i} + k_d e_{2,i} + u_{i-1} - u_i\right). \tag{3.21}$$

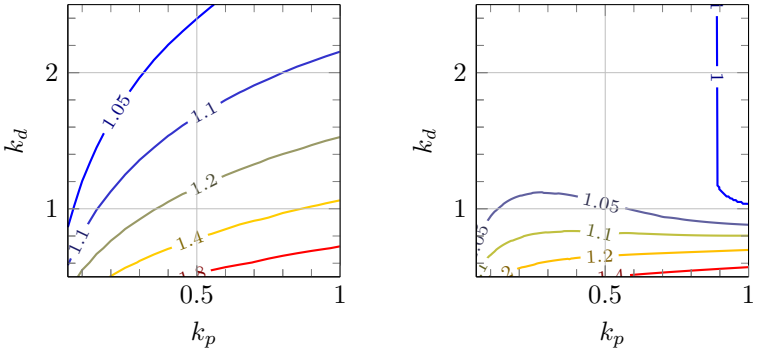
However, $e_{2,i}$ as described by (3.6) does contain the term $\dot{\gamma}$. It can be obtained by using knowledge of the designed γ-trajectory or through an observer. Using the states of (3.13) the state dynamics can be written as

$$\dot{x} = \begin{bmatrix} 0 & -1 & -h & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} & 0 & 0 \\ \frac{k_p}{h} & -\frac{k_d}{h} & -k_d & -\frac{1}{h} & -\frac{k_d}{h} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \dddot{\gamma}. \tag{3.22}$$

Using these equations and (3.16), the transfer function between γ and e_i is found to be

$$\frac{e_i(s)}{\gamma(s)} = s^3 \frac{e_i(s)}{\dddot{\gamma}(s)} = -\frac{s^2 + \tau s^3}{k_p + k_d s + s^2 + \tau s^3}. \tag{3.23}$$

At low frequencies, this transfer function has a small gain, the gain will go to 1 at high frequencies. In essence, at low frequencies the γ-trajectory can be



a: Headway $h = 0.5$ seconds. b: Headway $h = 1.5$ seconds.

Figure 3.2: $\sup_{\omega \in \mathbb{R}} |G_{FBD}(j\omega)|$ with $\tau = 0.1$ seconds for different values of k_p, k_d and h.

followed closely, and thus the error remains close to 0. At high frequencies, the γ-trajectory cannot be followed and thus the error has the same amplitude as the γ. The exact behavior is dependent on the system parameters. This behavior is confirmed by investigating the relation between the inter-vehicle distance and γ using (3.22), which yields the transfer function

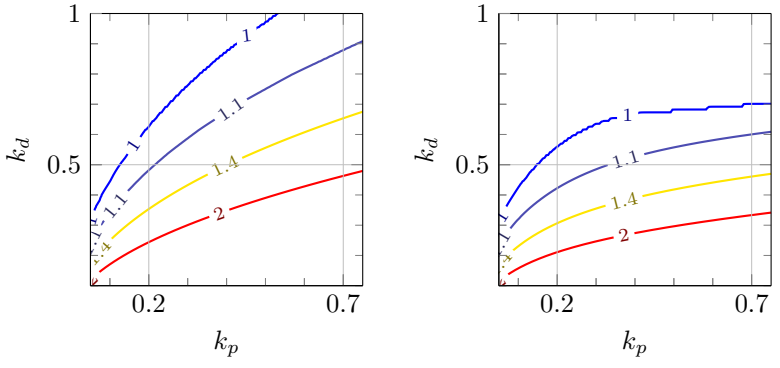
$$G_{FBD}(s) = \frac{d_i(s)}{\gamma(s)} = -s\frac{a_i(s)}{\dddot{\gamma}(s)} = \frac{k_p + k_d s}{k_p + k_d s + s^2 + \tau s^3}\frac{1}{1 + hs}. \tag{3.24}$$

The gain of this transfer function is close to 1 at low frequencies and goes to 0 at high frequencies. In other words, the additional inter-vehicle distance is approximately equal to γ at low frequencies. At high frequencies, this additional distance will be close to 0. The maximum gain of this transfer function is dependent on parameters k_p, k_d, h, and τ. In the controller design, parameters k_p, k_d, and h can be tuned. However, τ is a system property and cannot be adjusted. For this reason, the influence of k_p, k_d, and h have been analyzed for a given τ. The results of the analysis are shown in Figure 3.2. The maximum gain for some parameter sets is greater than 1. Therefore, γ-trajectories may be amplified in the d_i signal. For a large headway time, the maximum gain decreases and may even go to 1. For practical applications, the headway time can be constrained by other factors and increasing it may be infeasible.

Feedback controller assuming a constant γ

Without knowledge of $\dot{\gamma}$, the feedback control law of (3.21) can be combined with the error definition

$$e_{2,i} = v_{i-1} - v_i - ha_i. \tag{3.25}$$



a: Headway $h = 0.5$ seconds. b: Headway $h = 1.5$ seconds.

Figure 3.3: $\sup_{\omega \in \mathbb{R}} |G_{FBC}(j\omega)|$ with $\tau = 0.1$ seconds for different values of k_p, k_d and h.

The system with this controller can be written in the form

$$\dot{x} = \begin{bmatrix} 0 & -1 & -h & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} & 0 & 0 \\ \frac{k_p}{h} & -\frac{k_d}{h} & -k_d & -\frac{1}{h} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \dddot{\gamma}. \tag{3.26}$$

Using this system, the transfer function between e_i and γ is found to be

$$\frac{e_i(s)}{\gamma(s)} = s^3 \frac{e_i(s)}{\dddot{\gamma}(s)} = -\frac{k_d s + s^2 + \tau s^3}{k_p + k_d s + s^2 + \tau s^3}. \tag{3.27}$$

This equation is similar to (3.23) with an additional $k_d s$ term in the numerator. The general behavior is thus similar, this is confirmed by the transfer function

$$G_{FBC}(s) = \frac{d_i(s)}{\gamma(s)} = -s \frac{a_i(s)}{\ddot{\gamma}(s)} = \frac{k_p}{k_p + k_d s + s^2 + \tau s^3} \frac{1}{1 + hs} \tag{3.28}$$

which is comparable to (3.24). The difference becomes apparent when analyzing the maximum gain. The analysis for this system is shown in Figure 3.3. Using this controller, a maximum gain of 1 is obtainable with a larger parameter set. Since $\dot{\gamma}$ is ignored, the γ-trajectory is followed less aggressively. Thus, an overshoot in distance is less probable.

3.2.3 Transient Behavior

For a timely execution of the gap opening maneuver, the transient behavior of the controllers is investigated. First, the impulse response of the *FF* algorithm is analyzed using (3.18). This transfer function can be written in a state-space form as

$$\dot{x} = \left[\frac{-1}{h}\right] x + \left[1\right] u, \quad y = \left[\frac{1}{h}\right] x. \tag{3.29}$$

Using the standard A, B, and C matrix notation in this model, the impulse response $g(t)$ can be computed as

$$g(t) = Ce^{At}B = \frac{1}{h} e^{\frac{-t}{h}}. \tag{3.30}$$

The impulse response shows how the vehicle will return to a constant velocity v_{i-1} after the maneuver. Furthermore, it can be noted that this return is dependent on headway time h. Since $g(t) > 0 \; \forall \; t \in \mathbb{R}$ no overshoot of the desired gap is expected.

A similar analysis is performed for the *FBD* and *FBC* controllers using (3.24) and (3.28) respectively. Their impulse response is dependent on parameters k_p, k_d, τ, and h. A numerical example is used for this investigation, where $k_p = 0.2$, $k_d = 0.7$, $\tau = 0.1$ seconds and $h = 0.5$ seconds.

The corresponding A-matrix is equal for the *FBD* and *FBC*. In this numeric example its eigenvalues have strictly negative real parts. Therefore, the impulse response of (3.30) is bounded by

$$\|g(t)\| \leq ce^{-\lambda t}. \tag{3.31}$$

All eigenvalues of A have multiplicity equal to 1. Thus λ can be chosen as the absolute value of the largest real part of the eigenvalues of A (Hespanha, 2018). The chosen value of λ is 0.3660, the value of c is estimated numerically to be 0.9842 and 0.9464 for the *FBD* and *FBC* respectively. The resulting impulse responses of all controllers using the previously mentioned parameters can be found in Figure 3.4.

The impulse response of the *FF* is largest at $t = 0$ since the system reacts directly to changes in γ. Furthermore, there is no overshoot in d_i because $g(t) > 0$. The impulse response of *FBD* has a larger overshoot than that of *FBC*. The bound on the impulse response for *FBC* is slightly stricter than that of *FBD*.

To conclude, it is evident gap creation benefits from the feedforward approach. The feedforward terms in the control algorithm ensure that the system reacts instantaneously and adequately to changes in γ. To further investigate the behavior, a comparison using time simulations is provided in Section 3.4.

3.3 Trajectory Design

The previous section presented and analyzed the proposed controller design. To further investigate the performance of this design, simulations and experiments

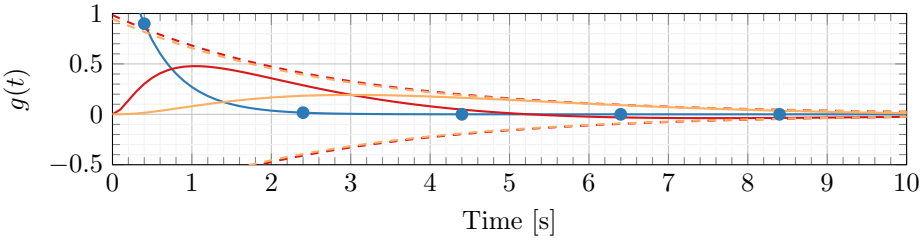


Figure 3.4: The $\frac{d_i}{\gamma}$ impulse response of the *FF* (—●—), *FBD* (——) and *FBC* (——), and the corresponding bounds for the *FBD* (- - -) and *FBC* (- - -).

are performed. An essential requirement for the simulations and experiments is the availability of an adequate γ-trajectory. This trajectory must be C^2 continuous and satisfy the desired initial and terminal conditions. This section proposes a method to compute such trajectories.

One way to obtain a smooth trajectory that satisfies constraints on the derivatives is the usage of a polynomial. A fifth-order polynomial can be used to describe a C^2 continuous γ-trajectory, such that three initial and three terminal conditions can be satisfied. The polynomial can be written as

$$\gamma(T) = c_1 + c_2 T + c_3 T^2 + c_4 T^3 + c_5 T^4 + c_6 T^5. \tag{3.32}$$

Where T is the time starting from the initiation of γ. Constants c_1 till c_6 are parameters used to give γ the desired behavior. Primarily, $\gamma(T)$ is designed to reach the gap size γ_{end} at time T_{end}, where T_{end} is the desired timespan of the gap opening maneuver. The trajectory of $\gamma(T)$ is designed such that $\gamma(T_{end}) = \gamma_{end}$, $\dot{\gamma}(T_{end}) = 0$, and $\ddot{\gamma}(T_{end}) = 0$. Initial values γ_{ini}, $\dot{\gamma}_{ini}$, and $\ddot{\gamma}_{ini}$ are considered such that the trajectory can be redesigned at any time. These conditions are fulfilled by selecting the constants

$$\begin{aligned}
c_1 &= \gamma_{ini}, \quad c_2 = \dot{\gamma}_{ini}, \quad c_3 = 0.5\ddot{\gamma}_{ini} \\
c_4 &= \frac{20\left(\gamma_{end} - \gamma_{ini}\right) - 3T_{end}\left(4\dot{\gamma}_{ini} + T_{end}\ddot{\gamma}_{ini}\right)}{2T_{end}^3} \\
c_5 &= \frac{-30\left(\gamma_{end} - \gamma_{ini}\right) + T_{end}\left(16\dot{\gamma}_{ini} + 3T_{end}\ddot{\gamma}_{ini}\right)}{2T_{end}^4} \\
c_6 &= \frac{12\left(\gamma_{end} - \gamma_{ini}\right) - T_{end}\left(6\dot{\gamma}_{ini} + T_{end}\ddot{\gamma}_{ini}\right)}{2T_{end}^5}.
\end{aligned} \tag{3.33}$$

In this chapter, a gap opening maneuver is used for the illustration through experiments and simulations. An example γ-trajectory for gap opening is shown in Figure 3.5. The parameters for the γ-trajectory, such as γ_{end}, in this graph are chosen purely for illustrative reasons and do not bear any significance. Since

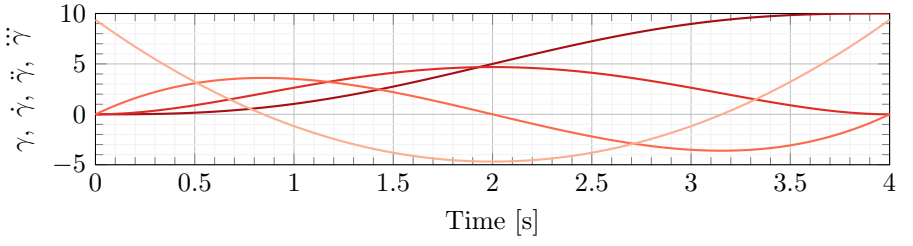


Figure 3.5: Example γ-trajectories from 0 to 10 meters in 4 seconds and their derivatives. With γ [m] (——), $\dot{\gamma}$ [m/s] (——), $\ddot{\gamma}$ [m/s^2] (——), and $\dddot{\gamma}$ [m/s^3] (——).

γ is a polynomial function of time, each derivative is a polynomial that is one order lower than the previous derivative.

For parameterization of the γ-trajectory, consider a following vehicle f opening the gap behind a preceding vehicle p. The γ-trajectory should be such that it can accommodate a new vehicle n when vehicle p is at a location $q_{end,p}$. The time constraint of the γ-trajectory should thus correspond with a spatial constraint using

$$\gamma_{end} = h_n v_p + L_n + r_n \tag{3.34}$$

$$T_{end} = \frac{q_{end,p} - q_p}{v_p}. \tag{3.35}$$

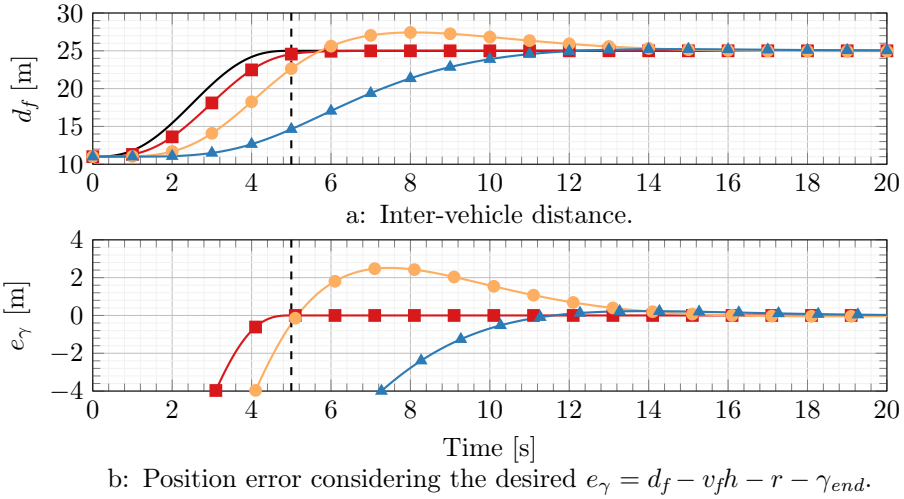
Where L_n is the length of the new vehicle, and subscripts p and n denote the vehicle to which a symbol belongs. It is assumed that the v_p is approximately constant and $v_f(T_{end}) = v_p(T_{end})$.

An important aspect of practical implementation is the handling of input constraints. The longitudinal input constraints of vehicles can generally be approximated by a combination of velocity and acceleration. In the case of the FF controller, the predicted velocity and acceleration trajectories can be computed using (3.19) and (3.20), or more sophisticated models that consider excitations of the preceding vehicle. For the other controllers, equivalent models can be used. Therefore, it is possible to assess the feasibility of the proposed γ-trajectory before executing it. If the γ-trajectory is infeasible, other solutions, such as a non-polynomial trajectory, can be proposed. In this chapter, the trajectory is assumed to be feasible in the simulations and experiments due to the choice of parameters used. However, the problem of infeasible trajectories is discussed in Chapters 4 and 5.

3.4 Simulations

Simulations for the gap opening maneuver with the different control algorithms are performed using a numerical example. These simulations consider a gap opening maneuver on a highway on-ramp to accommodate a tiny vehicle. The vehicles are simulated as one-dimensional point masses with the dynamics of (3.1) to (3.3). The chosen parameters are $v_p = 20$ m/s, $r = r_f = r_f = 1$ m, $L_f = 3$ m, $h = h_f = h_f = 0.5$ s, $\tau = 0.1$ s, $k_p = 0.2$, $k_d = 0.7$ and $T_{end} = 5$ s. The control parameters are taken from Ploeg et al. (2011). Based on v_p and T_{end} there are approximately 100 meters to open the gap. Distance d_f and error $e_\gamma = d_f - v_f h - r - \gamma_{end}$ are shown in Figure 3.6. In essence, $e_\gamma = 0$ implies that d_f can accommodate the merge of a *new* vehicle such that $d_i = v_i h + r \; \forall \; i \in \{f, n\}$. If $e_\gamma < 0$ the gap is too small and if $e_\gamma > 0$ there is additional space. In Hult et al. (2018), the merge is started if $|e_\gamma| \leq \epsilon_d$ where ϵ_d is a predefined threshold.

When the *FF* control algorithm is used, the behavior is such that $d_f(T_{end}) < hv_{nom}(T_{end}) + r + \gamma$, and $e_\gamma(T_{end}) = 0$. This is because $v_f(T_{end}) < v_{nom}$, the *new* vehicle can thus be accommodated but $d_f < d_n$. For a feedback controller, the *FBD* algorithm reaches $e_\gamma \geq 0$ earliest. However, it overshoots the desired gap and thus requires additional time to reach a stable situation with a small position error. The *FBC* algorithm reaches $e_\gamma \geq 0$ later, but it has a smaller overshoot, and the error is small earlier. When considering a feedback controller and a threshold ϵ_d for the merging maneuver, the *FBC* algorithm appears preferable.



a: Inter-vehicle distance.

b: Position error considering the desired $e_\gamma = d_f - v_f h - r - \gamma_{end}$.

Figure 3.6: Simulation results of the gap opening maneuver using three different controllers. With reference $hv_{nom} + r + \gamma$ (——), and the results for the *FF* (—■—), *FBD* (—●—) and *FBC* (—▲—) algorithm, with T_{end} (- - -).

When considering a merging maneuver, the *FF* controller is beneficial because $e_\gamma = 0$ at T_{end} and the *new* vehicle can therefore be accommodated at the desired time. The *FBC* algorithm is the better of the two feedback strategies. To further investigate the controllers under the influence of disturbances, experiments are presented in the next section. The *FF* controller is analyzed using the *FBC* algorithm as a benchmark feedback controller.

3.5 Experimental Demonstration

Experiments with mobile robots are used to demonstrate the performance of the proposed controller and compare it to the other permutations shown in this chapter. This section first describes the experimental setup, then the results are shown and analyzed. The first results describe the position error which is important to ensure safety. Then, the velocities are analyzed as an indication of the driving efficiency during the maneuver.

3.5.1 Experimental Setup

Experiments are performed using small differential-wheeled nonholonomic mobile robots (*e-pucks*) in a confined arena. The *e-pucks* were developed by Mondada et al. (2009) and an example is shown in Figure 3.7. Their left and right wheel are both connected to a stepper motor and can be actuated individually. Their control commands are computed on an external PC and transmitted wirelessly. The PC measures the vehicle poses using a camera above the arena. This localization system uses identifiers on top of the *e-pucks* as developed in Caarls (2009). The arena setup was previously used in Bayuwindra et al. (2020), where a more detailed description of the setup can be found.

The experiments are intended to demonstrate the performance of the controllers as well as the trajectory planning algorithm. Two distinct types of γ-trajectories are investigated. The proposed *feedforward* (*FF*) is analyzed and compared against a *feedback* (*FB*) control strategy. The *FBC* algorithm was used for the *feedback* controller. Two trajectory planning algorithms are investigated, Namely, the proposed polynomial shape described by (3.32), and a linear shape where γ is not C^2 continuous. The polynomial trajectory is used to illustrate the desired behavior of the controllers. It is expected that the controllers behave similarly when any other C^2 continuous trajectory is used. The linear trajectory starts at $\gamma(0) = 0$ and ends at $\gamma(T_{end}) = \gamma_{end}$. $\dot{\gamma}$ is well-defined throughout the trajectory, but $\ddot{\gamma}$ and $\dddot{\gamma}$ are not well-defined at $T = 0$ and $T = T_{end}$. This example of a non C^2 continuous γ-trajectory illustrates the necessity of C^2 continuity.

The proposed gap opening algorithm is intended for automotive applications. Therefore, the control is based on different longitudinal dynamics than those of

Further information can be found at http://www.e-puck.org.

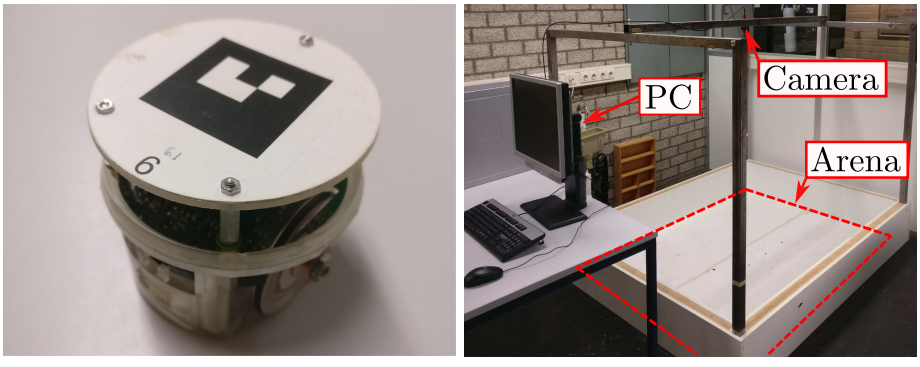


Figure 3.7: One of the robots used in this experiment (left) and the arena (right).

the *e-pucks*. The velocity of the *e-pucks* can be controlled directly. Thus, the longitudinal dynamics of (3.2) and (3.3) are considered in the computation of the control input. In essence, modeled accelerations and driveline dynamics are computed and stored on the PC to obtain the desired velocities.

The vehicles utilize a simple path following algorithm to drive laps around a specified path with a straight. The maneuvers are executed on the straight. Therefore, the lateral dynamics of the *e-puck* are not critical for the accurate modeling of an automotive application. Furthermore, a coordinate transformation is used to measure q_i, v_i, and a_i along the path. Due to this transformation u_i is adjusted. Furthermore, it is considered that the experiments have a centralized control setting. Meaning, a central PC computes the control inputs for all vehicles. However, the intended application of the algorithm is a decentralized platoon where each vehicle computes its own control inputs. The available knowledge of the vehicles is considered in the software on the PC. Moreover, communication delays are simulated by holding the control input u_p for one computation step. The system operates at approximately 25 Hz, thus the communication delay is around 0.04 seconds. All experiments were conducted using $\tau = 0.1$ seconds, $k_p = 0.2$, $k_d = 0.7$ $h = 0.5$ seconds and $r = 0.1$ meters.

The objective during the experiments is to open a gap of an appropriate size in the requested time. A secondary performance indicator is the amplitude of longitudinal excitations of vehicle f. These objectives are discussed in Sections 3.5.2 and 3.5.3 respectively.

3.5.2 Timely Execution and Position Error

For spatial or time-restricted merging scenarios, such as highway on-ramps, the timely execution of the gap opening maneuver is the primary objective. Error $e_{1,f}$ is used as an indicator of the maneuver completion because $e_{1,f} = 0$ implies $d_f = d_{r,f}$. An analysis regarding the safety of the maneuver can therefore focus

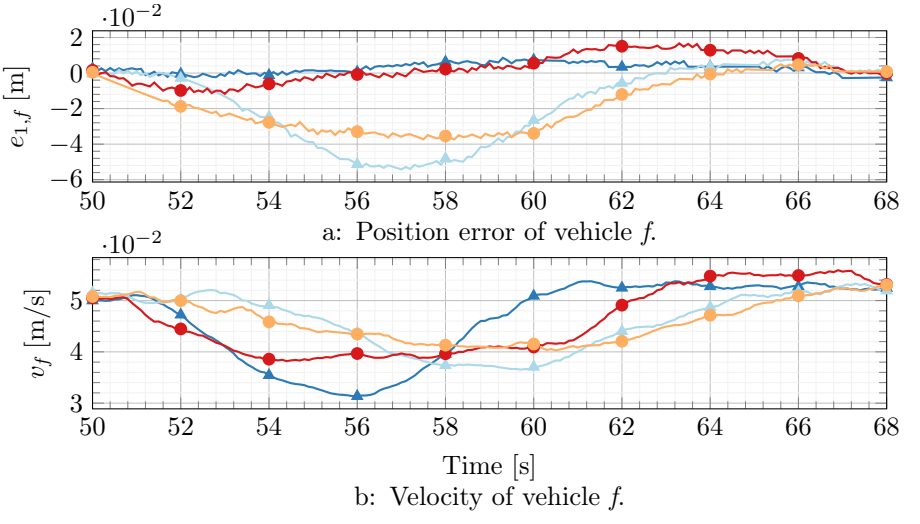


Figure 3.8: The behavior for the different γ trajectories and controllers during the gap opening maneuvers. For *FF Poly* (—▲—), *FB Poly* (—▲—), *FF Lin* (—●—) and *FB Lin* (—●—).

on the position error.

Experiments were conducted for a gap opening maneuver starting at $t = 50$ and ending at $t = 60$. Vehicle p is driving at 0.05 meters per second, and γ_{end} is 0.1 meters. The algorithms are denoted with *Poly* or *Lin* when using a polynomial or linear γ-trajectory respectively.

Figure 3.8 shows the error during the gap opening maneuver. The error when using the *FF Poly* algorithm remained close to 0. The *FF Lin* strategy introduces errors at $t = 50$ and $t = 60$, because $\ddot{\gamma}$ and $\dddot{\gamma}$ cannot be determined. However, the error goes to 0 when $\ddot{\gamma} = 0$ and $\dddot{\gamma} = 0$. It is shown that using this algorithm, the position error is zero at $t = 60$ but the velocity differs from the preceding vehicle. The error then grows a bit before it converges back to 0.

The *FB* control algorithms cause larger errors than both *FF* algorithms. The *FB Poly* algorithm causes a lower error at $t = 52$ but a higher peak error than the *FB Lin* algorithm. This can be attributed to the fact that γ feeds directly into the error definition. It is apparent that both *FB* control algorithms result in a negative position error at $t = 60$ and therefore the gap is not adequately large.

When $e_{1,f} \neq 0$, it is implied that the inter-vehicle gap is not the correct size and the maneuver is not yet completed. Therefore, only the *FF Poly* strategy has a timely maneuver execution. The other strategies do not satisfy the objective as the maneuver ends at approximately 67 seconds. Therefore, there is no

significant difference in the timeliness of these algorithms. In the context of a merge maneuver, this means that only the *FF Poly* strategy would result in a gap with the correct size at the predefined time. This is important to satisfy the spatial constraints of a highway on-ramp environment.

3.5.3 Longitudinal Excitations

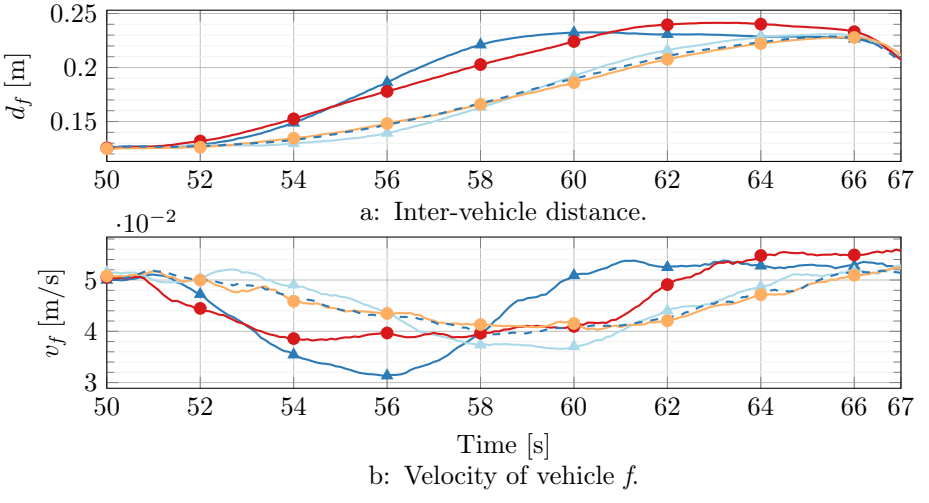
This section discusses the excitations introduced by the gap opening maneuver. The difference in velocity between the f and p vehicle is considered to isolate maneuver induced excitations. The performance indicator is the root mean squared of the velocity difference such that $v_{\mathrm{RMS}} = \mathrm{RMS}(v_f - v_p)$. This section will first show the inter-vehicle distances and velocities for the different algorithms. Then the corresponding v_{RMS} values are compared. The velocities cannot be measured directly therefore they are obtained by differentiating the positions. The resulting signals were filtered before the analysis to handle the noise.

Perturbations introduced by the gap opening maneuver are passed to subsequent vehicles in the platoon. Therefore, it is important that the excitations introduced during the maneuver are small. This section investigates the longitudinal excitations of vehicle f during the maneuver for the different control algorithms. Specifically, the inter-vehicle distances and velocities are discussed.

These experimental results were obtained simultaneously with the previous experiments. The inter-vehicle distance is approximately 0.125 meters for normal driving and 0.225 meters with an open gap. To objectively compare the excitations, the time required for the maneuver should be equal for all strategies. The previous results show that the *FF Poly* algorithm finishes at 60 seconds while the other algorithms finish at approximately 67 seconds. For a good comparison, an additional experiment using the *FF Poly* algorithm that finishes at 67 seconds is performed. The results for this experiment are denoted as *FF Poly (67 s)*. The results are shown in Figure 3.9.

The actual inter-vehicle distances and velocities without the coordinate transformation are analyzed. The maneuver was on a straight to minimize the effect of the transformation. However, at the end of the maneuver vehicle p takes a corner causing a decrease in absolute inter-vehicle distance.

The response during the gap opening procedure is shown in Figure 3.9. The *FF Poly* algorithm causes the largest velocity difference. This is to be expected since it opens the gap in the least amount of time. This emphasizes the necessity for the *FF Poly (67 s)* for a good comparison of the efficiency of the gap opening. The *FF Poly (67 s)* controller finishes the maneuver at 67 seconds, which is as expected. The behavior using this controller is very similar to that using the *FB Lin* controller. The *FB Poly* controller causes a larger maximum velocity deviation. The inter-vehicle distance increases later than with the *FF Poly (67 s)* algorithm. However, due to the higher velocity deviation, an appropriate gap is opened at a similar time.



a: Inter-vehicle distance.

b: Velocity of vehicle *f*.

Figure 3.9: Inter-vehicle distances and velocities for the different γ trajectories during the gap opening maneuver. For the *FF Poly* (—▲—), *FB Poly* (—▲—), *FF Lin* (—●—), *FB Lin* (—●—) and *FF Poly (67 s)* (- - -) algorithm.

Table 3.1: v_{RMS} for the different control strategies.

	Polynomial	Linear	Polynomial (67s)
Feedforward	0.0098 m/s	0.0086 m/s	0.0073 m/s
Feedback	0.0082 m/s	0.0071 m/s	n/a

The *FF Lin* algorithm is the most aggressive strategy. This strategy is the first to decelerate and reaches the requested inter-vehicle distance at 60 seconds. However, the vehicle is driving at a velocity $v_f \neq v_p$. Therefore, the maneuver is not yet finished. Afterwards, the velocity returned to the nominal value causing an overshoot in the inter-vehicle distance. The disturbances are then dampened over time to complete the maneuver.

The values of v_{RMS} between $t = 50$ and $t = 67$ can be found in Table 3.1. As expected, the *FF Poly* algorithm causes the highest excitations because the gap is opened the quickest. From the algorithms that complete the maneuver at 67 seconds, the *FF Lin* controller causes the highest excitations followed by the *FB Poly* controller. The *FF Poly (67 s)* controller performs slightly worse than the *FB Lin* controller. However, the previous analysis shows that their trajectories are very similar. The difference may thus be the result of measurement noise. Therefore, no significant loss in performance is established for the proposed *FF Poly* algorithm.

To conclude, the proposed *FF Poly* controller performs similarly to the *FB*

Lin controller. Both of them outperform the other permutations of the controller. It can therefore be concluded that the *FF Poly* controller does not show any reduction in performance even when the other controllers are changed so that the maneuver finishes at a similar time.

3.6 Discussion

Research on the merging of CAVs does not often consider CACC platoons. In prior research, the merging strategies for CACC platoons generally used heuristic methods for vehicle alignment. However, there is no focus on the fulfillment of time and spatial restrictions. In this chapter, the development of a cooperative platoon-based merging strategy is started by tackling the problem of gap opening. A scenario with a spatial constraint is considered. A variable gap control strategy is developed to merge maneuvers with spatial constraints, such as highway on-ramp scenarios. The controller uses feedforward terms to ensure the gap is opened in a predefined time. Due to the feedforward terms, there are some continuity constraints on the desired gap. Since the gap can be prescribed by the user, these constraints are easily met. The performance is demonstrated and analyzed using simulations and experiments with small mobile robots.

The proposed controller prescribes the desired longitudinal acceleration of the vehicle. To perform the merging maneuver, some higher-level control logic is required. In future work, the controller can be incorporated into the design of an on-ramp merging algorithm for CACC platoons. Chapter 4 proposes a control strategy for the merging maneuver based on the controller. In Chapter 6, the performance of the controller is demonstrated with passenger vehicles.

Chapter 4

A Control Strategy for Merging a Single Vehicle into a Platoon at Highway On-ramps

An important topic of research regarding cooperative platoons is merging vehicles into a platoon at highway on-ramps. This chapter proposes a control strategy for the merging of a single cooperative automated vehicle into a platoon of vehicles at highway on-ramps. The proposed strategy can handle large differences in initial positions and velocities, sensor noise, and disturbances caused by the platoon leader. Furthermore, the required controller transitions are designed such that the switch between regular platooning and the merging maneuver can easily be made by all vehicles. The proposed strategy is demonstrated using simulations. In the simulation environment communication delays, sensor noise, and disturbances of the platoon leader have been included. The proposed strategy is compared to a traditional strategy and shows a clear improvement in terms of noise handling. Furthermore, the proposed strategy behaves satisfactorily considering safety, efficiency, passenger comfort, and disturbance handling.

This chapter is based on Scholte et al. (2022)

4.1 Introduction

Road safety and traffic congestion are amongst the main challenges in current transportation systems. Dey et al. (2016) shows that road safety and traffic throughput can be improved using Cooperative Adaptive Cruise Control (CACC). This is a technique in which Connected Automated Vehicles (CAVs) drive closely behind each other and control their longitudinal motion using their on-board sensors and Vehicle-to-Vehicle (V2V) communication. Driving in such a string is sometimes referred to as platooning.

Platooning is a heavily researched topic due to its potential benefits. Multiple CACC control strategies have previously been proposed. One particular CACC control strategy that receives much attention is proposed in Ploeg et al. (2011). This controller design is validated and experimentally demonstrated. An extension of this control strategy is made in Chapter 3. This extension allows for gap-opening in the platoon to accommodate a new vehicle joining the platoon. The extended control strategy has an additional term in its error definition and its derivatives in a feedforward control law. The term and its derivatives can be set to zero for regular driving, in which case the resulting control is equal to that of Ploeg et al. (2011).

A topic of interest in the field of platooning is merging a new vehicle into an existing platoon at a highway on-ramp. Much of the work regarding the highway on-ramp merging of CAVs is summarized in the survey of Rios-Torres and Malikopoulos (2017a). Furthermore, this survey discusses the coordination at intersections since this is a similar scenario. The survey identifies the main distinction between control strategies as *centralized* and *decentralized* control strategies. In *centralized* approaches, a single controller globally decides at least one task for all vehicles. In *decentralized* strategies, each vehicle determines its own control policy based on communicated information about other vehicles.

Automated merging of CAVs on highways has been investigated extensively in recent years. One of the popular approaches for this problem is using a Model Predictive Control (MPC) scheme. Ntousakis et al. (2016), Rios-Torres and Malikopoulos (2017b), and Jing et al. (2019) each propose a control strategy in which each vehicle has an analytical optimal solution of its trajectory available. The trajectories can be solved numerically using the initial and desired final conditions when the maneuver is executed. The trajectories are updated periodically during the maneuver to correct for any measurement or actuator noise. A challenge arises when a vehicle gets closer to the merging location, the planning horizon for the optimal trajectory decreases. The control system may then become more sensitive to noise. This results in large longitudinal excitations of the vehicle. Furthermore, when noise is considered, it is difficult for the vehicles to accurately reach the exact predefined combination of time, position, velocity, and acceleration for merging. Therefore, when vehicles switch to the CACC algorithm, an initial error may result in undesired and potentially

dangerous situations. A switch to a CACC algorithm is not considered in the previously mentioned papers. However, Ntousakis et al. (2016) briefly discuss a switch to an Adaptive Cruise Control (ACC) algorithm.

Another example of MPC-based highway merging is found in Cao et al. (2015). In this work, the merging of one vehicle in front of or behind another at a highway on-ramp is investigated. The work is continued in Cao et al. (2019) where the problem of sensor noise is discussed. With the original method, noise could lead to the violation of state constraints in the optimization problem. This violation problem is solved by introducing three additional optimization variables which relax the equality constraints. In the proposed control scheme, the vehicles use the MPC algorithm when platooning and do not switch to a CACC algorithm. The planning horizon is set to be a fixed length and not dependent on the merging point. This avoids the problem of the small planning horizon. However, the disadvantage of this approach is that it becomes impossible to smoothly switch to a CACC strategy for platooning. Therefore, the benefits of driving with a CACC strategy cannot be obtained. The proposed solution of Cao et al. (2019) for enhanced robustness against sensor noise is not applicable when a controller switch is considered. Any relaxation of the terminal constraints to smoothen the planned trajectory would result in an initial error for the CACC algorithm. The problem would thus be shifted to the other controller. Using the CACC algorithm it cannot be guaranteed when initial errors have sufficiently dampened, causing potentially dangerous situations.

The highway merging problem can also be approached with a *fuzzy control approach*, in which there is a gradual change between two controllers. For instance, Milanés et al. (2011) proposes a control strategy that gradually changes the desired inter-vehicle distance based on the position of the preceding vehicle in the platoon. The controller design is validated using experiments at an on-ramp. A similar method is shown in Hult et al. (2018). The objective of this paper is to merge two platoons without the constraint of an on-ramp. The platoons are initially driving alongside each other. When a new platoon sequence is determined, the position error definition is gradually switched to align the vehicles. This switch is performed in the time domain rather than the spatial domain. In both studies, the initial inter-vehicle distances are relatively small. In real-world highway on-ramp scenarios, the differences may be larger. Therefore, the position error of platoon vehicles with respect to the new vehicle will be larger. A gradual switch is then not desired as large excitations may be introduced. Alternatively, one can wait with the controller switching until the vehicles are close. However, waiting longer before the maneuver is started will also result in higher excitations as the vehicles have less time to change their states. Furthermore, neither paper discusses the behavior of the error dynamics during the switch. It likely cannot be guaranteed that the error is sufficiently small at a predefined position or time to execute the merger.

A related application of fuzzy controllers is the control of CAVs at intersec-

tions. The main difference between this application and the highway merging scenario is that the vehicles are physically positioned far apart. One example of research on this topic is Milanés et al. (2010). This work investigates an intersection where two vehicles with communication cross. One of the vehicles is human-driven and the other is automated. When the vehicles are within 80 meters of each other, the automated vehicle decides its control actions based on the other vehicle. If the human-driven vehicle is coming from the right, it has priority, and the automated vehicle adjusts its trajectory. Fuzzy logic is used to determine the position of the gas and brake pedals based on the distance of both vehicles to the intersection point and the speed difference between the vehicles. Experiments show that the designed control strategy performs adequately. Furthermore, in Onieva et al. (2012) a similar scenario is investigated in which a *Fuzzy Rule-Based System*, determines whether an intersection action is necessary, and calculates the desired velocity. The parameters of the control system are tuned using a genetic algorithm. A large number of simulations are used to tune the controller and demonstrate its performance. It should be noted that in both these examples, fuzzy logic is used to determine the control inputs of the individual vehicle rather than to switch between two control targets. For merging in a platoon, switching between two control targets is one of the main objectives. The vehicle in the platoon must change its original target to the new vehicle. Likewise, the new vehicle must switch its individual controller to a cooperative one. Therefore, the type of fuzzy control used at intersections is unsuitable for the highway merging scenario.

Some recent research on intersection control of CAVs focuses on *virtual platooning*. This is a concept in which the vehicles at intersecting roads set up a platooning algorithm based on the one-dimensional distance to the intersection point. The platooning algorithm ensures that there is sufficient inter-vehicle distance for a safe crossing of the intersection. Examples of such research include Morales Medina et al. (2018) and Vaio et al. (2019). Due to the usage of platoons at intersecting roads, this solution has the potential to be more applicable to the highway on-ramp merging scenario. However, one major difference between the two scenarios is the considered range of differences in the initial states of the vehicles. In the previously mentioned work, the initial velocities of the vehicles are relatively close to their desired terminal velocities during experiments and simulations. For the highway on-ramp merging scenario, the new vehicle can initially be driving up to half of the velocity of the main lane platoon (Cao et al., 2015). It is difficult to assess how the algorithm will handle these large errors. Dependent on the tuning, the algorithm may result in high longitudinal excitations, or it may not be able to reduce the error enough. It should be noted that Vaio et al. (2019) provides an exponential bound on the convergence of the average velocity of all vehicles to a desired velocity. However, this bound does not prevent excessive excitations due to its exponential nature. Furthermore, in a platooning scenario, it is possible that disturbances are caused by

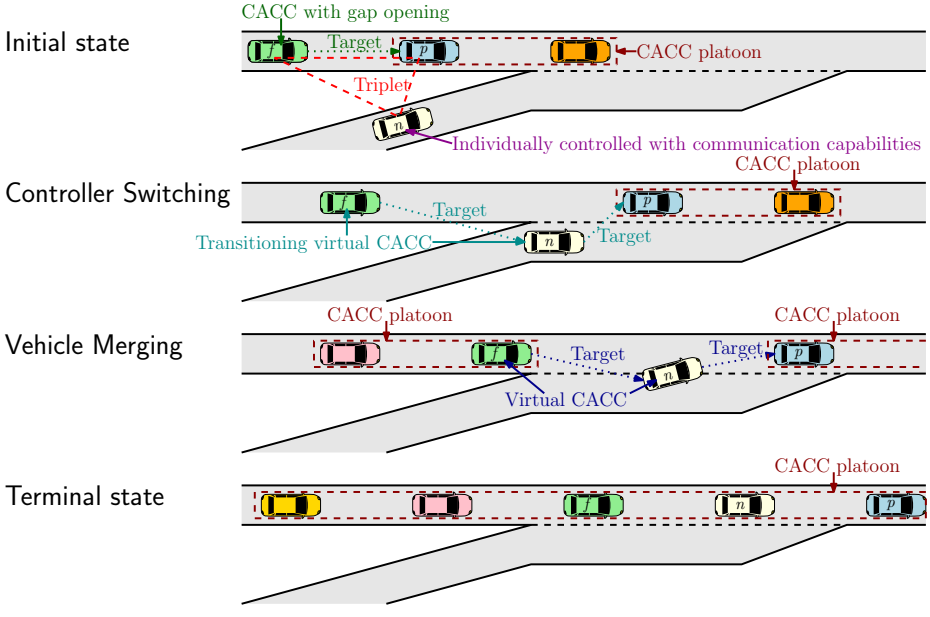


Figure 4.1: A visual summary of the proposed control system.

the behavior of vehicles ahead of the platoon. This may alter the convergence of the controller. Therefore, a control strategy that specifically considers large variations in initial conditions is desired for highway on-ramp merging.

In this chapter, a decentralized merging control strategy for CAVs in highway on-ramp environments is proposed. More precisely, the control strategy is designed for the merging of a single CAV into a CACC platoon. The strategy is briefly explained in Figure 4.1. Initially, a platoon of vehicles is driving on the main lane while an individually driven CAV is on the on-ramp. A *triplet* exists of the on-ramp vehicle and the two platoon vehicles between which it will merge. One vehicle in the platoon opens a gap by switching to a variable gap CACC controller. Then it creates a virtual platoon with the new vehicle using a transitional CACC controller and a virtual platooning approach. At the end of the maneuver, the transition to a conventional CACC controller is completed. The new vehicle is initially driving individually and switches to a transitional CACC controller with a virtual platooning approach. At the end of the maneuver, all vehicles involved form a steady-state platoon of CACC vehicles. The method can handle large differences in the initial states of the vehicles while being relatively insensitive to noise. The main contributions of this chapter are:

1. A novel control strategy for highway on-ramp merging maneuvers is proposed. The control strategy is a combination of CACC feedback, variable

gap CACC, and MPC controllers. The most important aspects of this strategy are handling large differences in the initial vehicle states, and being less sensitive to sensor noise and speed variations of the lead vehicle than traditional methods.

2. The chapter presents a method of transitioning between the various controllers. This reduces unwanted disturbances caused by controller switching.

3. The following vehicle needs to consider the original preceding vehicle and the new merging vehicle simultaneously. We introduce the usage of a CACC controller with the most relevant vehicle and, if needed, using a collision avoidance controller with the secondary vehicle.

4. The effectiveness and performance of the proposed control strategy are demonstrated using simulations. Measurement noise and communication delays are included. Furthermore, scenarios including velocity variations of the platoon leader are analyzed.

This chapter starts with a definition of the problem in Section 4.2. This includes the vehicle model and the proposed controller dynamics of the individual vehicles. In Section 4.3 the control strategy of the multi-vehicle system is defined. The section proposes a high-level controller regarding the roles of the vehicles, an optimized trajectory design, and a communication strategy. Simulations with the proposed control strategy are presented in Section 4.4. The simulations include a comparison with a traditional MPC strategy and specific scenarios to demonstrate the behavior. The noise sensitivity is analyzed separately using a large number of simulations. Conclusions and recommendations are given in Section 4.5.

4.2 Problem Statement

This section discusses the problem of a cooperative highway on-ramp merging scenario. Furthermore, the variable gap CACC controller is briefly explained. This controller is discussed in more detail in Chapter 3.

4.2.1 Highway On-ramp Merging Scenario

This section describes the merging maneuver. A graphic representation of the initial situation is provided in Figure 4.2. In essence, we consider a single vehicle at a highway on-ramp that joins a platoon of vehicles on the main lane. All vehicles considered in this maneuver are CAVs with CACC capabilities. We define a set of vehicles in the platoon $\mathcal{P}$ and a set of new vehicles $\mathcal{N}$. It is assumed all vehicles have a map of the environment such that information regarding their position with respect to the on-ramp is locally available. Moreover, we assume

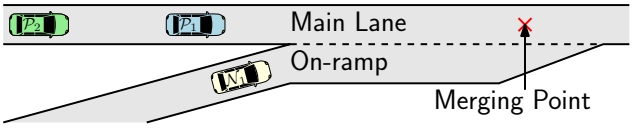


Figure 4.2: The initial situation of a highway on-ramp merge scenario.

that each vehicle has information of the location of a predefined *merging point*. The *merging point* is the position at which the on-ramp ends, and new vehicles must thus be on the main lane. This is an environmental feature and thus equal for all vehicles. This concept is used in other research to represent spatial constraints of a highway on-ramp (Jing et al., 2019; Lu and Hedrick, 2003). The goal is to have all vehicles in sets $\mathcal{P}$ and $\mathcal{N}$ create a single platoon on the main lane after the new vehicle performs a lane change. More specifically, the scenario in which the new vehicle joins between the two existing platoon vehicles is investigated. This scenario is relevant because for longer platoons it may not always be possible to join in front of or behind the existing platoon. Implicitly, a transition between controllers for some of the vehicles is required between the initial and terminal situation.

A decentralized controller is considered. This is a control strategy where all the controls are computed on the vehicles. It is the opposite of a centralized control strategy where often a *roadside unit* assists in the computation of the controls. The advantage of a centralized control strategy is that more information is available for the computation of the controls. However, additional infrastructure may be required for a centralized controller. A decentralized controller requires additional care regarding the design of the communication protocols and available information.

All vehicles in set $\mathcal{P}$ are initially assumed to be in a steady-state platoon and thus have the same initial velocity. Conversely, vehicles on an on-ramp typically have a lower velocity than the vehicles on the main lane (Cao et al., 2015). Therefore, the distance traveled between the first moment of communication and the time of the merge is typically lower for the on-ramp vehicle. Thus, the final position of the new vehicle in the platoon is generally not near the initial relative position of the vehicle.

To conclude, the main challenges found in the highway on-ramp merging scenario include handling of large initial differences between vehicle states and reaching the desired terminal states, transitioning the controllers of the individual vehicles such that a CACC platoon is formed containing all vehicles, and completing the maneuver within a predetermined space which is dictated by the design of the on-ramp. The proposed control strategy achieves these challenges and can do so in the presence of sensor noise.

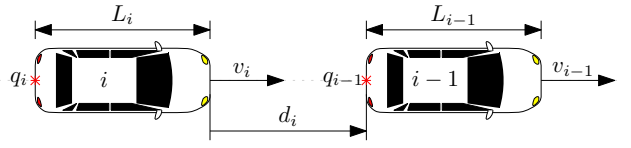


Figure 4.3: The coordinate and parameter definition of a platoon with two vehicles.

4.2.2 Vehicle Model and Vehicle Controller Design

The CACC control strategy assumes multiple vehicles driving along a one-dimensional path. The location of the vehicles along this path is denoted with q_i where subscript i denotes the vehicle's identifier. The position of each vehicle is measured at the middle of its rear bumper. The distance between the front bumper of a vehicle i and the rear bumper of a vehicle $i-1$ is denoted as d_i. The distance between points q_i and q_{i-1} can thus be described using d_i and vehicle length L_i. In essence, $d_i(t) = q_{i-1}(t) - q_i(t) - L_i$. The relation between the positions and inter-vehicle distance is visualized in Figure 4.3.

Each vehicle is assumed to be equipped with perception sensors such that it can measure the relative position and velocity of other vehicles. Furthermore, wireless communication is required for the CACC algorithm. Based on Ploeg et al. (2011) the following vehicle model is adopted

$$\dot{q}_i(t) = v_i(t), \tag{4.1}$$
$$\dot{d}_i(t) = v_{i-1}(t) - v_i(t), \tag{4.2}$$
$$\dot{v}_i(t) = a_i(t), \tag{4.3}$$
$$\dot{a}_i(t) = \frac{1}{\tau}u_i(t) - \frac{1}{\tau}a_i(t). \tag{4.4}$$

Where v_i, a_i, and u_i denote the velocity, acceleration, and external input of vehicle i respectively. The external input can be interpreted as the desired acceleration since (4.4) shows that a_i will converge to u_i. The desired acceleration cannot be achieved instantaneously. Therefore, the driveline dynamics are simulated using time constant τ. The constraints on the vehicle dynamics are not explicitly considered in this model. Instead, these constraints will be considered in the controller transition strategy and sequence manager.

Now, the variable gap CACC controller as described in Chapter 3 is presented. The desired inter-vehicle distance is determined using a constant time gap spacing policy. This distance $d_{r,i}$ is defined as

$$d_{r,i}(t) = r_i + h_i v_i(t) + \gamma_i(t), \tag{4.5}$$

where r_i and h_i denote a constant standstill distance and headway time. The variable $\gamma_i(t)$ is used for gap opening to accommodate the merge or to close a

residual gap during the controller transition. The position error is defined as

$$e_i(t) = d_i(t) - d_{r,i}(t). \tag{4.6}$$

The dynamics of the error are used to design a controller. From this point on, time argument t will be omitted where possible for readability. The controller design is based on the error states,

$$\begin{bmatrix} e_{1,i} & e_{2,i} & e_{3,i} \end{bmatrix}^\top = \begin{bmatrix} e_i & \dot{e}_i & \ddot{e}_i \end{bmatrix}^\top. \tag{4.7}$$

The error dynamics yield

$$e_{1,i} = d_i - d_{r,i} = q_{i-1} - q_i - L_i - r_i - h_i v_i - \gamma_i \tag{4.8}$$

$$e_{2,i} = v_{i-1} - v_i - \dot{\gamma}_i - h_i a_i \tag{4.9}$$

$$e_{3,i} = a_{i-1} - a_i \left(1 - \frac{h_i}{\tau}\right) - \ddot{\gamma}_i - \frac{h_i}{\tau} u_i \tag{4.10}$$

$$\dot{e}_{3,i} = -\frac{1}{\tau} e_{3,i} - \frac{1}{\tau} \xi_i + \frac{1}{\tau} u_{i-1}, \tag{4.11}$$

where

$$\xi_i = h_i \dot{u}_i + u_i + \ddot{\gamma}_i + \tau \dddot{\gamma}_i. \tag{4.12}$$

A control law for ξ_i that stabilizes the error dynamics can now be chosen. Using (4.11) the chosen control law is

$$\xi_i = k_p e_{1,i} + k_d e_{2,i} + u_{i-1}, \tag{4.13}$$

where scalars k_p and k_d are control parameters. Now (4.12) and (4.13) yield the control law

$$\dot{u}_i = \frac{1}{h} \left(k_p e_{1,i} + k_d e_{2,i} + u_{i-1} - u_i - \ddot{\gamma}_i - \tau \dddot{\gamma}_i \right). \tag{4.14}$$

It should be noted that the designed trajectory of gap distance γ_i requires C^2 continuity such that $\dddot{\gamma}_i$ can be obtained at all times. It can be shown that the error dynamics of the individual vehicles are stabilized for $h_i > 0$ and any $k_p > 0$ and $k_d > 0$ that satisfy $k_d > k_p \tau$ (Ploeg et al., 2011).

4.3 Merging Control Strategy

The proposed control strategy comprises multiple controllers that interact with each other. A complete overview of these controllers is provided in Figure 4.1. This section will explain the control strategy in detail. First, the initial state for the application is introduced. Then an overview of the components within the control strategy is given. Lastly, the various individual controllers are discussed separately.

4.3.1 Initial State

To introduce the merging control strategy, the initial state of its intended use is introduced. The strategy is designed for the merging of a single CAV into a CACC platoon at highway on-ramp environments. The strategy is executed after the position of the new vehicle within the platoon is defined. A group of three vehicles, called a *triplet*, is formed. The concept of *triplets* was previously used in Ploeg et al. (2018). The vehicles in this *triplet* are referred to as the preceding (p), new (n), and following (f) vehicle.

In literature, the methodology of selecting the position in the platoon is referred to as merging sequence management. One possibility is using a First-In-First-Out (FIFO) algorithm. In essence, this algorithm defines a *control zone* around the merging point. The order in which the vehicles enter the control zone is equal to the order in which they exit the control zone (Rios-Torres and Malikopoulos, 2017b). In other words, the sequence is thus determined using the distance to the merging point. The disadvantage of this method is that it has difficulty handling large differences in initial velocity. Some research approaches this problem by comparing the Estimated Time of Arrival (ETA) at the merging point (Eiermann et al., 2020; Wang et al., 2018). Other research tackled this problem by investigating optimal merging sequences based on the required trajectories of all vehicles involved (Cao et al., 2015; Jing et al., 2019). An optimization-based approach may yield better results, but it is more computationally heavy. Furthermore, the effect of measurement noise on the optimization is unknown. Therefore, merging sequence management is an ongoing field of research.

Due to its size, the problem of merging sequence management is outside of the scope of the current chapter and will be addressed in Chapter 5. The initial state of the proposed control strategy in this chapter includes a predefined *triplet*. It is assumed that this *triplet* is feasible, meaning that the merging maneuver can be executed with smooth trajectories within reasonable bounds for the longitudinal acceleration and jerk. For example, if the distance to the merging point is small and there are large velocity differences between the new vehicle and the platoon, the *triplet* may be infeasible. It is then not possible for the new vehicle to match the velocity of the platoon when entering the main lane. The exact conditions for a feasible *triplet* are discussed in Chapter 5 when the merging sequence manager is investigated.

It should be noted that a standard CACC strategy is used for cooperative driving outside of the maneuver. Therefore, the proposed strategy is bound to communication and automation restrictions posed for such CACC strategies. Wireless communication failures need to be addressed using the existing strategies for CACC driving (e.g., Ploeg et al. (2015)). Furthermore, if the communication with the new vehicle is lost the event can be handled as a cut-in by a non-cooperative vehicle (e.g., Milanés and Shladover (2016)). These existing fallback strategies are deemed sufficient to handle these scenarios. This belief is

strengthened by the fact that the proposed strategy does not impose additional risks compared to regular CACC driving. Further investigation into handling such events is therefore outside the scope of the current work.

4.3.2 Controller Overview

There are three main steps in the merging maneuver control strategy. These steps are visualized in Figure 4.1. In short, the steps are:

1. **Vehicle Alignment** When vehicles n and f have confirmed their role in the triplet, the vehicles will align themselves. Vehicle p does not need to give consent as its behavior is unaffected by the maneuver. Vehicle f opens a gap in the platoon while vehicle n drives to the desired position. Vehicle p continues driving as it normally would and reacts to what is ahead of it.

2. **Lane Change** At some point, when vehicles n and f have been sufficiently aligned, the lane change is initiated. Vehicle n merges into the main lane. At this point, vehicle f must remain at a safe distance to vehicles p and n.

3. **Platoon Formation** The merge is completed when vehicle n joins the platoon. The sequence of the platoon is redefined with vehicle n between vehicles p and f. Vehicles n and f thus drive with a CACC controller behind vehicles p and n respectively.

The vehicle alignment is one of the main contributions of the proposed control strategy. The longitudinal trajectories of vehicles n and f are important for the alignment. Vehicle n has the task to reach the desired position and vehicle f aims to create a gap that can accommodate vehicle n. Vehicle p is not involved in the alignment since its behavior is dependent on external factors, such as the excitations of a platoon leader. To explain the proposed control strategy for the vehicles during the alignment, an overview of the different controllers utilized is given in Figure 4.4. One of the most important time instances in this overview is t_{mp}, this is the moment at which vehicle n reaches the merging point. It should be noted that the merging point is an environmental feature. Assuming a steady-state platoon is achieved, t_{mp} is determined by the trajectory of vehicle p and some controller parameters such as headway time. Thus, t_{mp} cannot be changed directly by the control strategy. To ensure vehicle n is in the main lane at t_{mp}, time instance t_{lc} is defined as the moment at which vehicle n starts its lateral movement. The instance t_{lc} can be chosen such that there is sufficient time for a comfortable lane change maneuver.

Vehicle f is initially driving with a variable gap CACC controller behind vehicle p. At this moment the vehicle is opening a gap to facilitate vehicle n. When possible, the vehicle switches to a transitioning CACC controller targeting vehicle n which is its new target vehicle. The transitioning controller is based on the variable gap controller and is used to ensure a timely transition. In

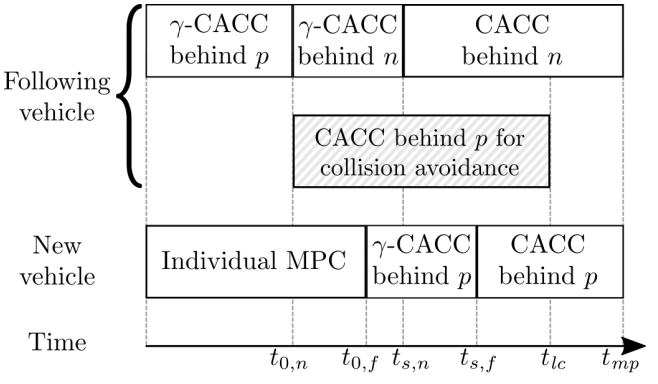


Figure 4.4: An example of the control strategy to provide an overview of the different controllers and when they are active.

essence, the γ-term and its derivatives are initialized such that the perceived initial error is zero. Then the terms are varied over time such that a steady-state platoon is obtained. The error will remain small throughout the transition which helps to achieve the goal of a timely transition. The time instance at which the transition starts is denoted as $t_{0,f}$. The time instance at which the transition is completed is defined as $t_{s,f}$. One constraint is that $t_{s,f}$ must be before or at t_{lc}. This ensures that a virtual steady-state platoon is obtained during the lateral maneuver. From instance $t_{0,f}$ up until t_{lc}, vehicle f needs to consider two vehicles. The main controller targets vehicle n, but a rear-end collision with p must be avoided. Therefore, an additional CACC algorithm targeting vehicle p is run in the background and used for collision avoidance. The desired acceleration provided by this additional controller is only followed when a safety-critical situation arises.

The control strategy of vehicle n is similar to that of vehicle f. The main difference is that vehicle n is initially using an individual MPC algorithm. The time at which the control switches to a transitioning CACC algorithm targeting vehicle p is denoted as $t_{0,n}$. Then, at time $t_{s,n}$ the transition to a traditional CACC algorithm is completed. Once more, $t_{s,n}$ should be before or at t_{lc} to ensure that a virtual platoon is established during the lateral movement.

It can be noted that the time instances regarding the controller transitions of both vehicles are not directly related. It is thus possible for any vehicle to start and finish its controller transition before the other vehicle starts its transition. However, since time instance t_{lc} is used as a reference for both vehicles, the transitions will be completed at the desired moment. This emphasizes the need for shared information in this control strategy. The proposed control strategy is intended for a decentralized control scheme. For this reason, inter-vehicle communication is important to ensure sufficient information is available for each

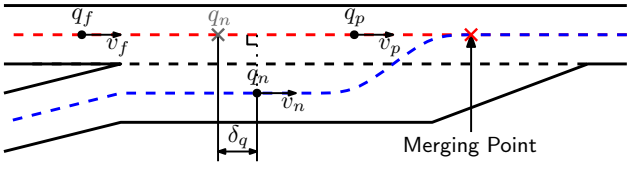


Figure 4.5: The main lane vehicle path (red) and on-ramp vehicle path (blue).

vehicle. To validate the applicability of the proposed control strategy, the information communicated between the vehicles is discussed in Section 4.3.6.

4.3.3 Vehicle Alignment

The longitudinal control of vehicles n and f is subject to the vehicle alignment strategy. Vehicle f is responsible for creating a gap to accommodate vehicle n. Vehicle n is responsible for aligning itself with this gap. These actions are performed simultaneously by using vehicle p as a reference. However, the control strategies will differ between vehicles n and f. Vehicle f is driving with a cooperative controller behind vehicle p during the alignment maneuver. Alternatively, vehicle n uses an individual controller in the first part of the maneuver. Therefore, their control strategies are discussed separately.

First, a coordinate system is discussed through which the vehicles are related. It is assumed that the main lane is a straight path. The path of the on-ramp vehicle is more complex. The path is divided into three segments. The first segment is the on-ramp, this includes a part driving to the highway and a part of the acceleration lane parallel to the main lane. This segment is followed by the lane change. The path of the lane change is designed by vehicle n. After the lane change, the third segment is started which coincides with the main lane path. The paths are visualized in Figure 4.5.

The position along the path is denoted with q. The merging point is defined as q_{mp} for both paths. This point is used to align the coordinates of both paths. When aligning the parallel part of the on-ramp path with the main lane it is important to consider the additional distance (δ_q) vehicle n travels due to the lane change. δ_q is calculated before and during the lane change. In Figure 4.5, δ_q is visualized as the longitudinal distance between the position of vehicle n and a projected version of vehicle n on the main lane. This distance can be used by vehicles n and f to align themselves with vehicles p and n respectively. The value of δ_q is based on the planned lateral trajectory and is elaborated upon later.

The vehicles are modeled as point masses moving along these paths. This choice is justified because the controller mainly focuses on longitudinal behavior. If the vehicle remains in certain acceleration and jerk bounds a point mass is an accurate enough representation. Similarly, the path of the lane change maneuver

must be designed such that it is feasible for a road vehicle. It is then assumed the low-level controllers can follow this path.

New Vehicle Controller Design

The task of vehicle n is to align itself with the gap in the platoon. Furthermore, it needs to drive using a traditional CACC algorithm once it starts its lateral maneuver to ensure sufficient inter-vehicle space. To achieve this an appropriate combination of location, velocity, and acceleration are the terminal conditions of its trajectory. The path of vehicle n is divided into three parts. First, the center of the on-ramp lane is followed up to the point of the lane change. Then a transitional path is designed that connects the center of the on-ramp lane to the main lane. The length of the transitional path is such that it spans a minimum time period. This time period is set to an appropriate length. For example, the average time of a human lane change maneuver. This length ensures that any issues regarding vehicle dynamical constraints are avoided, given that the path is continuous to a sufficient order. After the lane change, the path is defined as the center of the main lane. It is assumed that a low-level lateral controller follows the prescribed path. The lateral controller required to achieve this is not discussed in this paper. This section focuses on the longitudinal control strategy. Furthermore, the design of the lane change path is briefly discussed at the end.

The longitudinal control of vehicle n is split into two phases. First, an individual controller aims to align with the position behind vehicle p at a certain time. To complete the alignment the velocity of vehicle n must equal that of vehicle p. It should be noted that, since vehicle p is part of the platoon it may be required to change velocity when reacting to disturbances from the platoon leader. The second phase is a CACC controller behind vehicle p. Due to safety concerns, the second phase must be started before or at the start of the lane change. This section will focus on the individual controller. The switching of the control strategy between these two phases is an important topic that is discussed separately in Section 4.3.4.

The individual controller aims to have vehicle n arrive at the desired position (q_{lc}) and time (t_{lc}) at which the designed lane change starts. Position q_{lc} and time t_{lc} are calculated simultaneously using q_{mp} and the states of vehicle p. The value of t_{lc} is chosen such that a reasonable lane change maneuver can be performed before vehicle n reaches q_{mp}. This value is determined using the ETA at q_{mp} (t_{mp}) and a predefined time for the lateral movement ($\delta_{t,lc}$). Since the CACC controller is employed after the lane change maneuver, time t_{mp} is determined using the trajectory of vehicle p. In essence, t_{mp} is the time at which vehicle p reaches a position $q_{mp,p}$, which is defined as the position vehicle p would have during steady-state CACC driving when vehicle n is at q_{mp}. In other words,

$$q_{mp,p} = q_{mp} + L_n + r_n + h_n v_p. \tag{4.15}$$

Here L_n, r_n, and h_n are the length, standstill distance, and headway time of vehicle n. Furthermore, v_p denotes the velocity of vehicle p and is the assumed velocity of the platoon after the merge. The time instances t_{mp} and t_{lc} are now calculated using

$$t_{mp} = t + \frac{q_{mp,p} - q_p}{v_p} \tag{4.16}$$

$$t_{lc} = t_{mp} - \delta_{t,lc} \tag{4.17}$$

where t is the current time. It is assumed that vehicle n drives at velocity v_p from position q_{lc} onwards. Using the previous definitions, it can be noted that $q_{lc} = q_{mp} - \delta_{t,lc} v_p$.

The trajectory of vehicle n is determined by minimizing the cost function

$$J = \frac{1}{2} \int_0^{t_{lc}} \left(s^2(t)\right) dt. \tag{4.18}$$

Where $s(t)$ denotes the snap of the vehicle, which is the second derivative of acceleration. As shown in Ntousakis et al. (2016) the resulting trajectory is

$$q_i^*(t) = \frac{c_1 t^7}{7!} + \frac{c_2 t^6}{6!} + \frac{c_3 t^5}{5!} + \frac{c_4 t^4}{4!} + \frac{c_5 t^3}{3!} + \frac{c_6 t^2}{2} + c_7 t + c_8 \tag{4.19}$$

$$v_i^*(t) = \frac{c_1 t^6}{6!} + \frac{c_2 t^5}{5!} + \frac{c_3 t^4}{4!} + \frac{c_4 t^3}{3!} + \frac{c_5 t^2}{2} + c_6 t + c_7 \tag{4.20}$$

$$a_i^*(t) = \frac{c_1 t^5}{5!} + \frac{c_2 t^4}{4!} + \frac{c_3 t^3}{3!} + \frac{c_4 t^2}{2} + c_5 t + c_6 \tag{4.21}$$

$$j_i^*(t) = \frac{c_1 t^4}{4!} + \frac{c_2 t^3}{3!} + \frac{c_3 t^2}{2} + c_4 t + c_5. \tag{4.22}$$

Where $j(t)$ denotes the jerk profile. Coefficients c_1 to c_8 can be chosen such that the trajectory satisfies initial and terminal conditions on the position, velocity, acceleration, and jerk. This design freedom and the relative simplicity of the equations are the main reasons for this trajectory choice.

Using (4.4), control input $u_i = \tau j_i^* + a_i$ is shown to make the vehicle follow the desired jerk profile. To account for disturbances an MPC-type controller is used. In essence, the trajectories (4.19)-(4.22) are periodically determined based on the current measurements. At any instance at which the trajectories are determined, disturbances encountered during and after the previous instance are accounted for. Over time, the vehicle approximates the terminal conditions in the presence of disturbances. The inputs of this controller are the current and desired terminal position, velocity, acceleration, and jerk. While position, velocity, and acceleration are measurable, jerk is not. The current jerk is therefore computed using (4.4). Since τ, a_i, and u_i are assumed to be known, the jerk can be estimated at any time instance. It should be noted that noise from the acceleration measurement is present in this estimation.

The desired terminal states are $q_n(t_{lc}) = q_{lc}$, $v_n(t_{lc}) = v_p$, $a_n(t_{lc}) = 0$ and $j_n(t_{lc}) = 0$. In other words, vehicle n aims to drive with velocity v_p at position q_{lc} and time t_{lc}. Furthermore, the vehicle aims to end in a steady-state CACC platoon. Therefore, the desired acceleration and jerk are zero. It can be noted that the terminal states are dependent on v_p, this includes the determination of t_{lc}. The usage of an MPC-based controller aims to account for variations in v_p. Using (4.19)-(4.22), the optimal trajectory can quickly be calculated. Analytical expressions for coefficients c_1 to c_8 considering the initial and terminal constraints can be expressed offline. Then a numeric value can quickly be computed online by filling in the expression. The proposed MPC approach is thus computationally inexpensive and applicable in real-world environments.

To describe the lateral lane change trajectory the one-dimensional q coordinate is replaced by a two-dimensional x and y coordinate system. Where the x-axis is parallel, and the y-axis is perpendicular to the main lane. Now the y coordinates of the lane change trajectory y_{lc} during the lateral movement are expressed as a fifth-order polynomial function of x such that

$$y_{lc}(x) = c_1 + c_2 x + c_3 x^2 + c_4 x^3 + c_5 x^4 + c_6 x^5. \tag{4.23}$$

A fifth-order polynomial is chosen because its coefficients c_1 up to c_6 can be chosen such that initial and final conditions on the position, velocity, and acceleration y-direction are satisfied. For this reason, this type of trajectory has previously been used in literature for the modeling and control of lane change maneuvers (Krajewski et al., 2018; Papadimitriou and Tomizuka, 2003; Venkita et al., 2020). It is assumed that a low-level lateral controller follows the prescribed path (4.23).

The length of the lane change path in the x-direction is used to compute the entire path. The length in the x-direction is based on an average time for lane change maneuvers (T_{lc}) and the velocity of the platoon. In essence,

$$x_{lc} = x_{mp} - v_p T_{lc} \tag{4.24}$$

where x_{lc} and x_{mp} are the x-position of the start of the lane change and the merging point respectively. The arc length of the path is longer than the length in the x-direction. Consequently, the time of the maneuver ($\delta_{t,lc}$) is longer than T_{lc}. It should be noted that selecting an appropriate value for T_{lc} will ensure low lateral excitations. This helps meet the road boundary conditions since vehicle n can reach the main lane at x_{mp}.

To make the transformation from the two-dimensional space to the one-dimensional q coordinate, the arc length of the lane change trajectory (L_{lc}) is required. However, this problem is too complex to solve analytically. Therefore, a numerical measurement of the path is used to determine L_{lc}. This length is used to determine δ_q as

$$\delta_q(x) = L_{lc}(x) - (x_{mp} - x) \quad \forall\, x_{lc} \leq x \leq x_{mp}. \tag{4.25}$$

Where $L_{lc}(x)$ denotes the length of the path from position x to x_{mp}. If vehicle n has not yet reached the lane change, $\delta_q(x_{lc})$ describes the additional length of the entire maneuver. Furthermore, it can be noted that $q_{lc} = q_{mp} - L_{lc}(x_{lc})$ and $\delta_{t,lc} = L_{lc}(x_{lc})/v_p$.

Follower Vehicle Controller Design

Vehicle f is controlled using the variable gap controller of Section 4.2.2. The main aim is to have the gap opened at time t_{lc}. To achieve this, it is essential to define an appropriate γ-trajectory. Similar to the controller of vehicle n, the control strategy must be able to handle disturbances from vehicle p.

The initial conditions of the γ-trajectory at t_0 are chosen as arbitrary values such that

$$\begin{bmatrix}\gamma(t_0) & \dot{\gamma}(t_0) & \ddot{\gamma}(t_0) & \dddot{\gamma}(t_0)\end{bmatrix}^\top = \begin{bmatrix}\gamma_0 & \dot{\gamma}_0 & \ddot{\gamma}_0 & \dddot{\gamma}_0\end{bmatrix}^\top. \tag{4.26}$$

This allows the trajectory to be redefined at some time when γ and its derivatives are non-zero while C^2 continuity is maintained. Therefore, the possibility to update the trajectory in an MPC-type algorithm is available.

The terminal conditions of the trajectory at time t_{lc} are

$$\begin{bmatrix}\gamma(t_{lc}) & \dot{\gamma}(t_{lc}) & \ddot{\gamma}(t_{lc}) & \dddot{\gamma}(t_{lc})\end{bmatrix}^\top = \begin{bmatrix}\gamma_{lc} & 0 & 0 & 0\end{bmatrix}^\top. \tag{4.27}$$

Where

$$\gamma_{lc} = v_p(t_{lc})h_n + L_n + r_n \tag{4.28}$$

is a sufficiently large gap to accommodate vehicle n. Here the assumption is made that $v_n(t_{lc}) \approx v_p(t_{lc})$ because this is the desired terminal state of vehicle n. It should be noted that the gap opening maneuver only has terminal conditions on γ and its derivatives and not on the states of vehicle f. The reason for this is that the error can be assumed to remain small, because of the stabilizing controller and the small initial error. It can therefore be assumed that he terminal states of vehicle f are as prescribed if appropriate values for γ and its derivatives are chosen.

To satisfy the four initial and four terminal conditions a seventh-order polynomial is used. This polynomial is similar to the optimal trajectory in (4.19)-(4.22). Here γ replaces position q^* and its derivatives replace v^*, a^*, and j^*. At every time instance, the γ-trajectory is adjusted using the current values of γ and its derivatives to the desired terminal conditions of (4.27). This allows for the new vehicles to react to disturbances from vehicle p during the maneuver by altering the values of t_{lc} and $v_p(t_{lc})$.

Using this approach, the gap opening procedure is executed in a timely fashion. Due to the desired terminal conditions, the switch to a CACC controller targeting vehicle n at time t_{lc} should be relatively smooth. However, a more robust method of controller switching is proposed in the next section.

4.3.4 Controller Switching

One important aspect of the control strategy is the switching of the controllers in the vehicles. Most notably, vehicle n switches from an individual controller to a CACC controller behind vehicle p. Furthermore, vehicle f switches from a variable gap CACC controller behind vehicle p to a conventional CACC controller behind vehicle n. Both controller transitions are executed in a similar fashion.

The main challenge for the controller transition is the introduction of the CACC controller. In practice, it is near impossible to have the error and its first and second derivative zero. Based on the error dynamics, there will thus be some initial errors that require time to converge to zero. To avoid unsafe situations, these error dynamics will be managed.

The proposed solution is to introduce the γ-factor in the new CACC controller. Initial values for γ and its first and second derivatives are chosen such that the perceived initial error states are zero. Here, the perceived error states are those calculated using the local measurements and received communication. It should be noted that the perceived error is not identical to the actual error due to the measurement noise but it is a close approximation. Next, a suitable γ-trajectory is chosen such that the vehicle obtains the desired terminal states. Finally, the switch to a regular CACC controller can be completed.

The switching logic of both vehicles is similar and briefly explained in Algorithm 4.1. The initial controller for vehicle n is the individual MPC controller and for vehicle f is the gap opening CACC controller. The expected time of starting time of the lane change (t_{lc}) is computed to help determine the minimum and maximum time length of the transition. Then the initial ($t_{0,i}$) and the final ($t_{s,i}$) time of the transition can be computed. If there is a value for $t_{s,i}$ for which the expected states satisfy the proposed conditions, the transitional controller is started. When $t_{s,i}$ is reached the vehicle switched to a conventional CACC controller.

A more detailed description of the transitional controller of vehicles is discussed separately in the coming subsections. The transition of vehicle n is introduced first. Secondly, the switching strategy for vehicle f is discussed.

New Vehicle Controller Switch

A transition phase is introduced to switch the controller of vehicle n from the individual MPC controller to a CACC controller. In the transition phase, a CACC controller with a γ-factor is used. This allows the vehicle to transition from an initial state to the desired state in a predefined time.

Using the variable gap CACC controller a trajectory of γ is designed such that the desired trajectories (4.19)-(4.22) are followed. Since the variable gap CACC controller is stable, and the errors are initialized to be small, the errors

Algorithm 4.1: General overview of the controller switching logic

```
while Initial controller do
    Compute γ(t0), γ̇(t0), γ̈(t0), and γ⃛(t0) using current states;
    Compute the estimated trajectory of the new CACC target vehicle;
    Compute t_lc and the minimum and maximum switching times, t_i,min
     and t_i,max;
    Compute a set of possible values of t_s,i that satisfies
     t_i,min ≤ t_s,i − t_0,i ≤ t_i,max;
    Compute multiple expected trajectories for switching using a
     discretization of the set of t_s,i values;
    if ∃ a trajectory that satisfies bounds on the expected states and t_s,i
     then
        Start transition controller with smallest possible t_s,i;
    else
        if t_i,min + t_0,i ≤ t_lc then
            Start transition controller with t_s,i = t_lc
```

$e_{1,n}$, $e_{2,n}$, and $e_{3,n}$ can be assumed to be zero. Using (4.8)-(4.10) this yields

$$v_n^* = \frac{1}{h}\left(q_p - q_n^* - L_n - r_n - \gamma\right) \tag{4.29}$$

$$a_n^* = \frac{1}{h}\left(v_p - v_n^* - \dot{\gamma}\right) \tag{4.30}$$

$$j_n^* = \frac{1}{h}\left(a_p - a_n^* - \ddot{\gamma}\right). \tag{4.31}$$

Now, the γ-trajectory is obtained using differential equation

$$j_n^* = \frac{a_p - \ddot{\gamma}}{h} - \frac{v_p - \dot{\gamma}}{h^2} + \frac{q_p - q_n^* - L_n - r_n - \gamma}{h^3}. \tag{4.32}$$

It should be noted that this differential equation provides a solution for γ, $\dot{\gamma}$, and $\ddot{\gamma}$. Thus, a trajectory can be calculated from any initial values of those variables. This method poses two challenges. First, the optimal trajectories of q_n^* and j_n^* must be determined using the previously mentioned differential equation. This can be done analytically offline and will thus not pose any excessive computational load or other problems during the maneuver. Secondly, a prediction of q_p, v_p, and a_p is required. Since the strategy assumes a decentralized control architecture, these trajectories cannot be prescribed. The prediction will therefore be computed using the available information.

The optimal trajectories require initial and terminal conditions to be computed. The initial position, velocity, acceleration, and jerk can easily be measured or computed using a similar method as the individual MPC controller. The

terminal conditions are such that the desired distance to vehicle p is achieved at t_{lc} or earlier. The terminal velocity, acceleration, and jerk are set to match those of vehicle p. After the transition, the vehicle proceeds using the conventional CACC control law.

A prediction of the future behavior of vehicle p is required to solve the optimal trajectories. Since this vehicle is part of the platoon, it may react to a preceding vehicle. The future control inputs u_p are therefore unknown. In the prediction, it is assumed that u_p is zero for any future time instances. Now, the states of the preceding vehicle are defined as $x_p = \begin{bmatrix} q_p & v_p & a_p \end{bmatrix}^\top$. Using (4.1), (4.3), and (4.4) the dynamics can be written as $\dot{x}_p = A_p x_p$ where

$$A_p = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \frac{-1}{\tau_p} \end{bmatrix}. \tag{4.33}$$

Consequently, at time t_0 the states x_p at some future time instance $t \geq t_0$ can be predicted using

$$x_p(t) = e^{A_p(t-t_0)} x_p(t_0). \tag{4.34}$$

This results in the equations

$$a_p(t) = a_p(t_0) e^{-\frac{t-t_0}{\tau_p}} \tag{4.35}$$

$$v_p(t) = v_p(t_0) + a_p(t_0) \left(\tau_p - \tau_p e^{-\frac{t-t_0}{\tau_p}} \right) \tag{4.36}$$

$$q_p(t) = q_p(t_0) + v_p(t_0)\,(t-t_0) + a_p(t_0) \left(\tau_p^2 e^{-\frac{t-t_0}{\tau_p}} + \tau_p\,(t-t_0) - \tau_p^2 \right). \tag{4.37}$$

It should be noted that (4.36), (4.37), and (4.15) can be used to determine t_{mp}. However, in practice, this method is relatively sensitive to noise. Therefore, this method is not used and t_{mp} is calculated using (4.16).

To follow j_n^* the errors at initial time $t_{0,n}$ must be zero. As a result, unlike the variable gap controller described in Section 4.3.3, the initial values of γ and its derivatives, therefore, cannot be chosen arbitrarily. Appropriate values of γ and its derivatives are chosen such that the perceived errors are equal to zero at time $t_{0,n}$. Using (4.8)-(4.10), the computed initial values are

$$\gamma(t_{0,n}) = q_p(t_{0,n}) - q_n(t_{0,n}) - r_n - L_n - h_n v_n(t_{0,n}) \tag{4.38}$$

$$\dot{\gamma}(t_{0,n}) = v_p(t_{0,n}) - v_n(t_{0,n}) - h_n a_n(t_{0,n}) \tag{4.39}$$

$$\ddot{\gamma}(t_{0,n}) = a_p(t_{0,n}) - a_n(t_{0,n}) \left(1 - \frac{h_n}{\tau_n} \right) - \frac{h_n}{\tau_n} u_n(t_{0,n}). \tag{4.40}$$

The terminal values of γ and its derivatives at the end time of the switch ($t_{s,n}$) can be chosen arbitrarily. Their values are chosen such that the vehicle can

easily transition to a regular CACC controller. The terminal velocity of vehicle n is thus equal to that of vehicle p while the position and acceleration are such that the errors of the CACC controller are zero. The mimicking of the velocity is required since the desired inter-vehicle distance is velocity-dependent. If the position error is zero while the vehicle velocity is not matched, vehicle n will thus not be driving where vehicle f expects it. For a selected switching time $t_{s,n}$, there is now a unique solution for the coefficients in (4.19) and the trajectory of γ.

Using this control strategy, the predicted behavior of the vehicle can be investigated. Then an appropriate length of the controller transition and thus switching time $t_{s,n}$ can be selected. The desired length of the transition is defined as the shortest time for which certain conditions are fulfilled. First, conditions on the predicted velocity, acceleration, and jerk of the vehicle are set such that

$$v_{n,min} \leq v_n^*(t) \leq v_{n,max} \qquad \forall\, t_{0,n} \leq t \leq t_{s,n} \tag{4.41}$$

$$a_{n,min} \leq a_n^*(t) \leq a_{n,max} \qquad \forall\, t_{0,n} \leq t \leq t_{s,n} \tag{4.42}$$

$$j_{n,min} \leq j_n^*(t) \leq j_{n,max} \qquad \forall\, t_{0,n} \leq t \leq t_{s,n}. \tag{4.43}$$

This ensures a vehicle trajectory that is legal, feasible, and comfortable. It should be noted that these conditions must be chosen conservatively. The conditions may not be met during the maneuver due to the uncertainty of the trajectory of vehicle p. Secondly, a condition on the minimum and maximum value of γ is introduced. Such that

$$\gamma_{n,min} \leq \gamma(t) \leq \gamma_{n,max} \qquad \forall\, t_{\gamma,0} \leq t \leq t_{s,n}. \tag{4.44}$$

Here $t_{\gamma,0}$ is the first time instance at which γ is within thresholds $\gamma_{n,min}$ and $\gamma_{n,max}$. The γ condition is to ensure that when vehicle n is close to the desired position it will not deviate far from it. Big deviations will make it difficult for vehicle f to switch its controller. Time instance $t_{\gamma,0}$ is used to ensure the switching can be started while $\gamma(t_{0,n})$ is large. Lastly, the length of the transition is limited using the conditions

$$t_{n,min} \leq t_{s,n} - t_{0,n} \leq t_{n,max} \quad \& \quad t_{s,n} \leq t_{lc}. \tag{4.45}$$

Upper limit $t_{n,max}$ prevents undesirable long transitions. The lower limit $t_{n,min}$ bounds the space in which γ trajectories are examined. Furthermore, it limits the start of the transition in absolute time. In essence, the transition must be finished before t_{lc} to ensure a proper alignment before the lateral maneuver. Therefore, the switching maneuver must be initiated at time $t_{lc} - t_{n,min}$ at the latest. The limit in absolute time is the most important criterion and thus a switch is initiated at this time. If the trajectory is infeasible the generated errors will be handled by the CACC controller after the transition.

One of the main vulnerabilities of this strategy lies in (4.34). This equation requires a knowledge of a_p which is not directly measurable. Currently, the

transmitted desired acceleration u_p is used instead of a_p in the simulations. For more accuracy, an estimator could be designed. However, the sensitivity to errors in the estimated a_p is investigated in Appendix A.2 and found to be relatively small.

Following Vehicle Controller Switch

The transition algorithm for vehicle f is similar to that of vehicle n. The main difference is that vehicle f ends up driving behind vehicle n. Since vehicle n may still be maneuvering during the switch, the assumption that its control input u_n is zero cannot be made. Instead, when applicable, information transmitted by vehicle n is used to determine the switching strategy. When vehicle n has successfully merged and utilizes a conventional CACC, the assumption u_n is zero is used.

The control strategies of vehicle n aim to follow the optimal trajectory as described in (4.19). This knowledge is used by vehicle f to design its γ-trajectory. Vehicle n transmits a vector C that contains the coefficients c_1 to c_8 of q_n^*. Regarding the transmission, it is important to note the communication delay. The time of transmission is added to the message such that the communication delay can be determined and the current values of the coefficient can be computed by shifting the polynomial in time. It is assumed that both vehicles have access to the same global time, which can be obtained through the Global Navigation Satellite System (GNSS) sensors. These coefficients are used to predict q_n, v_n, and a_n. Now differential equation

$$j_f^* = \frac{a_n - \ddot{\gamma}}{h} - \frac{v_n - \dot{\gamma}}{h^2} + \frac{q_n - q_f^* - L_f - r_f - \gamma}{h^3}, \tag{4.46}$$

is used to determine the γ-trajectory. This ensures that the trajectory of (4.19) is followed and the predicted behavior of vehicle f can be investigated. The equivalent conditions of (4.41)-(4.44) for vehicle f are used.

It is important to note that the coefficients are only valid till the time that vehicle n intends to complete its switch. To account for this, the corresponding condition to (4.45) is defined as

$$\begin{cases} t_{f,min} \leq t_{s,f} \leq \min\left(t_{s,n}, t_{f,max}\right), & \text{if } \exists t_{s,n} \\ t_{f,min} \leq t_{s,f} \leq \min\left(t_{lc}, t_{f,max}\right), & \text{if } \nexists t_{s,n} \;\&\; t < t_{lc} \\ t_{f,min} \leq t_{s,f} - t_{0,f} \leq t_{f,max} & \text{otherwise.} \end{cases} \tag{4.47}$$

Furthermore, it should be noted that the predicted trajectory of vehicle n may change unexpectedly. For example, when it is switching from its individual controller to the transitional controller. This can cause unpredicted behavior of vehicle *f*. Its γ-trajectory will therefore be updated when the communicated terminal time of the validity of C changes more than a predefined bound.

4.3.5 Collision Avoidance

Vehicles n and f are required to switch their control strategy throughout this maneuver. For vehicle f, the avoidance of collisions with other vehicles is a challenge that requires special attention. This is especially important when vehicle f has its CACC controller aimed at vehicle n while this vehicle has not yet made the lane change. In this situation, vehicle n can drive past vehicle p. However, if vehicle f follows vehicle n, it will collide with vehicle p since both are in the same lane. This problem is inherent to the on-ramp merging scenario because vehicle n can be initialized ahead of the platoon. Methods restricting vehicle n from overtaking vehicle p are therefore undesirable and unachievable.

To achieve safe behavior of vehicle f, it needs to track multiple vehicles. This challenge was investigated in Semsar-Kazerooni et al. (2017). This work is based on a CACC controller that utilizes artificial potential fields. Meaning the preceding vehicle has a repulsive potential or an attractive potential when the trailing vehicle is too close or too far respectively. The repulsive potential causes the trailing vehicle to slow down and the attractive potential causes it to accelerate. When the trailing vehicle is following a vehicle on another lane, the repulsive potential of the preceding vehicle in its own lane is included in the control strategy. When the trailing vehicle is far away from the preceding vehicle, there is no contribution since the repulsive potential is zero. However, when the trailing vehicle is getting too close, the repulsive potential causes it to slow down. The CACC control law of (4.14) does not have specific repulsive and attractive components. Therefore, the work of Semsar-Kazerooni et al. (2017) cannot directly be applied.

An approach for a CACC controller similar to the one used in this paper is shown in the work of Hult et al. (2018). This concept also considers vehicle p only when it is too close. This was done by using a min-function on the position error. Assuming vehicle f's position error throughout the maneuver remains close to zero; it will adjust its control input when its distance to vehicle p is too small.

The proposed solution in this work aims to have vehicle f react when it is approaching vehicle n too rapidly. This is done by having vehicle f calculate the CACC control law without γ-factor aimed at vehicle p in the background. This results in a virtual control input $u_{f,p}$. This control input is compared to that based on vehicle n ($u_{f,n}$). Then, the control input u_f is defined as the minimum of the two, such that

$$u_f = \min(u_{f,p}, u_{f,n}). \tag{4.48}$$

If the merging maneuver is executed as expected, there will be a large distance between vehicles f and p during and after the transition. Thus, $u_{f,p} > u_{f,n}$ and $u_f = u_{f,n}$. If vehicle f is approaching vehicle p too rapidly, control input $u_{f,p}$ will become small and slow down vehicle f.

Table 4.1: An overview of the minimum information that needs be broadcasted by each vehicle.

Function	Vehicle p	Vehicle n
CACC	u_p	u_n
Alignment	q_p, v_p	L_n, r_n, h_n, $\delta_{t,lc}$
Controller transition		δ_q, C

4.3.6 Communication Strategy

To achieve a decentralized control scheme, inter-vehicle communication and locally available information are investigated. Some of the local information can be measured with the on-board sensors, other information should be stored in the memory of the vehicles. It is assumed all vehicles have an accurate map of the on-ramp environment. This map is used to translate two-dimensional coordinates into a one-dimensional q coordinate. Moreover, important locations, such as q_{mp}, are indicated on this map.

Generally, the communication in this type of system is based on broadcasting messages (Hult et al., 2018). Thus, the required outgoing messages of each vehicle are investigated. An overview is shown in Table 4.1. The table shows the function for which certain information is required. A more thorough analysis of the functions is given below.

For regular CACC driving, every vehicle requires control input u of its preceding vehicle. Additionally, vehicles typically transmit a Cooperative Awareness Message (CAM) that contains its position and velocity amongst other things.

During the alignment phase of the merging maneuver, not all vehicles may be able to measure each other's location yet. However, vehicle n bases its trajectory on v_p and q_p as shown in (4.16). The effect of the communication delay on the computation of t_{mp} is limited if an accurate time stamp is sent with v_p and q_p. Vehicle f can measure v_p and q_p but requires communicated data from vehicle n. First, it requires in L_n, r_n, and h_n in (4.28). Furthermore, $\delta_{t,lc}$ is required to compute t_{lc}. Alternatively, vehicle n can send its predicted t_{lc} such that vehicle f does not have to compute it.

At the start of the controller transition, it is assumed all vehicles can see each other. The measurements are performed in a two-dimensional frame. Therefore, vehicle n is required to send δ_q based on its planned trajectory. This will allow all vehicles to make the two-dimensional to one-dimensional transformation. Lastly, coefficient vector C is transmitted by vehicle n.

It should be noted that in this strategy vehicle p does not have and transmit knowledge of its future trajectory. Admittedly, if there are multiple merging vehicles, vehicle p may have severe longitudinal excitations. The performance in those scenarios may be improved if additional information is sent from vehicle p and incorporated in the control of vehicles n and f. This is outside the scope of

the current work and may be investigated in future research.

4.4 Simulation and Analysis of the Control Strategy

In this section, the control strategy is demonstrated and analyzed using simulations. First, the simulation environment and specifications are discussed. This includes the parameter tuning and the noise profile. Then, a comparison between the proposed strategy and a conventional MPC-based approach is presented. This is followed by simulation results regarding the general performance of the proposed strategy. Finally, noise sensitivity is addressed by analyzing multiple simulations with different noise signals.

4.4.1 Simulation Environment and Specifications

The simulations are performed using MATLAB/Simulink R2020a. A fixed-step solver, running at 100 Hz, is used to perform the simulations in Simulink. The model is written as a continuous-time model and the *Automatic solver selection* function is used. The vehicles are modeled as individual sub-models such that time delays and noise can easily be added to information flowing between them. The controllers running on the vehicles were written in MATLAB and added as a user-defined function. The vehicles are simulated as point masses using the vehicle dynamics of (4.1)-(4.4). Constraints in the vehicle dynamics are not directly modeled but considered when analyzing the results. The reasoning behind this decision is that the planned trajectory and controller design should stay clear of these constraints. This is best investigated if the possibility to cross these constraints exists in the model.

In the simulation set-up, a platoon of three vehicles is considered. The new vehicle must merge between the second and third. The first vehicle is simulated to create some dynamics in the behavior of the second. The new vehicle is initialized at a position such that it can easily reach the desired position. All vehicles are 5 meters long, their driveline dynamics are simulated using $\tau = 0.1$ s. Furthermore, the CACC controllers are tuned using the values $h = 0.5$ s, $r = 2$ m, $k_p = 0.2$ and $k_d = 0.7$ as obtained from Ploeg et al. (2011).

The platoon leader is driving at a constant velocity of 100 km/h. Since a steady-state platoon is initialized, this is the initial velocity of all platoon vehicles. Vehicle p is initially 500 meters away from the merging point. Vehicle n is initialized at 55 km/h and has an acceleration of 1 m/s^2 since it is trying to reach the highway velocity. The initial position of vehicle n is 50 meters ahead of vehicle p. This position is chosen such that it easily reaches the desired position in the platoon. This initialization is similar to that found in literature on highway on-ramp merging (Cao et al., 2015; Kesting et al., 2007). The effect of changing the initial position is analyzed in Appendix A.1. Furthermore, in

Chapter 5 the merging sequencing algorithm is analyzed and a wider range of initial conditions is used.

Reasonable values for the proposed merging control strategy were selected. First, the approximated duration of the lane change T_{lc} is 5 seconds. This is similar to the behavior of a human driver (Thiemann et al., 2008; Toledo and Zohar, 2007). Next, $t_{i,min} = 2$ s and $t_{i,max} = 5$ s is considered for the controller transitions, and the γ-trajectory is constrained using $a_{i,max} = -a_{i,min} = 1.2$ m/s^2, $j_{i,max} = -j_{i,min} = 0.8$ m/s^3 and $\gamma_{i,min} = -0.1$ m $\forall i \in \{n, f\}$. It should be noted that not all available conditions were used. This tuning could be altered when more specific behavior of the vehicles is desired.

To approximate some of the challenges encountered in a real-world environment, sensor noise was added using Gaussian noise profiles. The radar noise has a standard deviation of 20.9 cm and 0.141 m/s for position and velocity measurements respectively. The noise of the on-board sensors has a standard deviation of 0.048 m/s and 0.20 m/s^2 for the ego velocity and acceleration respectively. These values are based on experiments with a demonstrator platform (Schinkel et al., 2021). It should be noted that this noise is only applied to the measured values. For example, in (4.4) of the vehicle model, the real acceleration without noise is used. However, the noise is applied when the error is computed in the vehicle using for example (4.9), which then affects the control input. The magnitudes of these errors are based on an experimental setup. Furthermore, a time delay of 0.02 seconds was added to all communicated messages (Hoogeboom, 2020).

This section will continue with a comparison between the proposed strategy and a conventional MPC-based strategy. Then two additional scenarios are investigated to assess the performance of the proposed control strategy. In these scenarios, the leading vehicle either accelerates or decelerates during the maneuver. These scenarios are added to investigate the effect of the preceding vehicle reacting to environmental inputs. Then a more severe deceleration scenario is investigated to show the importance and performance of the collision avoidance controller. Lastly, the initial scenario in which the platoon leader has a constant velocity is repeated 100 times with different noise signals using the previously mentioned standard deviation values. The results are used to analyze the noise sensitivity.

4.4.2 Control Strategy Comparison

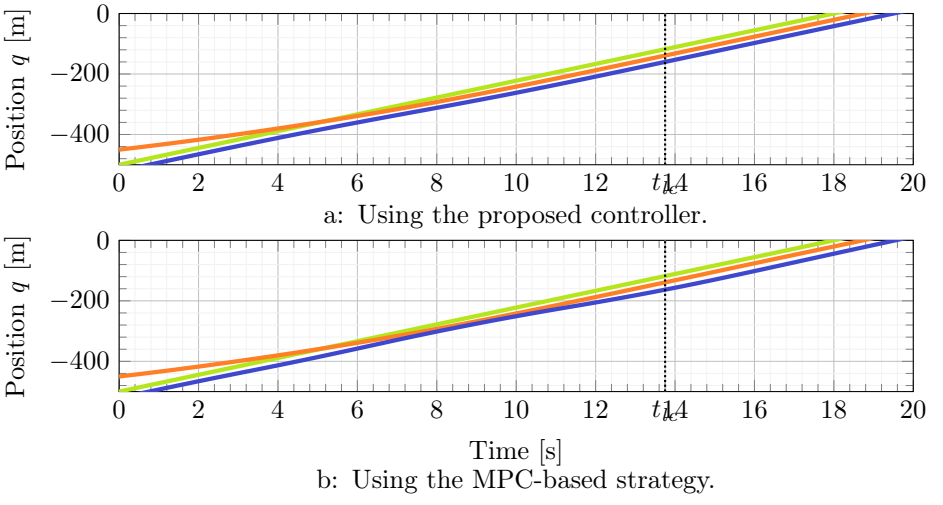
This section focuses on the comparison of the proposed strategy to a conventional MPC-based strategy. This controller is similar to that proposed in other literature (Ntousakis et al., 2016; Rios-Torres and Malikopoulos, 2017b). In essence, the MPC-based strategy is a version of the proposed method that does not use the transitional controller but iteratively calculates its trajectory until the lane change is started. The tuning of this MPC-based strategy is equal to that of

the MPC-based phase in the proposed strategy. As explained in Section 4.3.3, at every time step a trajectory based on equations (4.19)-(4.22) is planned that fulfills the initial and final conditions. This is done by calculating the values for constants c_1 to c_8. These constants can be expressed as a function of the current and desired terminal states. During real-time operation, the values can thus quickly be computed. The desired final states are those required at t_{lc}. Therefore, the planning horizon spans from the current time to t_{lc} and thus varies throughout the maneuver. Since the MPC-based strategy does not use a transitional controller, the MPC controller is thus used until t_{lc} when the control strategy is directly switched to the conventional CACC controller. For vehicle f the variable gap CACC controllers for gap opening and controller transition are replaced with the same MPC controller as vehicle n. There is a difference in the terminal conditions of the trajectory such that all vehicles are appropriately spaced at the end of the platoon. An assessment regarding the general performance of the proposed method is given in Section 4.4.3.

One important challenge implementing this MPC-based strategy is handling the small planning horizons when t_{lc} is approached. In literature, this problem is not always elaborated upon but problems with a small planning horizon were encountered in Ntousakis et al. (2016). There, the problem was solved by running an ACC controller in the background and using the most restrictive actuator command. This method is based on the ACC controller used after the merge. A comparable method for the CACC controller is hard to compute due to the integrator term in (4.14). The desired control input is thus not directly computed. To avoid extreme excitations, a saturation of ± 1.5 m/s^2 was added on control input u_i during the MPC controller. Using (4.4) of the vehicle model it is shown that this limits the acceleration of the vehicle. Furthermore, it can be shown that this saturation limits the jerk to ± 30 m/s^3. The bound for this saturation is tunable, a higher bound may result in higher excitations and a more restrictive bound may prevent reaching the desired terminal states. For demonstration purposes, a bound of ± 1.5 m/s^2 is deemed suitable.

Figure 4.6 shows the position of the vehicles over time. In this simulation, q_{mp} is defined as 0. The start of the lane change (t_{lc}) is indicated on the x-axis. It is shown that using either strategy the vehicles align at t_{lc}. The main visible difference between the two strategies is the behavior of vehicle f before t_{lc}. It is shown that vehicle f drives more forward when using the MPC-based strategy. This is visible at for example 8 seconds into the simulation. It can thus be concluded that there is a difference in the behavior between the two strategies. Furthermore, after t_{lc}, vehicle f appears to be further back using the MPC-based approach. This indicates that the strategies affect the behavior of the platoon after t_{lc}. To obtain a better understanding of the strategies more information needs to be analyzed.

The difference between the two strategies becomes more apparent when the velocities and accelerations in Figure 4.7 are examined. It is shown that the



a: Using the proposed controller.

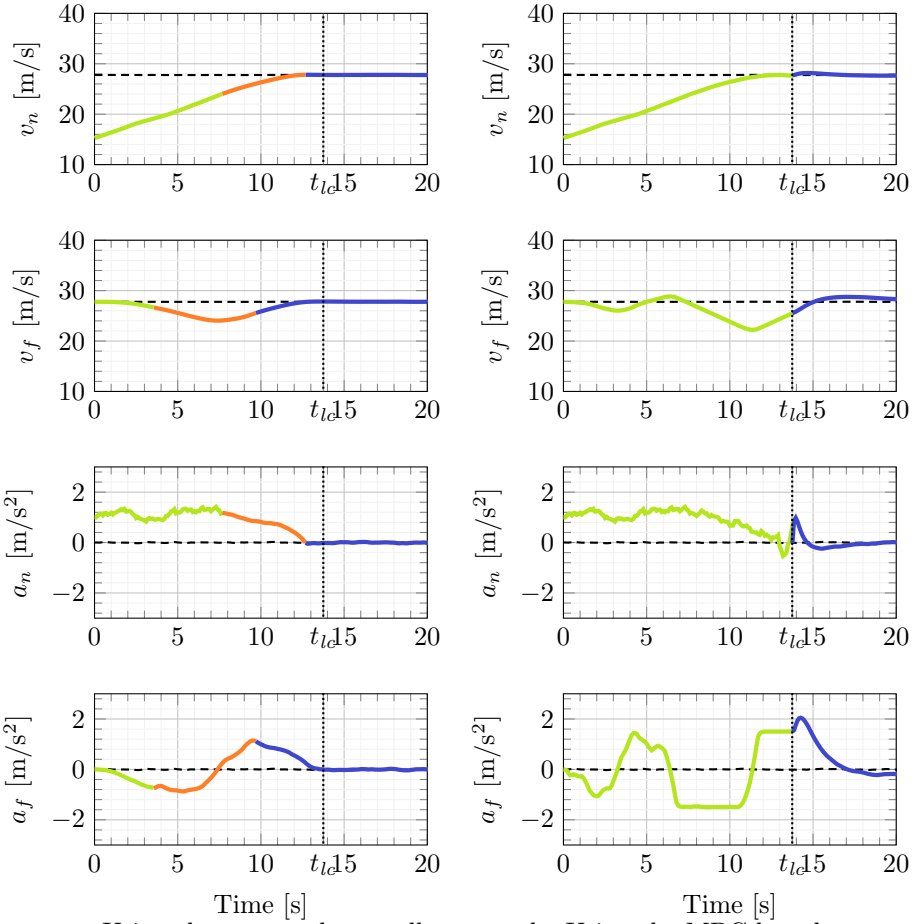
b: Using the MPC-based strategy.

Figure 4.6: The position of vehicle p (—), n (—), and f (—) during a merging maneuver.

4

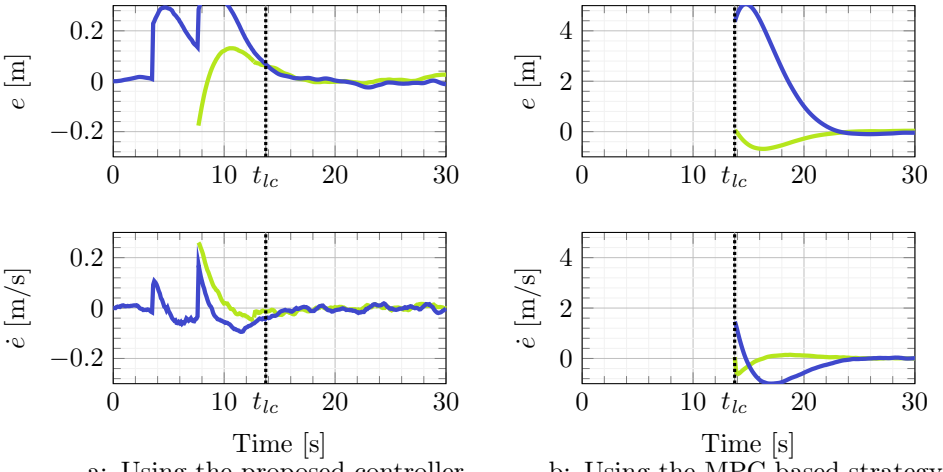
velocities and accelerations after t_{lc} remain close to constant for the proposed strategy. However, for the MPC-based strategy, excitations are found after t_{lc}. These excitations are caused by initial errors in the CACC controller at t_{lc}. This behavior is undesired since one of the requirements is to achieve a steady-state platoon at t_{lc}. Furthermore, it is shown that large excitations in the velocities and accelerations appear for the MPC-based strategy. This is especially a problem for vehicle f, the initial and final states are relatively close and only a gap needs to be created. The measurement noise can thus greatly influence the proposed trajectory. This causes the gap not to be adequately opened at the start. Then a more severe braking action is required at approximately 7 seconds into the simulation to open the gap. The saturation on u_i limits the acceleration of the vehicle. Then the vehicle accelerates to ensure the desired velocity is met at t_{lc}. However, the attempt is unsuccessful resulting in a too low velocity at t_{lc}. Afterwards, accelerations are required to match the velocity. For vehicle n, the trajectories appear reasonable and the bound on u_i is not reached. From these results, it can be concluded that the proposed strategy is preferable for the formation of a steady-state platoon.

The errors during the maneuver are shown in Figure 4.8. There is no error definition for the MPC-based controller, therefore no errors are plotted for times when an MPC-based controller is used. It should furthermore be noted that the scale is different for the two strategies. This is done because of the higher errors when the MPC-based strategy is used. Large errors occur after t_{lc}, which is especially problematic because the vehicles may be in the same lane. The



a: Using the proposed controller. b: Using the MPC-based strategy.

Figure 4.7: The velocity and acceleration of vehicles n and f; before (——), during (——), and after (——) the controller transition. The velocity and acceleration of the preceding vehicle is indicated (----).



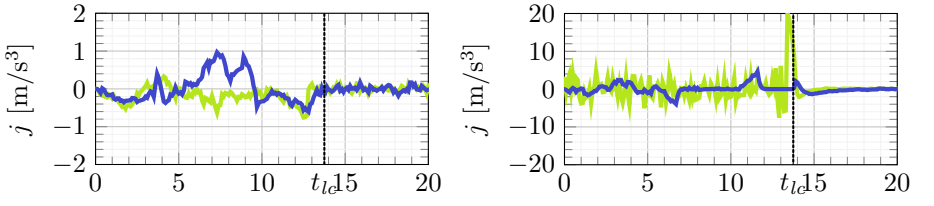
a: Using the proposed controller. b: Using the MPC-based strategy.

Figure 4.8: The error and its derivative for vehicle n (——) and f (——). It should be noted that the scale is different between the two strategies such that the details remain visible.

errors are partly caused by measurement noise and partly the saturations which are required to limit the excitations. For the proposed controller the errors remain within centimeter range and errors from the controller initializations have dampened out when t_{lc} is reached. This once more underlines the advantage of the proposed method.

To comment on the driver comfort, the jerk trajectories are examined in Figure 4.9. Once more, the scales are different for the two strategies because of the large differences between the two jerk profiles. In most literature, a longitudinal jerk of ± 3 m/s^3 is deemed acceptable (Hoberock, 1977; Shladover, 1991). The jerk trajectory using the MPC-based strategy reaches peaks of around 22 m/s^3 for vehicle n. This is near t_{lc} when the planning horizon is short. Similar behavior is found in Ntousakis et al. (2016). Vehicle f does not have high jerks at this part of the simulation because the saturation in u_i is reached resulting in a momentarily constant acceleration. However, a jerk of ± 4 m/s^3 is surpassed by vehicle f which is outside of the comfort bounds. The jerk of the proposed method stays within approximately ± 1 m/s^3 and thus well within the comfort bounds. Regarding passenger comfort, the proposed method is thus favorable.

A summary of the comparison is given in Table 4.2. The proposed strategy outperforms the MPC-based strategy for most performance indicators. This is caused by the lack of the transitional controller in the MPC-based strategy because that is its only difference from the proposed strategy. Most importantly, the errors for the proposed method are small which shows the additional safety



a: Using the proposed controller. b: Using the MPC-based strategy.

Figure 4.9: The jerk of vehicle n (——) and f (——) during a merge maneuver. It should be noted that the scale is different between the two strategies such that the details remain visible.

Table 4.2: A comparison of the two strategies, the errors after time instance t_{lc} are examined.

Proposed Strategy								
	e_f $[m]$	e_n $[m]$	$\dot{e}_f$ $[\frac{m}{s}]$	$\dot{e}_n$ $[\frac{m}{s}]$	a_f $[\frac{m}{s^2}]$	a_n $[\frac{m}{s^2}]$	j_f $[\frac{m}{s^3}]$	j_n $[\frac{m}{s^3}]$
Max	0.067	0.061	0.019	0.017	1.147	1.426	1.082	0.397
Min	-0.025	-0.009	-0.041	-0.028	-0.876	-0.047	-0.618	-0.821
RMS	0.017	0.019	0.013	0.010	0.430	0.676	0.259	0.186
MPC-based Strategy								
	e_f $[m]$	e_n $[m]$	$\dot{e}_f$ $[\frac{m}{s}]$	$\dot{e}_n$ $[\frac{m}{s}]$	a_f $[\frac{m}{s^2}]$	a_n $[\frac{m}{s^2}]$	j_f $[\frac{m}{s^3}]$	j_n $[\frac{m}{s^3}]$
Max	5.061	0.041	1.526	0.147	2.049	1.426	4.711	22.119
Min	-0.100	-0.685	-0.998	-0.615	-1.499	-0.712	-4.117	-20.838
RMS	2.284	0.328	0.525	0.152	0.882	0.689	0.895	2.272

provided by the proposed method. It should be noted that only errors after t_{lc} are considered because the MPC-based controller does not have an error definition.

The proposed controller is unconstrained, there are only bounds on the expected trajectory of the transitioning controller. However, due to the measurement noise, the actual trajectory may deviate from the expected trajectory. Table 4.2 shows that with the proposed strategy the bounds $a_i \leq \pm 1.2$ m/s^2 and $j_i \leq \pm 0.8$ m/s^3 are surpassed. This underlines the importance of choosing conservative values for these limits.

To conclude, the proposed strategy functions as expected and provides good behavior when noise is introduced. When compared to an MPC-based algorithm, improvements include efficiency (based on velocity and acceleration profiles), safety (based on error profiles), and passenger comfort (based on the jerk profiles). Overall, this is a strong indication that the proposed strategy can handle outside disturbances and is therefore beneficial over existing MPC-based algorithms.

4.4.3 Lead Vehicle Excitation Handling

The previous section showed the advantage of the proposed strategy over a conventional MPC-based strategy. This section discusses the performance of the proposed strategy when experiencing disturbances from the lead vehicle. Such disturbances are interesting because the environment in front of the leading cooperative vehicle is unknown. Especially in mixed-traffic applications, the platoon may encounter a slower driving vehicle requiring the platoon to slow down and match its speed. This section investigates the behavior of the vehicles if such a disturbance occurs during the merging maneuver. The focus of this section is longitudinal behavior. First, the position of the vehicles is discussed since this is the most important performance metric. Next, the velocity and acceleration profiles are looked at, to provide more insight into the behavior. Subsequently, the errors are examined to ensure safety during the maneuver. To conclude the longitudinal jerk is analyzed. This gives insight into the perceived user comfort. The section is concluded with a short analysis of the lateral behavior by examining δ_q.

The position of the vehicles over time for the two scenarios is plotted in Figure 4.10. The start of the lane change (t_{lc}) is indicated on the x-axis. It is apparent that the value of t_{lc} changes based on vehicle p's behavior. When vehicle p accelerates the merging point is reached earlier. Therefore, t_{lc} is earlier in this scenario. For the deceleration scenario, the opposite happens. After t_{lc}, vehicle n drives between vehicles p and f in both scenarios. The merging maneuvers thus appear to be executed successfully. To further examine the performance, the velocities and accelerations of the vehicles are examined.

The velocities and accelerations are shown in Figure 4.11. It is shown that the velocities are approximately constant and equal to that of the preceding vehicle after t_{lc}. This indicates that a steady-state solution is achieved. Additionally, the accelerations remain between ± 2 m/s^2. These are relatively high accelerations but feasible for some production vehicles. Their magnitude is mainly caused by the initial conditions. The controller transition does not appear to cause large accelerations. The individual MPC controller of vehicle n is affected most by the noise. This emphasizes the need for an early transition to the CACC controller.

The controller transitions are also visualized in Figure 4.11. It is shown that vehicle f tends to start the transition first. This is to be expected since vehicle n starts in front of the platoon. An appropriate distance between vehicles f and n can thus be established before that between vehicles n and p. The transitions are finished before t_{lc}. The CACC control strategy is thus employed during the lane change maneuver. The changes in acceleration and velocity during the controller switches appear smooth. The smoothness is commented on in more detail when the jerk is discussed.

Acceleration a_f during the deceleration scenario in Figure 4.11b shows that the collision avoidance intervention is performed. At approximately 7 seconds into the simulation, vehicle f approaches vehicle p too rapidly and the controllers

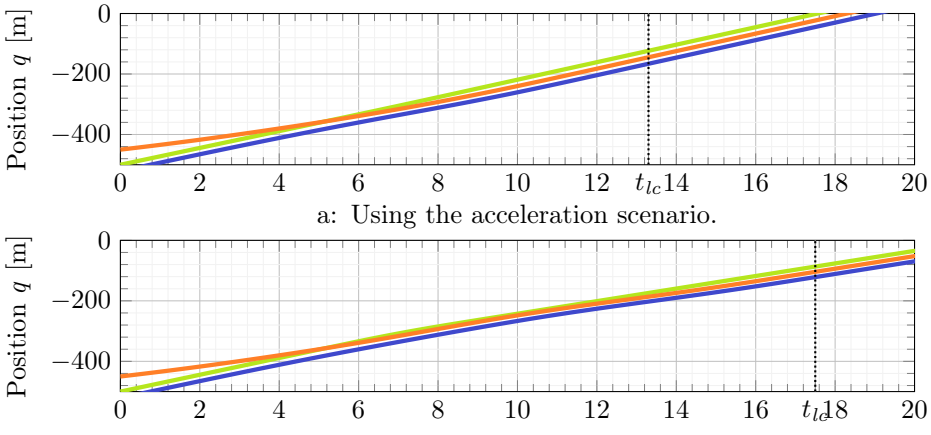


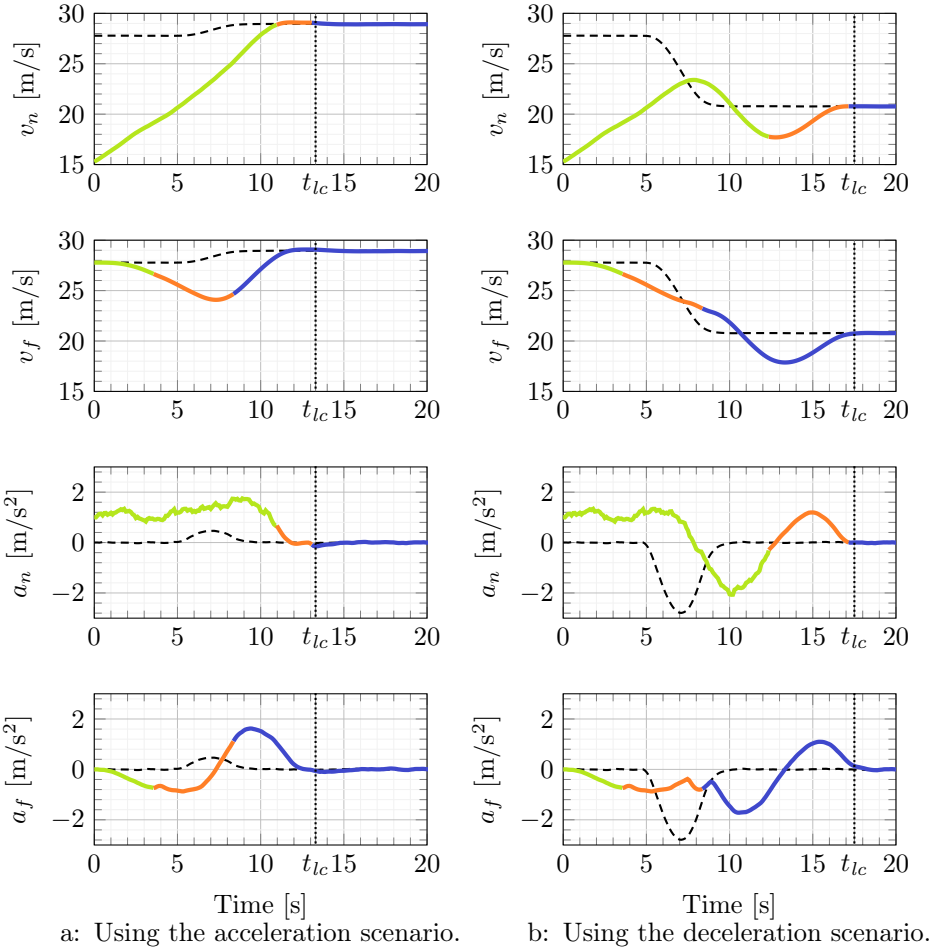
Figure 4.10: The position of vehicle p (—), n (—), and f (—) during a merging maneuver.

are switched. This results in a dip in the acceleration such that sufficient distance is maintained. The exact implications are discussed below when analyzing the error dynamics.

The position error and its derivative during the maneuvers are shown in Figure 4.12. The errors of vehicle n are only shown starting from the controller transition. This is because there is no error definition for the MPC-based controller. The position error stays within approximately ± 0.3 meters except during the collision avoidance maneuver. The error caused by the collision avoidance maneuver will be discussed separately. The derivative of the error generally remains between ± 0.3 m/s, which is an acceptable magnitude. After t_{lc} the errors are well within desired limits. This is the moment for which the magnitude of the errors is most critical as the vehicles are in the same lane. It can therefore be concluded that the maneuver is executed safely.

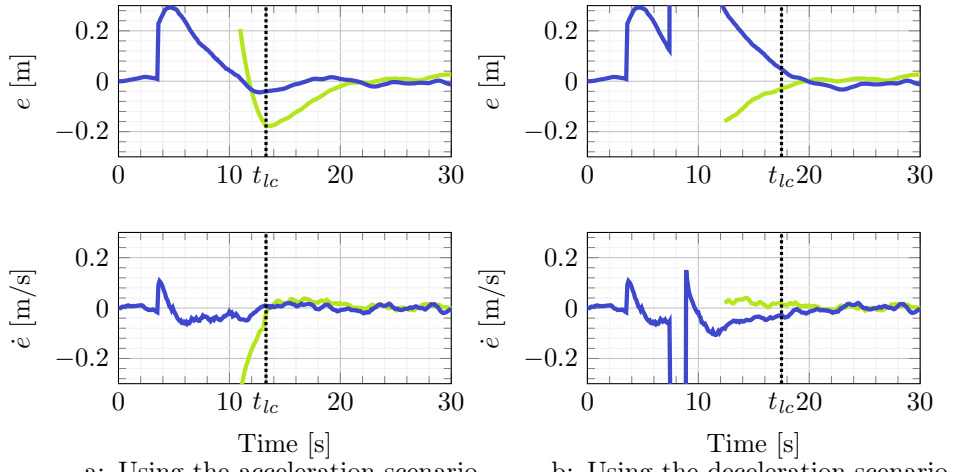
It should be noted that the controller transition aims at an error of zero at the initialization by choosing appropriate values for γ and its derivatives. It is shown that this goal is not completely achieved. This is due to the measurement noise of the radar and on-board sensors. The problem of error initialization is partially solved in vehicle f by using the coefficients of vehicle n's trajectory. Therefore, the error appears closer to zero at the initialization of the controller transition. However, for both vehicles, the error is well within the acceptable limits when the new controller is initialized.

The errors during the collision avoidance maneuver are shown in Figure 4.13. It should be noted that the error definition of vehicle f changes during this



a: Using the acceleration scenario. b: Using the deceleration scenario.

Figure 4.11: The velocity and acceleration of vehicles n and f; before (——), during (——), and after (——) the controller transition. The velocity and acceleration of the preceding vehicle is indicated (----).



a: Using the acceleration scenario. b: Using the deceleration scenario.

Figure 4.12: The error and its derivative for vehicle n (——) and f (——).

maneuver. Between approximately 7 and 9 seconds, the collision avoidance controller intervenes. Consequently, the error in this period is specified with respect to vehicle p. Outside of this period, the error is specified with respect to vehicle n. It is shown that the error during this maneuver becomes large. According to the definition of (4.8), a positive error means that vehicle f is too far behind vehicle p. The collision avoidance controller thus successfully keeps enough distance to vehicle p. The controller interrupted because vehicle f was approaching too rapidly. It is therefore concluded that the maneuver is performed safely. The contribution of the collision avoidance controller to the safety of the maneuver is examined in Section 4.4.4.

The jerk of the vehicles for both scenarios is shown in Figure 4.14. It is shown

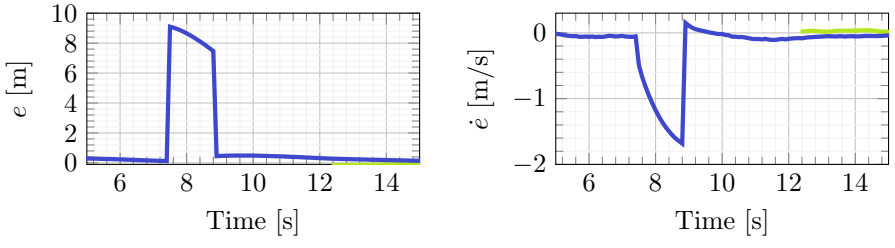
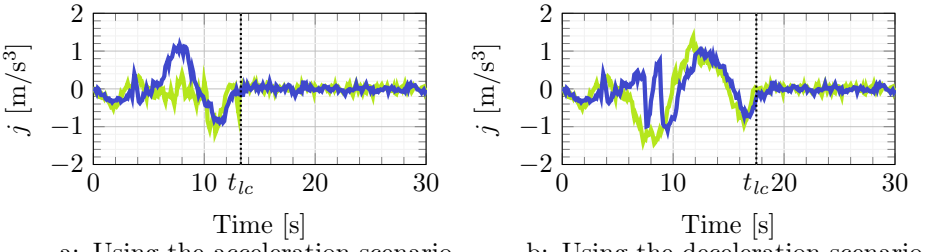


Figure 4.13: The error and its derivative for vehicle n (——) and f (——) during a part of the deceleration scenario. Showing the effect of the collision avoidance controller.



a: Using the acceleration scenario. b: Using the deceleration scenario.

Figure 4.14: The jerk of vehicle n (—) and f (—).

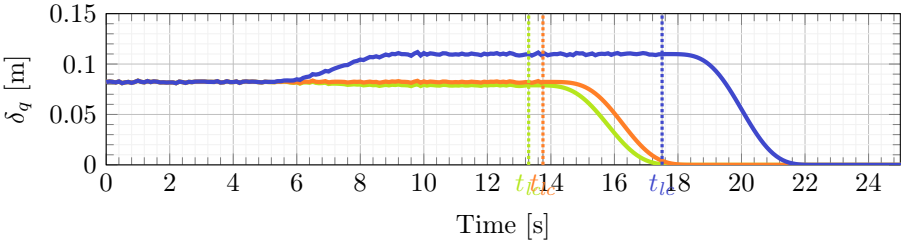


Figure 4.15: The predicted additional length δ_q due to the lateral movement during the acceleration (—), constant velocity (—), and deceleration (—) scenario.

that the jerk stays within acceptable bounds of $\pm$ 3 m/s^3. The interruption of the collision avoidance controller is clearly visible for the deceleration scenario. However, the resulting jerk also stays well within the limits and no peak jerks are experienced. Furthermore, it is visible that the controller transition was successfully initialized such that it does not create any peaks in the jerk. It can therefore be concluded that the controller transition is relatively smooth. Overall, the passenger comfort during this maneuver is considered acceptable.

Now that the longitudinal analysis is completed the lateral behavior of the lane change is discussed. The lane change is designed to travel 4 meters laterally. Vehicle n calculates an additional distance δ_q it needs to drive to achieve this. The behavior of the predicted additional distance δ_q can be found in Figure 4.15. The value of δ_q is approximately constant for constant velocities up to t_{lc}. After t_{lc} the value goes to zero because there is no additional distance to be traveled when vehicle n reaches the main lane. When the platoon accelerates or decelerates δ_q increases or decreases respectively. This variation in δ_q is caused by the fact that a time duration is prescribed for the lane change. The distance traveled in a given time at higher velocities is larger. Therefore, the additional distance is smaller due to the lateral movement.

The figure also shows that δ_q is noisy before t_{lc}. This is caused by the noisy radar velocity measurements which are used to determine the length of the path. After t_{lc}, a steady-state platoon is assumed and vehicle n's own velocity is used, which is measured using on-board sensors. The higher accuracy of the on-board sensors makes the estimation of δ_q much smoother. Lastly, it should be noted that the lane change takes around 5 seconds. At a velocity of 100 km/h this results in a trajectory of approximately 139 meters. The value of δ_q and including its noise is thus small in comparison to the total distance. The effect may be greater in scenarios with a lower velocity.

To conclude, the performance of the proposed method under excitations of the lead vehicle is demonstrated. The method reacts well to disturbances from the lead vehicle. A steady-state platoon can be realized before the lane change, despite measurement noise. In these simulations, only a single acceleration or deceleration event from the platoon leader is encountered. In Appendix A.3 simulations are performed where the leader has a sinusoidal velocity profile. The results show that the algorithm can handle these continuous excitations in a similar manner. The maneuver is executed efficiently, safely, and comfortably. The additional benefit of the collision avoidance controller will be investigated separately in Section 4.4.4. The effect of noise on the behavior of the vehicle is investigated in Section 4.4.5.

4.4.4 Collision Avoidance

In the previous section, the collision avoidance controller was briefly demonstrated. In this section, the necessity of the collision avoidance controller is demonstrated. A more severe braking maneuver is used in this demonstration, such that a collision occurs if the collision avoidance controller is not active. The positions of the vehicles in this simulation are shown in Figure 4.16. At approximately 14 seconds into the simulation, it is visible that not using the collision avoidance controller would result in a collision. Vehicle f overtakes vehicle p, which is not possible in real-world scenarios as they are in the same lane. When the collision avoidance controller is used, vehicle f remains behind vehicle p.

To get a better understanding of safety, the error dynamics are investigated. These dynamics are shown in Figure 4.17. It is shown that the error states remain small when the collision avoidance is not activated. However, it should be noted that after the start of the transition, the error is only specified with respect to vehicle n. As shown in the position plots, this does not result in the desired behavior. Furthermore, it is visible that the usage of the collision avoidance controller causes large positive errors. As discussed previously this means that the distance between the vehicles is too large. The danger of this is that an overshoot may occur when the gap is closed. An appropriate tuning can reduce this overshoot. For the current scenario, the position error after the collision avoidance maneuver (around 16 seconds) is 15 meters. When the gap

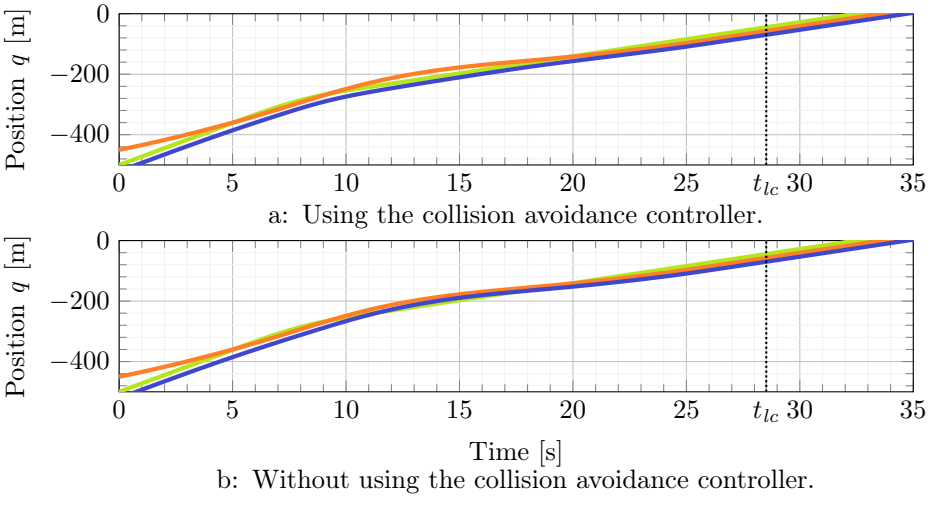


Figure 4.16: The position of vehicle p (—), n (—), and f (—) during a merging maneuver.

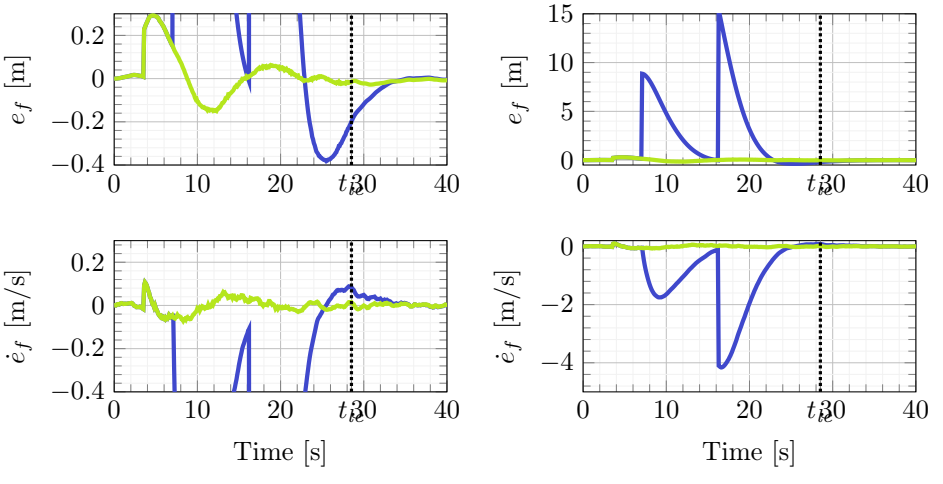
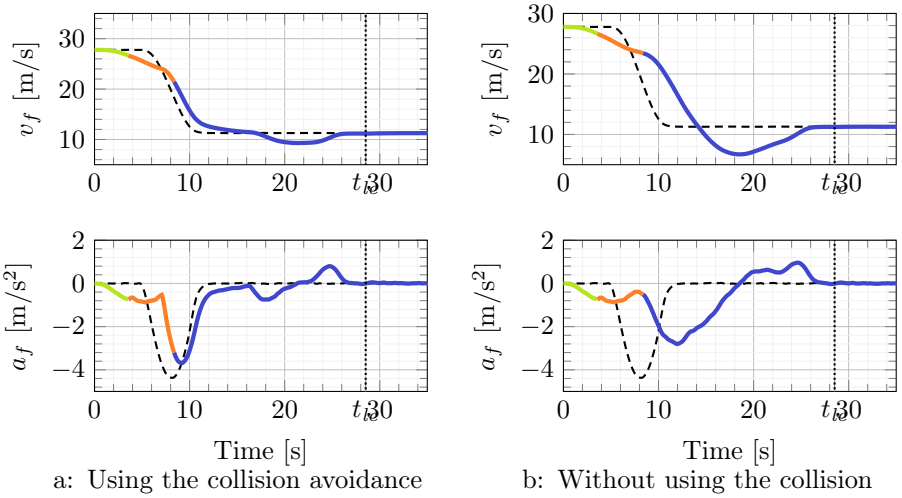


Figure 4.17: The error and its derivative for vehicle f when the collision avoidance controller is off (—) and on (—).



a: Using the collision avoidance controller.

b: Without using the collision avoidance controller.

Figure 4.18: The velocity and acceleration of vehicles n and f during the deceleration scenario; before (——), during (——), and after (——) the controller transition. The velocity and acceleration of the preceding vehicle is indicated (- - - -).

is closed the resulting overshoot is 0.4 meters. This is a reasonable amplitude and after t_{lc} the error is within 0.2 meters. Therefore, the collision avoidance controller appears to behave adequately.

Lastly, to assess the performance of the collision avoidance controller, the velocities and accelerations are briefly analyzed. In Figure 4.18 it is shown that the collision avoidance controller results in heavier decelerations. This is to be expected because an additional deceleration is required to avoid the collision that would occur if this controller isn't used. However, an unexpected result is the velocity is reduced more when the collision avoidance controller is not activated. The activation of the collision avoidance controller results in a velocity profile of vehicle f which is closer to that of vehicle p. This reduces the overshoot in velocity. This phenomenon may also be beneficial for any subsequent vehicles in the platoon.

To conclude, the collision avoidance controller behaves well and has a clear benefit. The usage of the collision avoidance controller may prevent head-on collisions during the maneuver. Furthermore, the velocity of vehicle p is mimicked more closely.

Table 4.3: A summary of the time instances, acceleration, and jerk for 100 simulations with different noise signals.

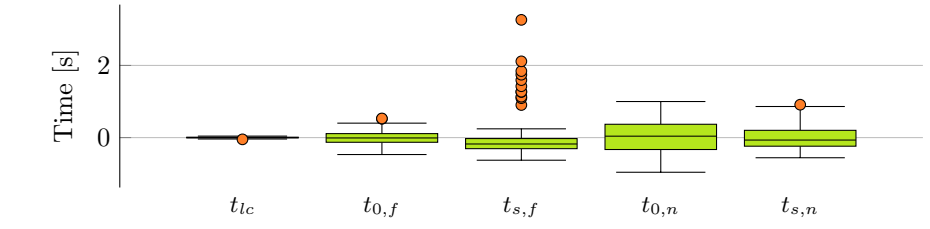
Time instances in seconds					
	t_{lc}	$t_{0,f}$	$t_{s,f}$	$t_{0,n}$	$t_{s,n}$
Mean	13.75	3.57	8.61	7.95	12.46
Max	13.79	4.10	11.87	8.95	13.37
Min	13.70	3.10	7.98	6.99	11.90
Acceleration in m/s^2					
	$\min(a_f)$	$\max(a_f)$	$\min(a_n)$	$\max(a_n)$	
Mean	-0.975	1.009	-0.053	1.485	
Max	-0.813	1.195	-0.027	1.677	
Min	-1.196	0.882	-0.097	1.345	
Jerk in m/s^3					
	$\min(j_f)$	$\max(j_f)$	$\min(j_n)$	$\max(j_n)$	
Mean	-0.649	0.915	-0.808	0.492	
Max	-0.402	1.244	-0.680	0.834	
Min	-0.923	0.681	-0.995	0.171	

4.4.5 Noise Sensitivity

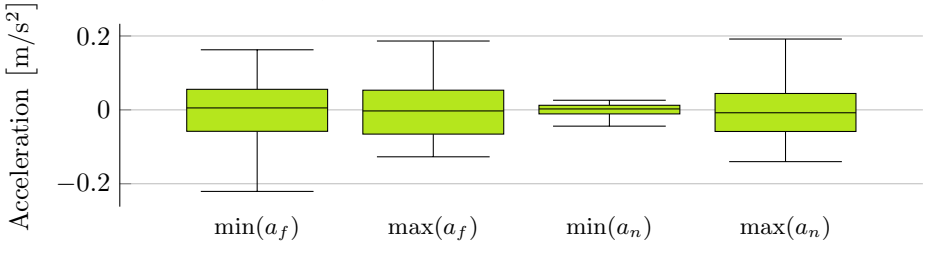
To evaluate the performance an artificial noise and communication delay was added in the simulations of the previous section. In this section, the effect of noise is further investigated by analyzing multiple simulations with different noise signals. The noise will have the standard deviation previously proposed in Section 4.4.1. In total, one hundred simulations were executed to investigate the influence of noise. In these simulations, the lead vehicle has a constant velocity.

First, the important time instances of the maneuver are analyzed in Figure 4.19a. It should be noted that the times in this plot are adjusted such that their mean time is at zero seconds. The average values can be found in Table 4.3. This is done since the mean values lay far apart which would influence the figure's readability. The value of t_{lc} is hardly influenced by the different noise signals. The time instances related to controller switching have more variety. Especially, $t_{s,f}$ has a large variety with a maximum of 3.26 seconds above its mean value. This occurs because the switch of vehicle f is recalculated after vehicle n initiates its switch. The recalculation is necessary because vehicle n alters the coefficients of its planned trajectory. The total transition time of vehicle f for can thus be more than five seconds. Overall, there are no extreme values that are outside the expected range. The results thus suggest that the high-level controller handles noise sufficiently well.

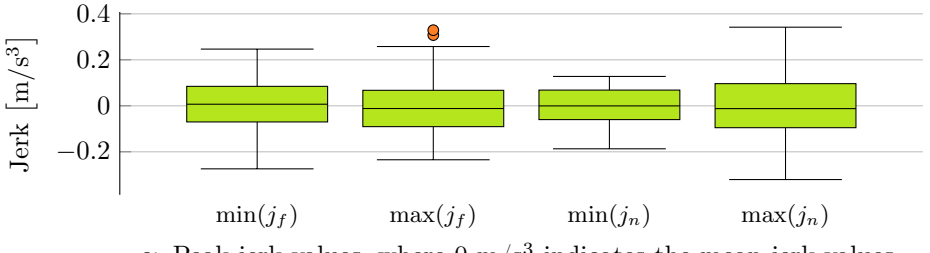
The variations in the acceleration are shown in Figure 4.19b, a summary is provided in Table 4.3. The acceleration of vehicle f lies within ± 1.2 m/s^2.



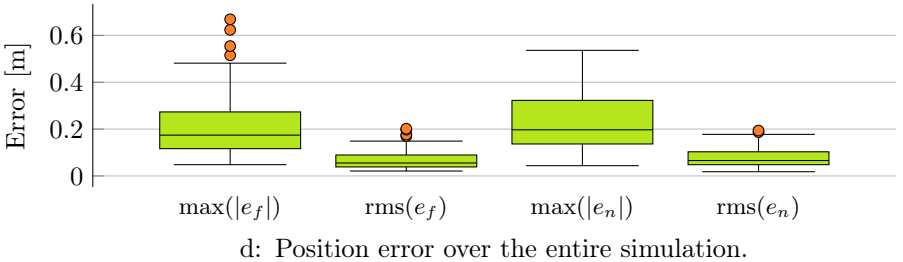
a: Time instances, where 0 seconds indicates the mean time instances.



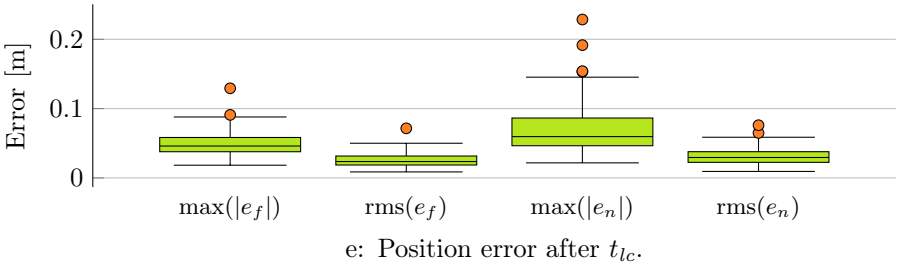
b: Peak accelerations, where 0 m/s^2 indicates the mean accelerations.



c: Peak jerk values, where 0 m/s^3 indicates the mean jerk values.



d: Position error over the entire simulation.



e: Position error after t_{lc}.

Figure 4.19: The variation of behavioral indicators for 100 simulations with different noise profiles. The mean values can be found in table 4.3.

With respect to this range, the variety is relatively large. However, no extreme peak values are found. This indicates the effect of noise on the acceleration is acceptable. Vehicle n has low decelerations since it needs to accelerate to the platoon velocity. The peak acceleration of 1.68 m/s^2 surpasses the limit of $\pm$1.2 m/s^2 that is set on the expected trajectory. However, the peak value is not unreasonably high, the bound is therefore chosen conservative enough for this simulation setup.

The results regarding jerk are provided in Figure 4.19c and Table 4.3. The jerk of vehicle f peaks at 1.24 m/s^3. Overall, the distribution of the jerk for different noise signals appears relatively uniform. The jerk behavior is acceptable, and the controller is sufficiently unaffected by the noise.

The position errors for vehicles f and n are shown in Figure 4.19d. Both vehicles have a maximum error of over 0.6 meters. However, it should be noted that the error is generally largest just after the controller switch. This is because the estimation of appropriate γ values is sensitive to noise. The error dampens over time and therefore the errors will be smaller after time t_{lc}.

The position errors after t_{lc} are shown in Figure 4.19e. This is an important performance indicator because, at t_{lc}, vehicle n initiates its lateral movement. After this time the vehicles may therefore be in the same lane which can cause a collision. A maximum position error of 23 centimeters is observed for vehicle n. The error is well within acceptable limits and shows the controller maintains safe behavior under the influence of noise.

A further investigation into the effect of specific disturbances is provided in the appendices. First, a shift in the initial position of vehicle n is analyzed in Appendix A.1. Next, an error in the estimation of a_p to initialize γ for the controller transition is examined in Appendix A.2. The appendices show that the proposed control strategy can sufficiently handle these disturbances.

4.5 Conclusion and Future Work

This chapter proposes a control strategy for the cooperative merging of a single CAV into a platoon of vehicles. The platoon is assumed to be driving with a conventional CACC algorithm outside of the maneuver. Therefore, the control strategy at a vehicle level is based on the conventional CACC controller such that controller transitions can easily be managed. Additionally, the usage of a CACC algorithm creates a feedback loop that helps the vehicle handle sensor noise. This is a great advantage over other methods, such as MPC-based strategies. Furthermore, the strategy and required controller transitions are designed such that excitations from the platoon leader can be handled. The platoon and new vehicle are properly spaced before a predefined location regardless of any action taken by the platoon leader. This alignment can be started when communication is established, even if there are large differences in the initial location and velocity of the new vehicle and the platoon vehicles.

The performance of the proposed control strategy is analyzed using simulations. Sensor noise and communication delay are added to the simulated sensors to emulate a realistic environment. The proposed strategy is shown to outperform a conventional MPC-based strategy in such circumstances. Furthermore, the response to excitations of a platoon leader is analyzed. The vehicle alignment is completed before a predefined position and possible collisions are avoided. Lastly, a simulation is repeated 100 times with different noise signals to analyze the noise sensitivity. The results indicate that the proposed control strategy is robust and suggest applicability in real-world scenarios.

In Chapter 5, algorithms to select the merging sequence will be investigated. Multiple merging sequence management approaches are presented in existing research. However, these approaches generally use knowledge about vehicles inside a *cooperation zone*. The usage of platoon-specific knowledge may help to account for subsequent vehicles which are inside the platoon but outside of the *cooperation zone*. Simulations with a wider range of initial conditions can be performed when an appropriate merging sequence management strategy is designed. This may result in additional insights regarding the proposed merging strategy. Furthermore, in Chapter 6 experiments concerning the proposed controller transition approach are performed. This will test and demonstrate the noise handling and real-world applicability of the proposed strategy.

Chapter 5

Sequence Management Algorithms for Highway Merging

In Chapters 3 and 4, a control algorithm for merging into cooperative platoons is proposed. This algorithm assumes that it is known between which two platoon vehicles the new vehicle will merge. However, strategies to determine the position within the platoon, known as merging sequence management, are still an open topic in research. The existing strategies used in literature can be divided into three categories, distance-based, time-based, and optimization-based strategies. This chapter proposes an optimization-based strategy that aims to minimize the acceleration of the last vehicle within communication reach. The proposed solution is compared to three benchmark strategies, one from each category. The four selected strategies are examined using a large number of simulations. The results show that distance-based methods are not suitable for highway on-ramp merging scenarios due to the differences in initial velocity of vehicles. The results for time-based and optimization-based strategies are relatively similar. Time-based strategies are easier to implement and less computationally expensive, giving them a possible preference. However, since both categories have potential, further research is required to determine the preferred sequence management algorithm. Based on the results, it is recommended to use time-based strategies as a benchmark and possible alternative in future research on optimization-based strategies.

5.1 Introduction

In this thesis, a control algorithm is proposed for the cooperative merging of a single Connected Automated Vehicle (CAV) into a platoon. Previously, in Chapters 3 and 4, the vehicle-level control algorithms to execute such maneuvers are proposed. One of the prerequisites for this proposed control strategy is that the two platoon vehicles between which the new vehicle will merge have been selected. In other words, the merging sequence is determined at the start of the maneuver. However, merging sequence management is an open problem in research (Chen, 2021; Jing et al., 2019). Several solutions to this problem have been proposed and they can be divided into three categories. First, there are distance-based methods, where the sequence is based on the position of the vehicles. Time-based methods sequence the vehicles based on the Estimated Time of Arrival (ETA) at the end of the on-ramp. Finally, optimization-based methods predict the trajectories of one or multiple vehicles for various sequences and enter the results into a cost function. The cost function then determines the optimal sequence, for example, one that minimizes the root mean square (RMS) value of the acceleration of all vehicles. In this section, the existing literature is reviewed and the proposed algorithm is briefly explained. The design and analysis of this proposed algorithm are further discussed in the remainder of this chapter.

One approach to determine is the merging sequence is using distance-based methods. These methods are the easy to implement because they only use one easily measurable state of each vehicle, their relative position. For this reason, these methods are often used when investigating a merging maneuver for which no specific sequencing method is designed (Rios-Torres and Malikopoulos, 2017b) or as a benchmark to develop new sequencing methods (Chen et al., 2020; Jing et al., 2019). In scenarios where the velocity of the new and main lane vehicles are equal, the expected desired position of the new vehicle is likely close to its relative initial position. A distance-based approach may then be beneficial and achieve comparable results to more advanced approaches (Chen et al., 2020). Furthermore, distance-based methods are easily understood by human drivers. Chae et al. (2018) shows that this can be advantageous when considering mixed traffic scenarios. The investigated problem is automated highway merging of an Automated Vehicle (AV) that aims to merge between two conventional human-driven vehicles in congested traffic situations. If there is insufficient space for the merging vehicle it will drive close to the target lane to indicate its intent to the vehicle closest behind itself. The intention is to compel the driver to slow down and create space. The solution is validated using experiments with one automated vehicle and two human-driven vehicles. This shows that distance-based methods do not necessarily require communication between the vehicles for sensing or maneuver execution.

Distance-based methods are also used for cooperative intersection control.

The vehicles are then sequenced in the order in which they enter a Cooperation Zone (CZ) around the intersection in which each vehicle can communicate with its neighbors. This method is often referred to as First-In-First-Out (FIFO), or alternatively *First-Come-First-Serve* (Dresner and Stone, 2008). Examples are given in Morales Medina et al. (2018) and Vaio et al. (2019), in which virtual platoons are created to cross the intersection. Virtual platooning is a technique that converts the positions of the vehicles to a one-dimensional coordinate based on a fixed point in the environment. In this case, the middle of the intersection. A platoon is then formed in terms of this one-dimensional coordinate system. The technology was first introduced in Uno et al. (1999) in the context of cooperative merging. Due to the usage of platooning, the theory from the cooperative intersection problem may thus be applicable to the cooperative merging problem. A difference between the problems is the expected difference in initial velocity. The intersection problem often assumes similar velocities for all vehicles. However, for the merging problem, it can be assumed that the vehicle on the on-ramp has a much lower initial velocity than the vehicles on the highway (Cao et al., 2015).

Distance-based policies are suboptimal when handling large initial velocity differences (Wang et al., 2021). In essence, if two vehicles enter the CZ at the same time, they require the same average velocity to reach the predefined merging point at the same time. If the initial velocities differ, additional accelerations and decelerations are required to ensure the average velocity is matched. In the context of merging into a platoon, this means that if a new vehicle must reach a gap near its initial position while its initial velocity is lower than that of the platoon, additional accelerations and decelerations will be required. A position further back in the platoon would limit these excitations and be preferable. To consider the initial velocities of the vehicles, a time-based method can be used. For example, in Wang et al. (2018) and Eiermann et al. (2020), the sequence is based on the Estimated Time of Arrival (ETA) of each vehicle. To determine the ETA, a trajectory is computed using a user-defined maximum acceleration to go from the initial velocity to the highway speed. The vehicles are then sequenced according to their ETA. Such a method can also be implemented in connected human-driven vehicles. This is validated using human-in-the-loop simulations with a driving simulator (Wang et al., 2021).

Alternatively, an optimization-based policy can be used. Such a policy predicts the trajectories of the vehicles for different merging sequences. The optimal sequence can then be chosen based on a cost function. In Jing et al. (2019) an optimization-based sequencing policy is used for highway on-ramp merging. The trajectory planning and merging sequencing are treated separately. The trajectories are calculated using a predetermined optimal function such that the expected behavior of the vehicles can be computed. This behavior is used in the sequencing algorithm. The cost function penalizes the acceleration and jerk of all vehicles in the CZ. Using this cost function, a game-theory approach is utilized to achieve a consensus on the optimal sequence. It is shown that this method

outperforms the First-In-First-Out (FIFO) policy. This is expected based on the shortcomings of the FIFO policy regarding variations in initial conditions. However, it is unknown how this optimization-based policy compares to time-based methods. Furthermore, in Mahbub et al. (2020) a decentralized optimal control strategy is used for situations in which CAVs have a potential conflict. These include intersections, roundabouts, and on-ramps. The problem is divided into an upper-level and a lower-level optimization problem. The upper-level problem decides the merging sequence and aims to minimize the total time to process CAVs. Based on this sequence all vehicles are provided a time at which they will enter the conflict zone. The lower-level controller computes the optimal control input for each vehicle and ensures the times provided by the upper-level controller are met.

In Cao et al. (2014) the merging sequence and trajectory planning problems are combined into one optimal control problem. In essence, the problem aims to minimize a cost function that is the sum of four components. The first two components constrain all inter-vehicle distances and the position of the new vehicles. The other components minimize the deviations from desired velocities and the accelerations of all vehicles. This cost function requires the position, velocity, and acceleration of all vehicles. The optimal control problem is then solved in real-time using Model Predictive Control (MPC) which results in a control input for all vehicles simultaneously. It should be noted that due to the information required for the cost function and its solution the method is mainly applicable for a centralized system layout, meaning a single controller determines the trajectories of all vehicles.

In this chapter, an optimization-based sequencing strategy is proposed. The aim of this strategy is to minimize the peak accelerations of the last platoon vehicle in the CZ or its equivalent. Since excitations of a vehicle affect those of all subsequent vehicles, this approach has the potential to minimize excitations of subsequent platoon vehicles that are outside of the cooperation set. In literature, acceleration is often used as an indicator of efficiency. Using this performance metric, the resulting trajectories of all vehicles within the platoon may thus be more efficient. Often a newly designed strategy is solely compared to a distance-based strategy as a benchmark. Due to known shortcomings of distance-based strategies, other strategies may be a more suitable benchmark and provide a better insight into the performance of the newly developed strategy. Therefore, the proposed strategy is compared to distance-based, time-based, and optimization-based strategies which are based on literature. The performance of the proposed strategy is analyzed and the benchmarks are also compared to each other. Using a large number of simulations with different initial conditions and disturbances, the performance of the proposed strategy is assessed. Furthermore, based on the findings, recommendations regarding benchmarks for future research can be formed.

The remainder of this chapter is ordered as follows. First, in Section 5.2

the merging problem and on-ramp environment is introduced. An important property regarding sequencing methods is the communication strategy. This determines which vehicles can communicate and therefore be involved in the proposed sequence. Next, in Section 5.3 an individual controller is proposed that can be used by the new vehicle when it has not yet found a feasible merging sequence. The desired position and velocity for entering the new vehicle into the main lane are known. However, the individual controller also determines the desired time. This time can later be used in the time-based sequencing benchmark. A detailed description of the sequencing strategies is given in Section 5.4. Then, in Section 5.5 the simulation setup and the merging scenarios is explained. The section also contains the simulation results and their analysis. Lastly, Section 5.6 gives conclusions regarding the merging strategy and discusses the implications of the results.

5.2 Problem Statement

The overarching aim of this thesis is to design a control strategy for merging a single vehicle into a cooperative platoon at highway on-ramps. If long platoons are driving on the highway, merging in front of, or behind the platoon may be infeasible or undesirable. The platoon is assumed to have small inter-vehicle distances, requiring it to create space when accommodating new vehicles. To successfully perform the maneuver, some platoon vehicle is therefore required to perform actions that are not in its own interest. The previous chapters describe an algorithm to successfully complete these maneuvers, assuming it is known where the new vehicle will join, and which platoon vehicle is required to open a gap.

The specific aim of this chapter is to develop a sequencing algorithm for highway on-ramp merging. This refers to the sequence of vehicles after the merging maneuver. The sequence thus dictates where in the platoon the new vehicle will be driving and by extension the trajectories of all vehicles involved in the maneuver, including all subsequent platoon vehicles.

It is assumed that all vehicles involved in the maneuver are CAVs. Therefore, the vehicles can share information amongst each other. This information can include physical states, such as velocity and position, and intend such as a merging sequence or desired acceleration.

This section introduces the subproblems tackled in this chapter. Secondly, the assumptions for the on-ramp merging scenario are discussed. Lastly, assumptions regarding the communication structure are specified in detail.

5.2.1 Subproblems

To solve the main problem, the following three subproblems are considered:

1. How should the new vehicle behave when it has not (yet) established communication with other vehicles on the highway?
2. How should the merging sequence be determined when the new vehicle has established communication with some platoon vehicles on the highway?
3. How should the new vehicle use the merging sequence management strategy?
4. How should the vehicles react when unexpected behavior of the platoon leader renders the chosen sequence infeasible?

The first problem is tackled by designing an individual controller, in Section 5.3. The aim of this controller is for the new vehicle to achieve the desired main lane velocity at the end of the on-ramp. Since there is not yet any interaction with the platoon, the time at which the end of the on-ramp must be reached is decided by the controller.

The second problem is that of merging sequencing and will be determined by the merging sequence manager. This problem is the main topic of this chapter. Due to the high variety of existing approaches in literature, three benchmark approaches with different philosophies are included in the analysis. Simulation results are analyzed to create a broad understanding of the different methodologies.

The last two problems will be handled by a high-level controller. This controller is responsible for deciding which lower-level controller to use. It considers what actions to take if previously unnoticed vehicles establish communication with the new vehicle. Furthermore, the high-level controller decides how to react to unforeseen disturbances in the platoon.

5.2.2 System Assumptions

To analyze and develop the strategy, certain assumptions are made when modeling the system. The main assumptions are:

1. All vehicles in the scenario can determine their own states, including position, velocity, and acceleration. However, sensor noise may be added to their velocity and acceleration measurements.
2. All vehicles in the scenario are equipped with perception sensors to determine the relative position and velocity of nearby vehicles. These measurements are subject to sensor noise.
3. All vehicles in the scenario are CAVs with the ability to broadcast and receive information through Vehicle-to-Vehicle (V2V) communication.
4. All vehicles on the main lane are part of the platoon. There is only a single vehicle on the on-ramp.

5. The merging sequence is decided by the new vehicle. Platoon vehicles always accept the chosen sequence and create a gap if necessary.

6. Only longitudinal control is considered; the lateral control can be performed by existing technology.

The assumptions regarding the available technology on all vehicles are quite restrictive because it requires 100% market penetration of highly advanced CAVs. This is for the purpose of analyzing the proposed method. Extensions to mixed traffic scenarios must be made in future research. The study is limited to longitudinal control. This is justified by the results of Chapter 4 which show that lateral control has a limited influence on the system's performance.

5.2.3 Communication Range

One important aspect of the problem is the communication range of the new vehicle. In essence, it is assumed that the new vehicle can only communicate with vehicles that are within a certain range. This range is constrained by the communication equipment. The problem considers V2V communication therefore the range is centered around the new vehicle. The usage of a Road Side Unit (RSU) is therefore not considered. It should be noted that for some communication strategies a communication range may not applicable (e.g., when using internet). However, it is likely that the new vehicle will always consider a limited number of vehicles on the highway. Some version of the communication range problem will therefore remain.

Figure 5.1 shows the communication range and its implications for a highway on-ramp scenario. The figure highlights the main issue, namely that not all platoon vehicles are necessarily inside the communication range when the merging sequence is determined. The set of vehicles within communication range at time k is defined as cooperation set $\mathcal{C}_k$. Ideally, the merging sequence is determined such that it also considers its impact on platoon vehicles outside the communication range. It should be noted that the vehicles inside the communication range will change throughout the maneuver. Initially, the new vehicle may be far away from the highway, in which case no vehicles will be in the communication range. As the new vehicle approaches the highway, the number of vehicles inside the communication range increases. As can be deducted from the figure, this is dependent on the on-ramp geometry. For modeling purposes, the on-ramp is divided into a ramp and an acceleration lane. The acceleration lane is parallel and attached to the highway. The shape of the ramp is an important aspect of the on-ramp environment because the communication range is expressed in the two-dimensional distance. The closer the ramp is to the main lane, the earlier communication between the new vehicle and platoon can be established. The definition of the on-ramp is only considered in modeling the communication range, the longitudinal controller of the new vehicle uses a one-dimensional

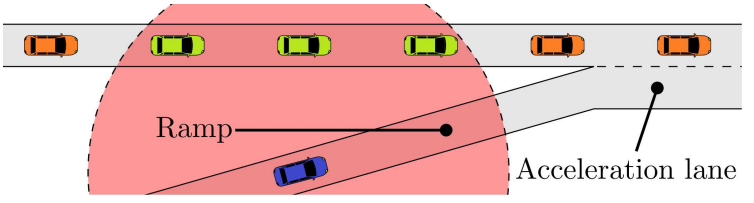


Figure 5.1: A visualization of the communication range (□) of the new vehicle (■), with platoon vehicles in the communication range (■), and platoon vehicles outside the communication range (■).

coordinate system and is not influenced by the on-ramp geometry. Lastly, it can be possible for vehicles to leave the communication range, dependent on the velocity difference between the new vehicle and the platoon.

5.3 Individual Control Strategy

This section discusses the individual control strategy for the on-ramp merging problem. This individual strategy can be used by the new vehicle before communication with the platoon is established. The intent of the new vehicle is to merge onto the highway. Therefore, achieving the highway velocity at the end of the on-ramp is seen as its main goal. In the following subsection, a method to plan a trajectory to the end of the on-ramp is proposed. This method uses a cost function to determine the optimal duration of the trajectory. Next, the influence of the duration on the cost is illustrated using a numerical example.

5.3.1 Trajectory Optimization for the Individual Controller

It is important to consider how the new vehicle approaches the highway before any vehicles are in its communication range. This behavior is required for the execution of the merging maneuver and to produce accurate simulations. For this problem, the desired terminal position, velocity, and acceleration of the designed trajectory are known. However, the desired final time is not known because there is no highway vehicle detected with which the new vehicle should interact. Therefore, the main challenge is to determine t_f^*. This can be done using optimization techniques (Chiang, 1992) and a similar solution is previously proposed in Zhou et al. (2018, 2019).

To determine t_f^*, is considered a point mass and a linearized one-dimensional longitudinal vehicle model is defined as

$$\dot{x}_1(t) = x_2(t) \tag{5.1}$$
$$\dot{x}_2(t) = x_3(t) \tag{5.2}$$

$$\dot{x}_3(t) = u_j(t). \tag{5.3}$$

Here x_1, x_2, and x_3 denote the position, velocity, and acceleration respectively. Control input $u_j(t)$ is the desired jerk trajectory. This is different from the previously proposed vehicle model in Chapter 3. The simplified model does not consider any delays or driveline dynamics, which is common for this type of research (e.g., Jing et al. (2019); Mahbub et al. (2020); Ntousakis et al. (2016); Rios-Torres and Malikopoulos (2017b); Vaio et al. (2019); Zhou et al. (2019)). In essence, the optimization prescribes a planned trajectory that can serve as an input to the lower-level vehicle controllers. These lower-level controllers consider the vehicle dynamics to ensure adequate tracking of the reference trajectory. A reference jerk trajectory is followed during the experiments presented further on in this thesis; the lower-level control solution is presented in Section 6.2.3. The vehicle dynamics should however be considered when defining the constraints or optimality criterion. For longitudinal vehicle dynamics, a combination of the velocity, acceleration, and jerk trajectories can be used to compute the necessary constraints in most cases. For example, the required wheel power can be computed using the vehicle mass, velocity, and acceleration. However, this is something that should carefully be considered when specific constraints on the driveline dynamics are present. Some constraints may not be computable with these trajectories, requiring a more sophisticated model. For a typical highway merge, it is assumed the initial and desired final conditions are such that the computed trajectories remain within a feasible range for the vehicle. To ensure feasibility, the planned trajectories can be investigated at every time step before they are communicated to the lower-level controllers. When desired, these reference trajectories can also be examined using a more sophisticated longitudinal vehicle model that includes specific driveline dynamics. If the proposed trajectories are infeasible, other control methods can be used such as the approach with state constraints of Zhou et al. (2018, 2019).

The objective is to achieve the maximization

$$\max_{u_j(t),\, t_f} J(u_j(t), t_f), \tag{5.4}$$

with

$$J(u_j(t), t_f) = \int_{t_0}^{t_f} \left(-\frac{1}{2} u_j^2(t) - w_{ind} \right) dt. \tag{5.5}$$

Here t_0 and t_f denote the initial and final time. Furthermore, w_{ind} is a positive constant term that has been added to penalize a large t_f, its purpose and amplitude are discussed later. It is apparent that $-\frac{1}{2} u_j(t)^2 \leq 0$. Based on the application, desired initial and terminal conditions for the states can be defined. The objective is to find the optimal $u_j(t)$ which maximizes $J(u_j(t), t_f)$ between the desired initial and terminal states which includes an optimal value of t_f. It should be noted that $u_j(t)$ is the desired reference jerk trajectory. The optimal solution is therefore one that minimizes the longitudinal jerk of the vehicle,

which is linked to passenger comfort (Hoberock, 1977). Intuitively, if t_f is small the excitations need to be large to reach the desired terminal states resulting in a small negative $J(u_j(t), t_f)$. If t_f is large the integral in $J(u_j(t), t_f)$ spans a large time but $u_j(t)$ remains small which can lower the total cost. A small value of w_{ind} is used to avoid unreasonably large values of t_f.

The initial states are those of the vehicle at t_0. Essentially, the initial states are defined as $x_{1,0} = x_1(t_0)$, $x_{2,0} = x_2(t_0)$, and $x_{3,0} = x_3(t_0)$. The final states $x_{1,f} = x_1(t_f)$, $x_{2,f} = x_2(t_f)$, and $x_{3,f} = x_3(t_f)$ are such that at t_f the vehicle is at the start of its lane change maneuver. The vehicle must then drive at a desired velocity and have zero acceleration. The final states are thus defined as

$$x_{1,f} = q_{mp} - x_{2,f}\tau_{lc} \tag{5.6}$$

$$x_{2,f} = v_f \tag{5.7}$$

$$x_{3,f} = 0. \tag{5.8}$$

Here q_{mp} denotes the merging point, τ_{lc} denotes the time required for the lane change and v_f is the desired final velocity which can be chosen freely.

To solve this problem, the Hamiltonian (Chiang, 1992) is formed as

$$\mathcal{H}\left(x(t), \lambda(t), u_j(t)\right) := -\frac{1}{2}u_j^2\left(t\right) - w_{ind} + \lambda_1(t)x_2(t) + \lambda_2(t)x_3(t) + \lambda_3(t)u_j(t). \tag{5.9}$$

Here λ denotes the co-state vector. As previously discussed, no input and state constraints are considered, therefore the Lagrangian is defined as

$$\mathcal{L}\left(x(t), \lambda(t), u_j(t)\right) := \mathcal{H}\left(x(t), \lambda(t), u_j(t)\right). \tag{5.10}$$

Now expressions for the co-states can be determined using

$$\dot{\lambda}_1^*(t) = -\frac{\partial \mathcal{L}^*}{\partial x_1} = 0, \qquad \lambda_1^*(t) \equiv c_1 \tag{5.11}$$

$$\dot{\lambda}_2^*(t) = -\frac{\partial \mathcal{L}^*}{\partial x_2} = -\lambda_1^*(t), \qquad \lambda_2^*(t) \equiv -c_1 t - c_2 \tag{5.12}$$

$$\dot{\lambda}_3^*(t) = -\frac{\partial \mathcal{L}^*}{\partial x_3} = -\lambda_2^*(t), \qquad \lambda_3^*(t) \equiv \frac{1}{2}c_1 t^2 + c_2 t + c_3. \tag{5.13}$$

These equations are valid almost everywhere. Here c_1, c_2, and c_3 denote constant coefficients. To obtain the optimal values of these coefficients, the partial differential equation

$$\frac{\partial \mathcal{H}}{\partial u_j} = -u_j^*(t) + \lambda_3^*(t) = 0 \tag{5.14}$$

is used. This yields $u_j^*(t) = \lambda_3^*(t) = \frac{1}{2}c_1 t^2 + c_2 t + c_3$. Using (5.1), (5.2), and (5.3) it can be stated that

$$x_1(t) = \frac{1}{5!}c_4 t^5 + \frac{1}{4!}c_2 t^4 + \frac{1}{3!}c_3 t^3 + \frac{1}{2}c_4 t^2 + c_5 t + c_6 \tag{5.15}$$

$$x_2(t) = \frac{1}{4!}c_4t^4 + \frac{1}{3!}c_2t^3 + \frac{1}{2}c_3t^2 + c_4t + c_5 \tag{5.16}$$

$$x_3(t) = \frac{1}{3!}c_4t^3 + \frac{1}{2}c_2t^2 + c_3t + c_4. \tag{5.17}$$

The six coefficients can now be chosen such that they satisfy $x_{1,0} = x_1(t_0)$, $x_{2,0} = x_2(t_0)$, $x_{3,0} = x_3(t_0)$, $x_{1,f} = x_1(t_f)$, $x_{2,f} = x_2(t_f)$, and $x_{3,f} = x_3(t_f)$. To satisfy the initial conditions, $c_6 = x_{1,0}$, $c_5 = x_{2,0}$, and $c_4 = x_{3,0}$. The terminal conditions yield

$$c_1 = -\frac{60\left(12x_{1,0} - 12x_{1,f} + 6t_fx_{2,0} + 6t_fx_{2,f} + x_{3,0}t_f^2 - x_{3,f}t_f^2\right)}{t_f^5} \tag{5.18}$$

$$c_2 = \frac{12\left(30x_{1,0} - 30x_{1,f} + 16t_fx_{2,0} + 14t_fx_{2,f} + 3x_{3,0}t_f^2 - 2x_{3,f}t_f^2\right)}{t_f^4} \tag{5.19}$$

$$c_3 = -\frac{3\left(20x_{1,0} - 20x_{1,f} + 12t_fx_{2,0} + 8t_fx_{2,f} + 3x_{3,0}t_f^2 - x_{3,f}t_f^2\right)}{t_f^3}. \tag{5.20}$$

Using these conditions, the trajectories of $x_1(t)$, $x_2(t)$, and $x_3(t)$ can be computed. It should be noted that for any initial and desired terminal values the coefficients will be dependent on t_f. Therefore, the optimal t_f^* will be obtained next.

Now based on (5.6), (5.7), and (5.8) let

$$b_1(t_f) := x_1(t_f) + \tau_{lc}x_2(t_f) - x_m \tag{5.21}$$

$$b_2(t_f) := x_2(t_f) - v_f \tag{5.22}$$

$$b_3(t_f) := x_3(t_f). \tag{5.23}$$

The transversality conditions are defined as

$$\lambda_1^*(t_f^*) = \beta_1\frac{\partial b_1^*}{\partial x_1}(t_f^*) + \beta_2\frac{\partial b_2^*}{\partial x_1}(t_f^*) + \beta_3\frac{\partial b_3^*}{\partial x_1}(t_f^*) = \beta_1 \tag{5.24}$$

$$\lambda_2^*(t_f^*) = \beta_1\frac{\partial b_1^*}{\partial x_2}(t_f^*) + \beta_2\frac{\partial b_2^*}{\partial x_2}(t_f^*) + \beta_3\frac{\partial b_3^*}{\partial x_2}(t_f^*) = \beta_1\tau_{lc} + \beta_2 \tag{5.25}$$

$$\lambda_3^*(t_f^*) = \beta_1\frac{\partial b_1^*}{\partial x_3}(t_f^*) + \beta_2\frac{\partial b_2^*}{\partial x_3}(t_f^*) + \beta_3\frac{\partial b_3^*}{\partial x_3}(t_f^*) = \beta_3. \tag{5.26}$$

These yield $\beta_1 = \lambda_1^*(t_f^*)$, $\beta_2 = \lambda_2^*(t_f^*) - \lambda_1^*(t_f^*)\tau_{lc}$, and $\beta_3 = \lambda_3^*(t_f^*)$. The Hamiltonian then becomes

$$\mathcal{H}^*(t_f) = -\frac{1}{2}\left(u_j^*\left(t_f^*\right)\right)^2 - w_{ind} + \beta_1\frac{\partial b_1^*}{\partial t_f}(t_f^*) + \beta_2\frac{\partial b_2^*}{\partial t_f}(t_f^*) + \beta_3\frac{\partial b_3^*}{\partial t_f}(t_f^*). \tag{5.27}$$

It can be noted that using (5.21), (5.22), (5.23), and the definitions of β_1, β_2, and β_3, the equations (5.27) and (5.9) match. t_f^* can be obtained using an additional

transversality condition $[\mathcal{H}^*]_{t=t_f^*} = 0$ (Chiang, 1992). Consequently, t_f^* is the solution of

$$c_1 x_{2,f} + \frac{1}{2}\left(\frac{1}{2}c_1 {t_f^*}^2 + c_2 t_f^* + c_3\right)^2 - x_{3,f}\left(c_2 + c_1 t_f^*\right) - w_{ind} = 0. \tag{5.28}$$

If there is a non-unique solution for (5.28), the smallest positive solution is used.

Using t_f^* the trajectory of the vehicle can be computed using (5.15) up to (5.20). This trajectory can then be tracked using a standard longitudinal controller. Since this is an analytical solution, it is computationally inexpensive to compute t_f^* and this can be done in real-time. To get a better understanding of the cost function and the purpose of w_{ind} a numerical example follows in the next section.

5.3.2 Numerical Example of the Optimal Final Time Computation

To understand the cost function and the influence of t_f on its value, a numerical example is provided in this section. For this example, the initial and terminal conditions are chosen as $x_{1,0} = 0$ m, $x_{2,0} = 10$ m/s, $x_{3,0} = 1$ m/s^2, $x_{1,f} = 350$ m, $x_{2,0} = 25$ m/s, $x_{3,f} = 0$ m/s^2. Furthermore, t_0 is chosen to be 0 seconds and w_{ind} to be 0.01. The resulting trajectories will be investigated to ensure the velocity is less than 27.78 m/s, the acceleration is within ± 1.2 m/s^2 and the jerk is within ± 0.8 m/s^3.

Following the procedure in Section 5.3.1, t_f^* is found to be 18.41 seconds. The cost for different values of t_f is illustrated in Figure 5.2. It is shown that for small values of t_f the cost is high. This is because high excitations are required to achieve the desired terminal states in such a short time. After the optimum is reached the cost increases again, this is due to the longer range of the integral. For large values of t_f the value of $\int_{t_0}^{t_f}\left(-\frac{1}{2}u_j^2(t)\right)dt$ decreases, the low excitations then outweigh the range of the integral. However, such extreme values of t_f are undesirable for practical implementation since the average velocity will be very low. Hence, the term $-w_{ind}$ is introduced in the cost function to ensure a global minimum and avoid high values of t_f. The value of w_{ind} can be chosen by the user. If the value is too high the resulting value of t_f^* is strongly affected. If the value is too low, t_f^* may become unreasonably large. In this example, the influence influence $\int_{t_0}^{t_f}(-w_{ind})\,dt$ on t_f^* is relatively small due to its shallow slope. The value of t_f^* is 0.07 seconds smaller when w_{ind} is included in the cost function. Furthermore, the ideal value of w_{ind} is scenario dependent and the current choice appears conservative enough to apply to multiple scenarios. Fine-tuning of this factor is therefore not required.

The planned trajectories using t_f^* are shown in Figure 5.3. For illustrative purposes, trajectories for which the duration is extended and shortened by 5 seconds have also been included. When the trajectory is shortened by 5 seconds, the average velocity must be higher to reach the same final position. This

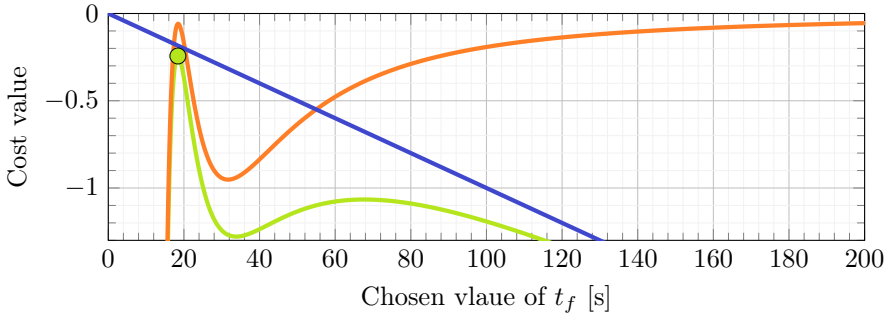


Figure 5.2: The cost for different values of t_f, with the total cost $J(u_j(t), t_f)$ (—), the value of $\int_{t_0}^{t_f} \left(-\frac{1}{2}u_j^2(t)\right) dt$ (—), and the value of $\int_{t_0}^{t_f} (-w_{ind})\, dt$ (—). With the cost associated with t_f^* marked as (●).

is achieved by overshooting the desired velocity causing a high peak velocity that exceeds the bound of 27.78 m/s. Therefore, a high acceleration is required followed by a deceleration to return to the desired velocity. Due to this behavior, the trajectory in case of the shortened duration exceeds the acceleration and jerk bounds. When the duration is extended the average velocity during the maneuver must be lower. Therefore, the vehicle decelerates directly as the maneuver begins. Then, a slightly higher acceleration is required to achieve the desired final velocity. The acceleration bound is therefore exceeded in case of the extended duration. The proposed trajectories using t_f^* stay within the user-specified bounds. It can therefore be decided to follow this proposed trajectory.

Figure 5.2 shows that t_f^* indeed results in a global maximum of the cost function. This demonstrates the theory of Section 5.3.1. This theory can be used to compute the individual control input. Furthermore, t_f^* may be used in a time-based sequencing manager. More information regarding this application is given in Section 5.4.4.

5.4 Merging Sequencing Management

This section introduces the proposed merging sequence manager. The manager can be divided into a high-level controller and a sequencing strategy. The high-level controller determines if the merging sequence should be computed, and the merge should be initiated. The sequencing strategy then determines the preferred sequence for the platoon after the merge. In this work, a method that uses an optimization that considers the last vehicle in the communication range is proposed.

In literature, multiple other approaches are presented for merging sequencing. The existing methods can be divided into three categories: distance-based, time-

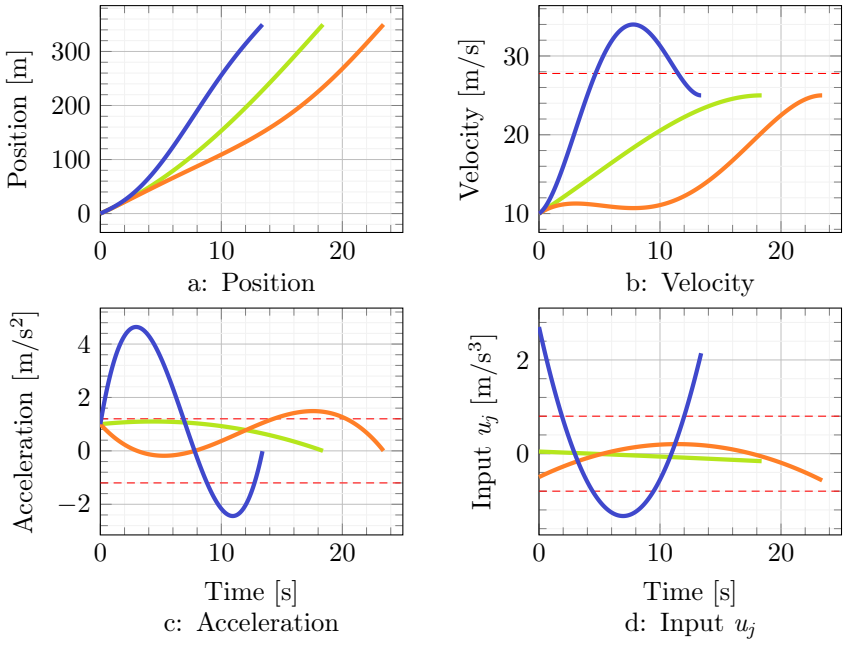


Figure 5.3: The optimal position, velocity, acceleration, and input u_j trajectories for $t_f = t_f^*$ (—), $t_f = t_f^* + 5$ (—), and $t_f = t_f^* - 5$ (—). The bounds on the velocity, acceleration and jerk are included (----).

5

based, and optimization-based. These terms are previously explained in the introduction. In this section, a benchmark method for each of these categories is proposed. These benchmarks are used to assess the performance of the proposed strategy. The high-level controller will be equal for all strategies.

5.4.1 High-level Control

The high-level controller is important for the practical implementation of the merging sequence manager. This controller is run on the new vehicle which coordinates the merging maneuver. In essence, the high-level controller determines which controller is active at any given time. Its design is briefly explained in *Algorithm 5.1*. It is an iterative controller that at every time step k examines the set of vehicles in the cooperation set $\mathcal{C}_k$. If the vehicles in $\mathcal{C}_k$ change and no following vehicle has been selected the sequence management algorithm is run. Therefore, the sequence manager can be run again if the new vehicle is merging at the end but is avoided if a platoon vehicle has been instructed to open a gap. The sequence manager uses state vectors $\vec{x}$ of all available vehicles, which contains their current position q, velocity v, acceleration a, desired acceleration

u, and possible values of gap term γ and its derivatives.

Disturbances from the platoon leader may render a chosen sequence infeasible during the maneuver, requiring the sequence to be altered for successful completion of the maneuver. The low-level controller of the new vehicle computes predicted states $\mathbf{X}_{k,new}$. If its predicted acceleration $\vec{a}_{k,new}$ or jerk $\vec{j}_{k,new}$ is outside of bounds τ_a or τ_j, the sequence is recalculated. This requires a gap-closing action of the previous following vehicle and a gap-opening action of the current following vehicle. These additional actions should be avoided if possible, which is to be considered in the resequencing algorithm.

The high-level controller is employed from the moment the new vehicle approaches the on-ramp. The sequence manager is not triggered until a vehicle has entered $\mathcal{C}_k$. The high-level controller, including the resequencing function, is run until the Cooperative Adaptive Cruise Control (CACC) transition of the new or following vehicle is started. It is thus important that the following vehicle communicates if it starts its CACC transition. When the vehicles start their transition, they are close to CACC platooning. Therefore, gap-closing and opening maneuvers triggered by a sequence change need to be finished within a short period of time. This decreases the chance of finding a feasible other sequence at this point.

Algorithm 5.1: High-level controller for cooperative merging into platoons.

Result: ControllerActive, PreID, FolID
PreID = $\emptyset$; `// Index of the preceding platoon vehicle`
FolID = $\emptyset$; `// Index of the following platoon vehicle`
while *CACCtransition = false, k++* **do**
  $[\mathcal{C}_k, \vec{x}_{k,i} \forall i \in \mathcal{C}_k]$ = detectVehicles;
  if $(FolID \neq \emptyset \ \& \ \mathcal{C}_k \neq \mathcal{C}_{k-1})$ *OR* $PreID \notin \mathcal{C}_k$ **then**
    [FolID,PreID] = SequenceManager($\vec{x}_{k,new}, \vec{x}_{k,i} \forall i \in \mathcal{C}_k$);
  end
  if $PreID \neq \emptyset$ **then**
    $\mathbf{X}_{k,new}$ = CooperativeController($\vec{x}_{k,new}$,$\vec{x}_{k,pre}$);
    if $|\vec{a}_{k,new}| > \tau_a$ *OR* $|\vec{j}_{k,new}| > \tau_j$ **then**
      [FolID,PreID] = ResequenceManager($\vec{x}_{k,new}, \vec{x}_{k,i} \forall i \in \mathcal{C}_k$);
      $\mathbf{X}_{k,new}$ = CooperativeController($\vec{x}_{k,new}$,$\vec{x}_{k,pre}$);
    end
  else
    $\mathbf{X}_{k,new}$ = IndividualController($\vec{x}_{k,new}$);
  end
end

5.4.2 Last Vehicle Optimization

For long platoons, not all platoon vehicles are in cooperation set $\mathcal{C}_k$, the proposed sequencing algorithm uses this fact. The algorithm aims to minimize the peak acceleration of the last vehicle in cooperation set $\mathcal{C}_k$. The importance of the last vehicle is because its trajectory influences all subsequent platoon vehicles outside $\mathcal{C}_k$. This algorithm may thus yield a sequence that minimizes the impact on platoon vehicles outside of $\mathcal{C}_k$.

A general description of the algorithm is given in *Algorithm 5.2*. The algorithm starts with predicting the future states of the lead vehicle $\mathbf{X}_{lead}$, which consists of positions $\vec{q}_{lead}$, velocities $\vec{v}_{lead}$, accelerations $\vec{a}_{lead}$, and jerks $\vec{j}_{lead}$. This is done using its current states $\vec{x}_{lead}$ and an expected future desired acceleration of 0. Then the states of the new vehicle $\mathbf{X}_{new}$ are predicted for merging directly behind the lead vehicle. If the predicted accelerations $\vec{a}_{new}$ and jerks $\vec{j}_{new}$ are within the user-specified bounds τ_a and τ_j, the future states of the following vehicle are predicted and analyzed. If its accelerations and jerks are within the bounds, a series of simulations using the linear state-space model of Chapter 3 is used to predict the behavior of subsequent vehicles. The state equations of this model are,

$$\dot{\vec{x}}_i = \begin{bmatrix} 0 & -1 & -h & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} & 0 & 0 \\ \frac{k_p}{h} & -\frac{k_d}{h} & -k_d & -\frac{1}{h} & -\frac{k_d}{h} & -\frac{1}{h} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \vec{x}_i + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\frac{\tau}{h} \\ 0 \\ 1 \end{bmatrix} \dddot{\gamma}_i, \tag{5.29}$$

where $\vec{x}_i = \begin{bmatrix} e_i, & v_i - v_{i-1}, & a_i, & u_i, & \dot{\gamma}, & \ddot{\gamma} \end{bmatrix}^\top$. Then the maximum absolute predicted acceleration of the last vehicle is added into a cost-vector $\vec{c}$. Note that when investigating the option of merging the new vehicle behind the platoon, its own maximum absolute predicted acceleration is added. The index with the smallest cost in $\vec{c}$ is selected as the optimal merging sequence.

For the resequencing manager, the cost for moving forward c_f, moving backward c_b, and keeping the current sequence c_c is determined using *Algorithm B.4*. The sequence is altered to move the new vehicle forward if $c_f w_{opt} < c_c$ and $c_f < c_b$. Similarly, the new vehicle is moved backward if $c_b w_{opt} < c_c$ and $c_b \leq c_f$. Here, w_{opt} is a user-specified weight factor to prevent the sequence from being altered too rapidly and too often.

It is apparent that the prediction of future states required for this method is computationally heavy. Since the behavior of the vehicles can be approximated using the state-space representation of (5.29), the simulations are relatively computationally inexpensive. However, the distance- and time-based methods presented in the following sections require less computation.

Algorithm 5.2: Last vehicle optimization sequence manager.

Result: PreID, FolID

PreID = $\emptyset$; // Index of the preceding platoon vehicle
FolID = $\emptyset$; // Index of the following platoon vehicle
leadID = $\min(\mathcal{C}_k)$; // Determine platoon leader
$\mathbf{X}_{lead} = \text{PredictLeadStates}(\vec{x}_{lead})$;
foreach $i \in \mathcal{C}_k$ **do**
  $\mathbf{X}_{new} = \text{MergeBehind}(\mathbf{X}_i, \vec{x}_{new})$;
  if $|\vec{a}_{new}| < \tau_a$ & $|\vec{j}_{new}| < \tau_j$ & $i \neq \max(\mathcal{C}_k)$ **then**
    $\mathbf{X}_{i+1} = \text{GapOpeningPredict}(\mathbf{X}_i, \vec{x}_{i+1})$;
    if $|\vec{a}_{i+1}| < \tau_a$ & $|\vec{j}_{i+1}| < \tau_j$ & $i+1 < \max(\mathcal{C}_k)$ **then**
      foreach $j \in \mathcal{C}_k$ & $j > i+1$ **do**
        $\mathbf{X}_j = \text{lsim}(\mathbf{X}_{j\text{-}1}, \vec{x}_j)$;
        if $j = \max(\mathcal{C}_k)$ **then**
          $\vec{c}(i) = \max(|\vec{a}_j|)$;
        end
      end
    else if $|\vec{a}_{i+1}| < \tau_a$ & $|\vec{j}_{i+1}| < \tau_j$ **then**
      $\vec{c}(i) = \max(|\vec{a}_{i+1}|)$;
    else
      $\vec{c}(i) = \infty$;
    end
    $\mathbf{X}_{i+1} = \text{lsim}(\mathbf{X}_i, \vec{x}_{i+1})$;
  else if $|\vec{a}_{new}| < \tau_a$ & $|\vec{j}_{new}| < \tau_j$ **then**
    $\vec{c}(i) = \max(|\vec{a}_{new}|)$;
  else
    $\vec{c}(i) = \infty$;
  end
end
if $\exists \vec{c}(i) \neq \infty$ **then**
  PreID= $\min_i(\vec{c}(i))$;
  if *PreID* $\neq \max(\mathcal{C}_k)$ **then**
    FolID = PreID+1 ;
  end
end

5.4.3 Distance-based Sequencing

The distance-based sequencing method is relatively straightforward. The new vehicle compares its one-dimensional position coordinate q_{new} to that of the vehicles in $\mathcal{C}_k$. The resulting sequence is then such that $q_{pre} \leq q_{new} < q_{fol}$, where q_{pre} and q_{fol} are the positions of the selected preceding and following vehicles respectively. If q_{new} is smaller than the position of the last vehicle in $\mathcal{C}_k$, the new vehicle checks if it is within one inter-vehicle distance behind this vehicle. If this is true, the new vehicle merges behind the platoon. Otherwise, the new vehicle continues to use its individual controller. The algorithm is explained in *Algorithm B.1.*

The resequencing algorithm compares the position of the new vehicle to that of the preceding and following vehicles. If $q_{new} > q_{pre} + \tau_{dist}$ the vehicle is moved one position forward. If $q_{new} < q_{fol} - \tau_{dist}$ the vehicle is moved one position backward. Here, τ_{dist} is a user-specified threshold that prevents too frequent switching of the sequence. The algorithm is shown in *Algorithm B.5.*

5.4.4 Time-based Sequencing

The time-based sequencing algorithm orders the vehicles based on their Estimated Time of Arrival (ETA) at q_{mp}. To determine this, the most forward vehicle in $\mathcal{C}_k$ is assumed to be the platoon leader and the states of the subsequent vehicles are based on the desired steady-state platoon formation. The velocity of subsequent vehicles near q_{mp} is thus assumed to be equal to the velocity of the lead vehicle v_{lead}. Time $t_{lead,mp}$ at which the lead vehicle reaches q_{mp} can be predicted using measured or transmitted information. For a subsequent platoon vehicle i, time $t_{i,mp}$ at which it reaches q_{mp} can then be predicted using the desired spacing policy of steady-state platoon driving. The knowledge that vehicle i intends to bring the position error to zero provides additional insight into its future behavior.

For the new vehicle, the intended ETA can be determined using t_f^* based on the equations in Section 5.3. Its final desired velocity is v_{lead} since the vehicle aims to drive in a steady-state platoon upon entering the main lane. The trajectory to determine t_f^* goes up to the intended start of the lane change. The final position is determined using merging point q_{mp} and the time required for the lane change τ_{lc}. The new vehicle thus intends to reach q_{mp} at time $t_{new,mp} = t_f^* + \tau_{lc}$.

The algorithm places the new vehicle ahead of the first vehicle i for which $t_{new,mp} < t_{i,mp}$ holds. If this inequality does not hold for the last vehicle in $\mathcal{C}_k$, the new vehicle predicts time $t_{behind,mp}$ at which it reaches q_{mp} if it merges behind the last vehicle. This time instance is calculated using

$$t_{behind,mp} = t_{last,mp} + (r_{new} + L_{new}) / v_{lead} + h_{new}. \tag{5.30}$$

If $t_{new,mp} < t_{behind,mp}$ the vehicle merges behind the platoon. If this inequality does not hold the vehicle continues using the individual controller. This check

provides the opportunity to merge behind the platoon while preventing unreasonable sequences to be selected. A more detailed overview using pseudocode is given in *Algorithm B.2*.

The time-based resequencing manager uses a similar philosophy as the sequence manager. It requires times $t_{pre,mp}$ and $t_{fol,mp}$ of the preceding and the following vehicle respectively. These times are determined based on the platoon leader and a steady-state platoon as if the new vehicle does not join. Furthermore, the t_f^* of the new vehicle is used, which is determined without considering the gap in the platoon. If $t_f^* + \tau_{lc} < t_{pre,mp} - \tau_{time}$ the algorithm checks if it is feasible to move the new vehicle forward. Similarly, if $t_f^* + \tau_{lc} > t_{fol,mp} + \tau_{time}$ the algorithm checks if it is feasible to move the new vehicle backward. If a feasibility check is passed, the corresponding action is executed. τ_{time} denotes a user-specified threshold that is introduced to prevent switching too frequently. If τ_{time} is too high, resequencing will hardly occur. If τ_{time} is low, the sequence will be changed too frequently due to measurement noise affecting the calculations of the ETAs. A detailed explanation is given in *Algorithm B.6*.

5.4.5 Conventional Optimization

The conventional optimization algorithm is designed similarly to the last vehicle optimization algorithm in Section 5.4.2. The difference is in the cost criterion. The conventional algorithm uses the sum of the RMS of the accelerations of all vehicles in cooperation set $\mathcal{C}_k$ and the new vehicle as cost criterion. This criterion is similar to that in Jing et al. (2019). The predicted accelerations are dependent on the longitudinal vehicle model and the applied merging control strategy. Therefore, existing methods need small adjustments to be implemented in combination with the novel merging strategy of Chapters 3 and 4. The resulting algorithm is summarized in *Algorithm B.3*.

In short, the trajectories of all vehicles are predicted for every possible merging sequence. Then the RMS values of their accelerations up to reaching q_{mp} can be computed. For each feasible sequence, the sum of all RMS values is used as its cost. The feasible sequence with the lowest cost is selected as the proposed sequence. If none of the sequences is feasible, no sequence is selected and the new vehicle continues to drive using the individual controller.

For the resequencing manager, the cost for moving forward c_f, moving backward c_b, and keeping the current sequence c_c is determined using *Algorithm B.7*. Similar to the last vehicle optimization algorithm, the sequence is altered to move the new vehicle forward if $c_f w_{opt} < c_c$ and $c_f < c_b$. The new vehicle is moved backward in the sequence if $c_b w_{opt} < c_c$ and $c_b \leq c_f$. Here, w_{opt} is a user-specified weight factor to prevent too frequent switching between sequences.

This concludes the explanation of the merging sequencing methods. The following section provides simulations using these four methods. These simulations provide a better understanding of the performance of the different methods.

5.5 Analysis of the Sequence Management Algorithms

Based on the designs of the different sequencing methods some strengths and weaknesses can be expected. To get a better understanding of their behavior, simulations are performed. This section discusses the simulations and analyzes their results. First, the simulation setup is introduced, this includes important parameters such as sensor noise and communication delay. Then the simulation scenarios and selected performance indicators are discussed. Finally, the simulation results are presented and analyzed.

5.5.1 Simulation Setup

The simulation setup is important to consider when analyzing the results. This does not just include technical aspects of the simulations but also the chosen parameters. This section will provide such information for all simulations. The simulations are performed using MATLAB/Simulink R2020a. The simulations are performed in Simulink using a fixed-step ODE3 solver, running at 100 Hz.

The simulations regard a single vehicle merging into an existing platoon of eleven vehicles. In these simulations, all vehicles are parameterized equally. The parameters can be found in Table 5.1. The dynamics of the vehicles are modeled using,

$$\dot{q}_i(t) = v_i(t) \tag{5.31}$$
$$\dot{v}_i(t) = a_i(t) \tag{5.32}$$
$$\dot{a}_i(t) = \frac{1}{\tau}u_i(t) - \frac{1}{\tau}a_i(t). \tag{5.33}$$

Here, τ is a time-constant that represents the driveline dynamics, and control input u_i is the desired acceleration. Additionally, a saturation of a_{min} and a_{max} is applied to the acceleration to avoid infeasible behavior. This model is based on a non-linear third-order model. By choosing an appropriate control law structure using the aerodynamic drag coefficient and the mechanical drag, input-output linearization is achieved (Ploeg et al., 2011; Sheikholeslam and Desoer, 1993; Stankovic et al., 2000). It can be noted that by adding the driveline dynamics this simulation model is different from the simplified model of (5.1)-(5.3). The reason for this is that the simplified model is used to quickly perform the optimization and create a reference trajectory. The simulation model is used to examine and demonstrate the behavior of the strategies. Therefore, the simulation model aims to mimic a passenger vehicle more accurately by introducing driveline dynamics and a saturation on the acceleration. If vehicle specifications are desired and known, a more sophisticated model can be used that considers aspects such as the maximum wheel power. However, for the purpose of a general investigation into the behavior of the sequencing methods, this is not deemed necessary. Manual checks are performed to ensure that the resulting trajectories

Table 5.1: Simulation parameterization.

Vehicle and CACC parameters							
τ	a_{min}	a_{max}	L	r	h	k_p	k_d
0.1 s	-3 m/s^2	2 m/s^2	5 m	2 m	0.5 s	0.2	0.7

Controller transition parameters				
$t_{i,min}$	$t_{i,min}$	τ_a	τ_j	$\gamma_{i,min}$
1 s	5 s	$\pm$0.8 m/s^2	$\pm$1.2 m/s^3	-0.1 m

Sequencing and resequencing parameters			
w_{ind}	w_{opt}	τ_{dist}	τ_{time}
0.01	1.15	$0.5(L_{new} + r_{new} + h_{new} v_{lead})$ m	$0.1\left(\frac{L_{new}+r_{new}}{v_{lead}} + h_{new}\right)$ s

Table 5.2: Standard deviation values of the sensor noise used in the simulations.

Signal	Radar position	Radar velocity	On-board velocity	On-board acceleration
Standard Deviation	20.9 cm	0.141 m/s	0.048 m/s	0.20 m/s^2

are within a feasible range for passenger vehicles. An example of such a check is an investigation into the velocities which is included for every simulation.

For the controller transition, the parameters of the new and following vehicles are equal. Thresholds τ_a and τ_j are equal for the controller transition and sequencing algorithm. In the sequence recalculation algorithms, thresholds τ_{dist} and τ_{time} are dependent on the velocity of the platoon leader, which helps to operate at different velocities.

Multiple scenarios are used in the simulations, considering different initial conditions and behavior of the platoon leader. However, there are some commonalities in all scenarios. The platoon leader is placed 500 meters before q_{mp}. The platoon is driving at 80 km/h and subsequent platoon vehicles are initialized such that a steady-state platoon is formed. Unless otherwise specified, the new vehicle is initialized 75 meters behind the platoon leader and at 65% of the platoon velocity. Its initial acceleration is set to 1 m/s^2.

To analyze the effect of noise on the algorithms, sensor noise is added to the simulations. White noise is added to the position and velocity measurements of the radar and the velocity and acceleration measurements of the on-board sensors. The standard deviations can be found in Table 5.2. These values are based on experiments with a demonstrator platform (Schinkel et al., 2021). The noise is only applied to measurement data and not in the dynamics of the vehicle model such as the mapping from desired to actual acceleration in (5.33).

One important part of the simulations is modeling the communication range. The set of vehicles in the communication range is dependent on the shape of the

on-ramp. In these simulations, the maximum distance between the new vehicle and the platoon vehicle i ($d_{max,i}$) is defined as

$$d_{max,i} = \begin{cases} |q_{mp} - q_{new} - L_{al} + |q_{mp} - q_i - L_{al}|| \,, & q_{mp} - q_{new} \geq L_{al} \\ |q_{new} - q_i| \,, & \text{otherwise.} \end{cases} \tag{5.34}$$

Where L_{al} is the length of the acceleration lane. In other words, if the new vehicle has not yet reached the acceleration lane, $d_{max,i}$ is the summation of both vehicles' distances to the start of the acceleration lane. Otherwise, $d_{max,i}$ can be determined using the one-dimensional distances. Using a user-specified communication range $d_{\mathcal{C}}$, vehicle i is in cooperation set $\mathcal{C}_k$ if $d_{max,i} \leq d_{\mathcal{C}}$.

For these simulations, the communication range $d_{\mathcal{C}}$ is set to 350 meters. This value is based on experiments with the demonstrator platform where stable communication could be established over this distance. Based on a typical highway environment, L_{al} is set to be 300 meters.

To complete the model of the communication network, a time delay of 0.02 seconds was added to all communicated messages (Hoogeboom, 2020). The communication is sampled at 25 Hz with a zero-order hold. The communication range, delay, and frequency are included in the simulations. However, there are other aspects of communication that have not been considered such as packet dropouts and variable time delays. These aspects can be included in future work either in simulations or experiments.

5.5.2 Simulation Results

This section presents and analyzes the simulation results. Two scenarios are presented here, which are:

- **Constant Velocity**; During these simulations, the platoon leader drives a constant velocity. The initial conditions of the new vehicle are varied. Furthermore, the simulations are repeated with different noise signals. The aim of this scenario is to provide a general impression of the performance of the different sequencing algorithms.
- **Decelerating Leader**; The vehicle leader decelerates during the maneuver. This additional challenge possibly requires recalculation of the merging sequence. This is an important scenario because it indicates how the algorithm behaves if the platoon needs to make a sudden braking action which can compromise safety. This scenario has been simulated with multiple leading vehicle trajectories and noise signals.

For further analysis, an acceleration scenario is presented in Appendix B.3. This scenario is similar to the deceleration scenario, but the platoon leader accelerates during the maneuver, resulting in different behavior of the new vehicle. In essence, it is intuitively expected that with a deceleration action the preferred

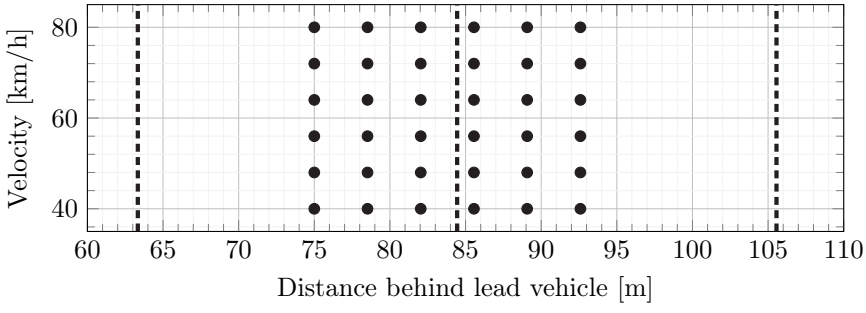


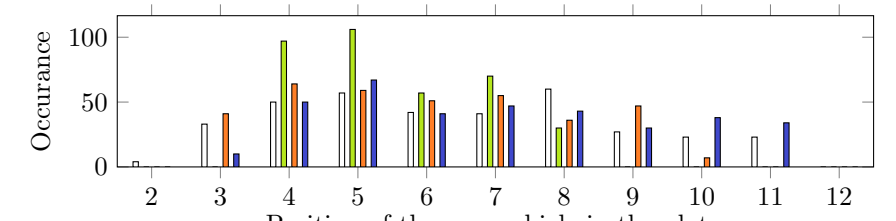
Figure 5.4: The initial conditions of the new vehicle (•) and the initial positions of the fourth, fifth, and sixth platoon vehicles (----).

merging slot moves forward and with an acceleration action it moves backward. Furthermore, Appendix B.4 presents a scenario in which the velocity of the lead vehicle is constantly changed by putting sinusoidal trajectories on its desired acceleration. Since the velocity changes differ from the predicted behavior, they affect the performance. An important aspect is the behavior of the resequencing algorithm. The sequence might need to be adjusted to ensure a safe merging maneuver. However, the sequence should not be altered too often because this will introduce additional dynamics which can create unsafe situations. The insight gained from these additional scenarios is limited and the most important conclusions can be obtained without the appendixes.

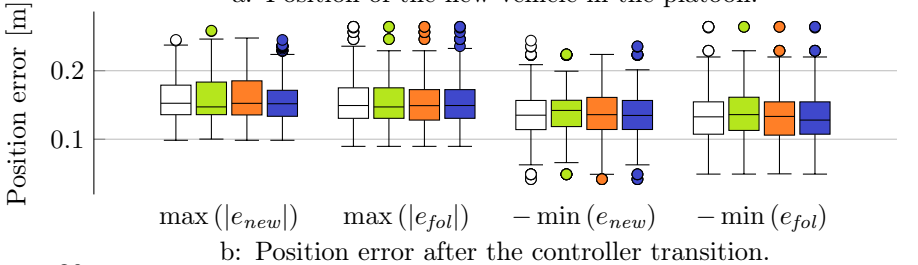
Constant Velocity

To gain a general understanding of the behavior of the different sequencing methods, simulations are performed prescribing a constant velocity of the platoon leader. The initial position and velocity of the new vehicle are varied. The new vehicle is first initialized 75 meters behind the leading vehicle. Subsequently, the starting position is moved backward five times, each time by one-sixth of a vehicle position difference. This results in a total of six starting positions with different distances to the nearest vehicle. The initial velocity of the new vehicle is varied between 50% and 100% that of the platoon leader with increments of 10%. A complete overview of the initial conditions is given in Figure 5.4. Furthermore, each combination of initial position and velocity is simulated 10 times with different noise signals. The standard deviations of Table 5.2 are used in every simulation.

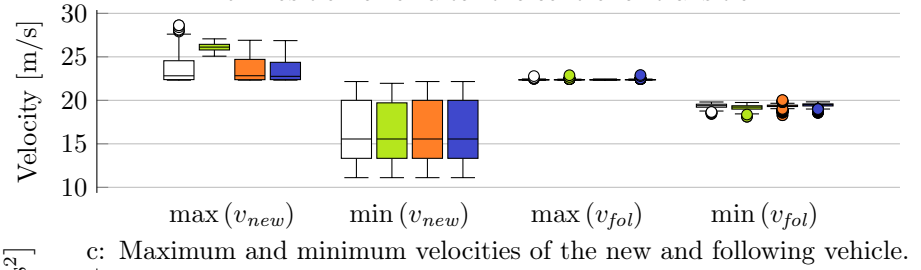
The positions of the new vehicle in the platoon are shown in Figure 5.5a. The distance-based method has the least spread in determining the position. The method primarily selects positions 4 and 5, which are close to the new vehicle's initial position, as shown in Figure 5.4. The time-based method has



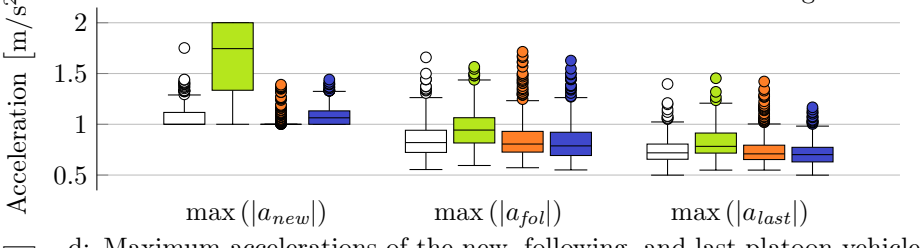
a: Position of the new vehicle in the platoon.



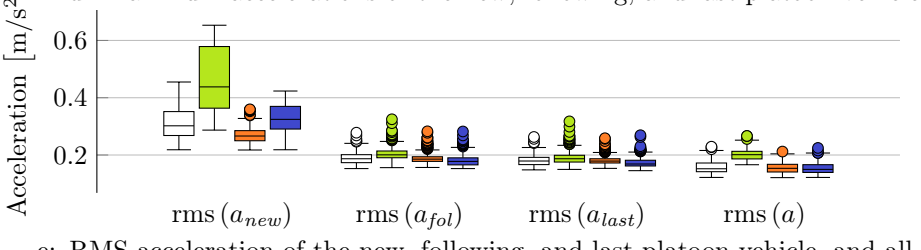
b: Position error after the controller transition.



c: Maximum and minimum velocities of the new and following vehicle.



d: Maximum accelerations of the new, following, and last platoon vehicle.



e: RMS acceleration of the new, following, and last platoon vehicle, and all vehicles.

Figure 5.5: The results for 360 simulation parameterizations using the constant velocity scenario for the last vehicle optimization (□), distance-based (■), time-based (■), and conventional optimization (■) sequencing methods.

a larger spread in position selection. However, the proposed and conventional optimization-based methods have the largest spread in the proposed sequences. Moreover, it appears that the time-based method selects positions further forward than the optimization-based strategies. Fewer platoon vehicles are affected when the new vehicle merges further back. In addition, the gap-opening platoon vehicle gets more time to open the gap and consequently requires lower accelerations. Hence, this behavior can be explained by the proposed optimization criteria.

The maximum absolute and minimum position errors are shown in Figure 5.5b. The results are plotted using standard boxplots. Fifty percent of the data falls within the rectangles where the horizontal line indicates the median value. The whiskers show the spread of the remaining fifty percent of the data. The maximum length of the whisker is 1.5 times the length of the rectangle, often referred to as the interquartile range. Any data points that falls outside of the maximum length of the whiskers are plotted as outliers. The peak errors remain below 0.3 meters. The minimum errors are important because a negative error means that the vehicle is too close to its predecessor. These also remain below 0.3 meters indicating a safe execution. From this, it can be concluded that the previously proposed merging algorithm performs well in all simulations. As a result, none of the sequencing methods produces a dangerous situation. Consequently, there seems to be no advantage of choosing one sequencing method over the other from a safety point of view.

No bounds are imposed on the predicted velocities. Nonetheless, in Figure 5.5c it is shown that the velocities remain in a feasible range for highway merging. On average, the maximum velocity of the new vehicle is the highest when the distance-based algorithm is used. This is due to the forward position of the new vehicle in the platoon. However, the proposed sequencing algorithm causes the highest outliers, which can be explained by its outliers in forward position as shown in Figure 5.5a. Overall, the effect of the sequencing algorithms on the minimum and maximum velocities remains relatively small. The minimum velocities of the new vehicle are mainly determined by the initial states, which is why there is almost no difference in the results for the different algorithms.

The peak accelerations are summarized in Figure 5.5d. Most noticeable is that the maximum acceleration of the new vehicle is highest when using the distance-based method. The saturation of 2 $\mathrm{m/s^2}$ in the vehicle model is reached. This behavior is to be expected because when the new vehicle is initialized at a lower velocity, it needs high accelerations to reach a position close to its initial relative location. Essentially, the average speed of the new vehicle must match that of the platoon, requiring additional accelerations. The peak accelerations stay well below the saturation value of 2 $\mathrm{m/s^2}$ when the other three sequencing methods are used. This indicates that the proposed sequences result in achievable trajectories for all vehicles. The peak acceleration of the new vehicle is lowest when the time-based method is used. This is expected since t_f^* used in

the time-based method is the preferred merging time for the new vehicle. Consequently, the resulting series reflects the interests of the new vehicle. Surprisingly, the peak accelerations of the other vehicles using the time-based method are similar to those using the optimization-based methods. This indicates the potential of the time-based method. The conventional optimization-based method results in lower outliers than the last vehicle optimization method. However, there appears to be little difference between the two optimization-based methods.

Finally, the RMS values of the accelerations are presented in Figure 5.5e. The results are similar to those of the peak accelerations. The distance-based method results in significantly higher accelerations than the other methods. The time-based method results in lower accelerations for the new vehicle than the optimization-based methods. However, the accelerations of the other vehicles are similar for these three methods. The last vehicle optimization method does not reduce the accelerations of the last vehicle or that of all participants compared to the conventional optimization-based method. This is likely due to the sensor noise that strongly influences trajectory predictions.

To conclude, the merging algorithm can handle the behavior of each of the presented sequencing methods. Based on the position errors, it can be stated that the safety was not compromised in any simulation. Based on the accelerations and thus the efficiency of the maneuvers, the time- and optimization-based methods seem more favorable than the distance-based method. However, it is impossible to draw definitive conclusions on the best method among them based on this set of simulations.

Decelerating Leader

This scenario is used to investigate the outcome of the merging sequencing methods when the platoon leader temporarily performs a severe braking action. The vehicle brakes with a maximum deceleration of -3 m/s^2. The simulation is repeated with ten different trajectories of the platoon leader. These trajectories shift the braking by one second. The two of the ten velocity and acceleration trajectories of the platoon leader are shown in Figure 5.6. Each trajectory of the platoon leader is simulated ten times with different noise signals.

An overview of the simulation results is given in Figure 5.7. The selected positions of the new vehicle are shown in Figure 5.7a. The spread in the sequences during these simulations is smaller than that for the constant velocity scenario. This can be explained by the limited initial conditions. The distance- and time-based methods place the new vehicle further forward than the optimization-based methods. This is to be expected and similar to the constant velocity scenario.

Figure 5.7b shows that despite the severe braking action, the position errors remain small for all sequencing methods. This indicates that the maneuver is performed safely regardless of the method used.

The velocities in Figure 5.7c generally remain between 5 and 25 m/s. The

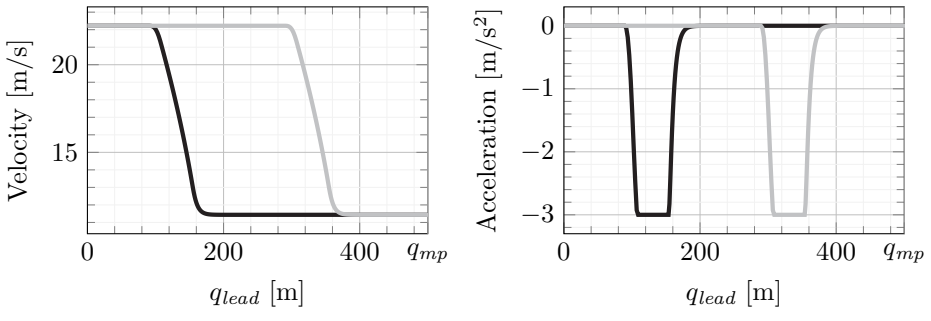


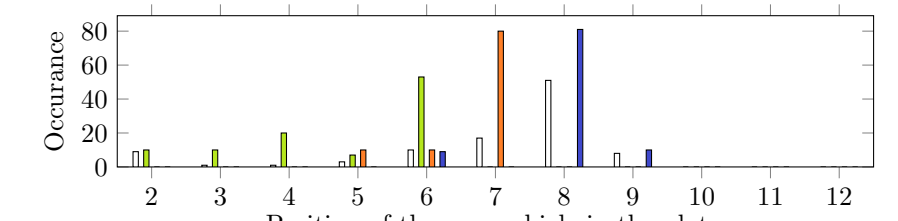
Figure 5.6: The velocity and acceleration trajectories of platoon leader during the simulations with the earliest (—) and latest (—) braking action in the decelerating leader scenario.

exception is the maximum velocity of the new vehicle which exceeds 25 m/s when the distance-based algorithm is used. This is caused by the forward position selected by this algorithm. Furthermore, the small spread in maximum velocities indicates the relative late resequencing performed by this strategy.

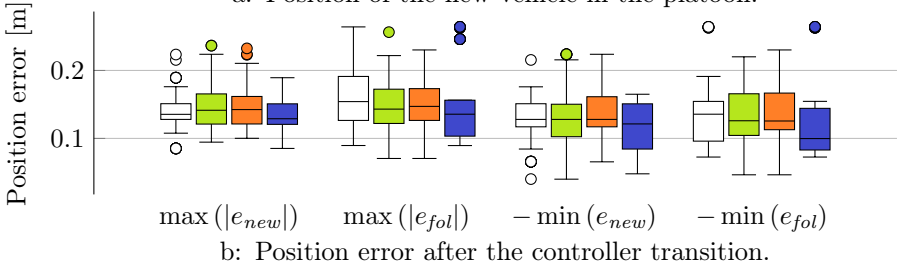
The peak accelerations in Figure 5.7d are strongly influenced by the saturation on the vehicle model and the excitations of the lead vehicle. In general, the optimization-based strategies appear to result in sequences that are more favorable to the new vehicle. The advantage for the new vehicle can be explained by its position further back in the platoon. The new vehicle has more time to adjust its velocity, and lower accelerations are required. The distance- and time-based methods result in sequences that are more favorable to the gap-opening and subsequent platoon vehicles. These sequencing methods select a position further forward. Therefore, the gap-opening action is likely to be performed more rapidly, causing the gap-opening vehicle to have a lower velocity when the platoon leader starts braking. Therefore, a lower additional deceleration is required to react to the platoon leader.

Similar behavior is found for the RMS values in Figure 5.7e. The main difference is the performance of the distance-based method. Its value for the new vehicle is larger than that of any other method. Moreover, the values for the other vehicles are closer together for the four different methods. There is no clear ranking possible with respect to these performance indicators.

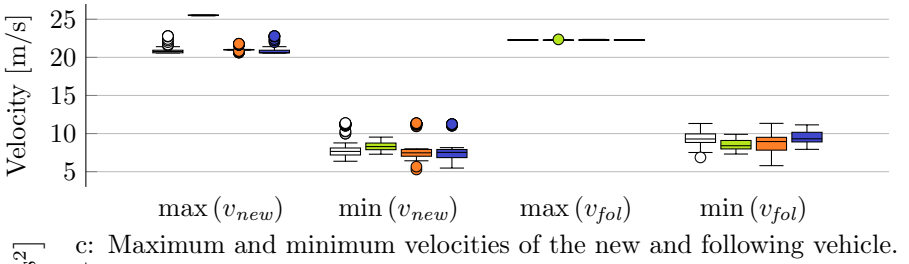
These simulations show the performance of the sequencing methods for a scenario in which the platoon leader performs a severe braking action. Combined with the results from the constant velocity scenario, it can be concluded that these results are highly dependent on the exact situation. Therefore, it is not possible to select a preferred sequencing method purely on these simulations. The main conclusion is that a sufficiently robust merging control strategy is required to ensure the safe execution of the merge regardless of the sequenc-



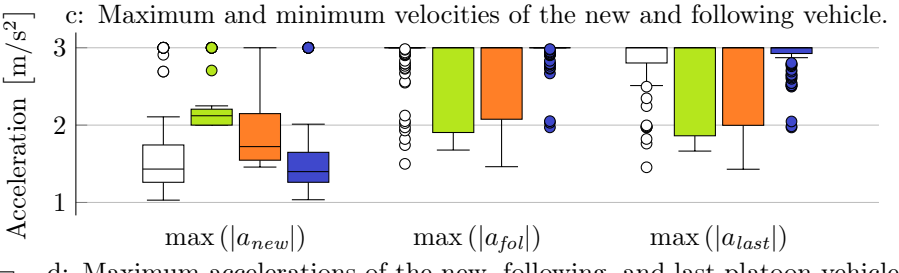
a: Position of the new vehicle in the platoon.



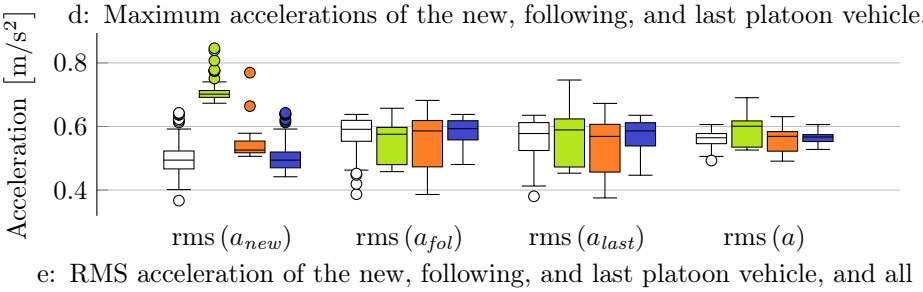
b: Position error after the controller transition.



c: Maximum and minimum velocities of the new and following vehicle.



d: Maximum accelerations of the new, following, and last platoon vehicle.



e: RMS acceleration of the new, following, and last platoon vehicle, and all vehicles.

Figure 5.7: The results for 100 simulation parameterizations using the decelerating leader scenario for the last vehicle optimization (□), distance-based (■), time-based (■), and conventional optimization (■) sequencing methods.

ing strategy. Moreover, the time- and optimization-based methods both have potential.

5.6 Conclusion and Discussion

This chapter proposes a sequencing strategy for the merging at highway on-ramps. The proposed optimization-based strategy aims to minimize the acceleration of the last platoon vehicle within the communication range of the new vehicle. Theoretically, this has the potential to minimize the accelerations of all subsequent platoon vehicles outside of the communication range. The performance is analyzed using three different benchmark strategies. These can be categorized as a distance-based, a time-based, and an optimization-based strategy. Furthermore, these benchmarks are compared to each other to obtain a general understanding of the advantages and disadvantages of different merging sequence strategies. To perform the analysis, a wide range of simulations are performed. Initial conditions, leading vehicle excitations, and noise signals are varied.

The simulations show the influence of the sequencing strategies on the execution of the merging maneuver. The merging maneuver designed in the previous chapters can cope with all sequencing strategies and the resulting position errors remain small. However, there is a performance difference in terms of accelerations during the maneuver. Lower accelerations indicate that the maneuver is executed more efficiently. As expected, the distance-based strategy causes high accelerations when the new vehicle is initialized at a lower speed than the platoon. A large average velocity is then required to reach its desired position. The difference in performance in these simulations between the time-based and two optimization-based strategies is small. The selected positions vary, but the resulting accelerations remained within a similar range. The advantage of one strategy over others is dependent on the exact scenario. Finally, the proposed last vehicle optimization strategy does not outperform a conventional optimization strategy. This may be caused by the sensor noise which causes uncertainty in the predictions used for the optimization.

Apart from the simulation results, some conclusions can be made from a general analysis of the strategies. An important conclusion is that the time-based strategy is easily implementable and has a performance comparable to the optimization-based strategies. It is easier to implement than the optimization-based strategies because it is less dependent on the expected behavior of the vehicles and thus their model and control strategy. Often, literature uses a distance-based strategy as a benchmark when investigating new sequencing methods. However, due to the performance of the time-based strategy, the author recommends using a time-based sequencing strategy as a benchmark in future research.

Another important aspect that cannot be investigated using the simulations is the willingness of Original Equipment Manufacturers (OEMs) to implement

the policy. The optimization-based policies are dependent on the predicted trajectories. For a shared solution between multiple OEMs, an agreement on the desired trajectories and performance metrics is therefore required. Furthermore, the gap-opening platoon vehicle needs to leave its optimal position in the platoon to facilitate the maneuver. This gives an incentive for manufacturers to trick the system and maintain their position in the platoon. Some literature propose methods in which platoon vehicles communicate their own cost of opening a gap. This cost can be manipulated to lower the chances of needing to open the gap. The information of platoon vehicles required for distance-based and time-based policies can more easily be verified. Time-based methods leave the opportunity for the new vehicle to manipulate its desired ETA. However, since this method is focused on the needs of the new vehicle, there is less incentive to manipulate such data. Therefore, distance-based or time-based policies can be beneficial from an OEM standpoint.

The simulations presented in this chapter and the corresponding appendix give an insight into the influence of the different sequencing algorithms on the existing platoon. It furthermore demonstrated the performance of the previously designed merging control strategy in a wide range of circumstances. However, an important performance indicator of a sequencing strategy is its effect on the traffic flow. A more detailed insight into this topic can be obtained using longer flow simulations in an on-ramp environment. Furthermore, experiments with a physical setup can be beneficial to further investigate the influence of outside disturbances on the sequencing strategies. For experiments, a large number of vehicles is required such that multiple sequences exist. A larger number of sequences allows for a wider range in initial conditions and disturbances to be investigated. In essence, if a new vehicle merges into an existing platoon of two vehicles, the set of conditions to ensure a sequence is selected is much smaller than if the platoon exists of six vehicles. To allow for enough vehicles in the experiments, they may be performed with scaled-down mobile robots. Such experiments are easier and cheaper to perform than those with full-scale passenger vehicles, but some details regarding driveline dynamics or sensor behavior may be less accurate. It is recommended to include perturbations of the lead vehicle in the experiments, such as acceleration or braking actions.

Chapter 6

Experiments of the Longitudinal Merging Controller

A control algorithm for merging into a cooperative platoon is proposed in Chapters 3 and 4. In the current chapter, the performance of the longitudinal controllers is demonstrated using experiments with two full-scale vehicles. The aim of the experiments is to form a two-vehicle platoon before a predefined position is reached. Several scenarios with different behaviors of the preceding vehicle are investigated. In all experiments, the maneuver is successfully executed. The experiments demonstrate the performance of the merging vehicle. Its algorithm contains all individual components of the gap-opening vehicle's control algorithm. Due to the limited number of vehicles, the performance of the gap-opening vehicle is not demonstrated directly. However, the successful results of these experiments are an indicator of the performance of the gap-opening vehicle.

6.1 Introduction

The control strategy of Chapters 3 and 4 enables an individually driven vehicle to join a platoon of cooperative vehicles. The controllers are designed considering disturbances, such as noise and delays. Some disturbances have been added in simulations to validate the controllers. To further demonstrate their applicability, the controllers are implemented on an experimental setup. The setup exists of two full-scale passenger vehicles. In the experiments, a two-vehicle platoon is formed before a predetermined merging point is reached.

In Section 6.2 the methodology is discussed. This includes details about the experimental setup, implementation of the proposed controllers, and the environment. The experimental results are provided and analyzed in Section 6.3. Finally, the conclusions are given in Section 6.4.

6.2 Methodology

To test the designed approach, experiments were performed using two full-scale vehicles. The focus of the experiments is on the performance of the longitudinal controllers and the transitions in the control strategy. Since only two vehicles are available, the experiment cannot demonstrate a full merging maneuver, which requires at least three vehicles. Instead, the formation of a two-vehicle platoon is performed. In essence, a merging vehicle would like to form a platoon behind a preceding vehicle. A virtual line is used to represent the merging point. The steady-state platoon must be formed at a certain time period before this line is crossed. This time period is intended for the lane change maneuver.

6.2.1 Experimental Platform

The experiments are performed with two modified Renault Twizys. These are small two-person electric vehicles as shown in Figure 6.1. The vehicles are modified and used as the demonstrator platform of the i-Cave project (Hoogeboom, 2020; I-Cave, 2016). The Twizys have been equipped with multiple additional sensors, namely a front-facing radar (Bosch MRRevo14F Radar), Inertial Measurement Unit (IMU) (Bosch mm5.10 IMU (Bosch Motorsport)), and Global Navigation Satellite System (GNSS) receiver (u-blox EVK-M8T (u-blox, 2018)). Sensor fusion is performed on a real-time computer that is added for the control of the vehicles. This computer is also capable of actuating the complete vehicle. The throttle position signal from the accelerator pedal is replaced by a custom signal on the Controller Area Network (CAN) to actuate the electric motor. Furthermore, additional pressure can be applied to the brake master cylinder using a retrofitted electric motor and camshaft. Lastly, the steering column of the Twizy is replaced with a model which has electrical power steering. The motor in this column can be used to control the steering system. Both vehicles are

Figure 6.1: The vehicles used for the experiments.

equipped with a Vehicle-to-Everything (V2X) communication module (Rendits Router (Severinson)) to transmit and receive data. This enables the following vehicle to anticipate the future behavior of the preceding vehicle. A detailed description of the hardware and interfacing between is given in Hoogeboom (2020) and shown in Figure 6.2.

A real-time computer is used to interact with the additional hardware. Its task is to obtain information from the sensors, process this information, compute control commands and communicate these commands to the actuators. A general overview of the processes is given in Figure 6.3. Two software models regarding the perception are shown in blue; namely, the *host tracker* and the *target tracker*. The *host tracker* is used to determine the states of the host vehicle. In essence, a Kalman filter is used to fuse the information of the GNSS and IMU with the motor speed (Hoogeboom, 2020). This creates an estimation of the vehicle states at a frequency that is not limited by that of the GNSS signal. Furthermore, the redundancy in information can improve the quality of the estimated states compared to the raw data. The host vehicle states are combined with the radar measurement and communicated data in the *target tracker* module. Here the preceding vehicle's states are estimated using a Kalman filter. The cooperative controller module then computes the commands for the actuators using communicated data and the estimated states of the host and target vehi-

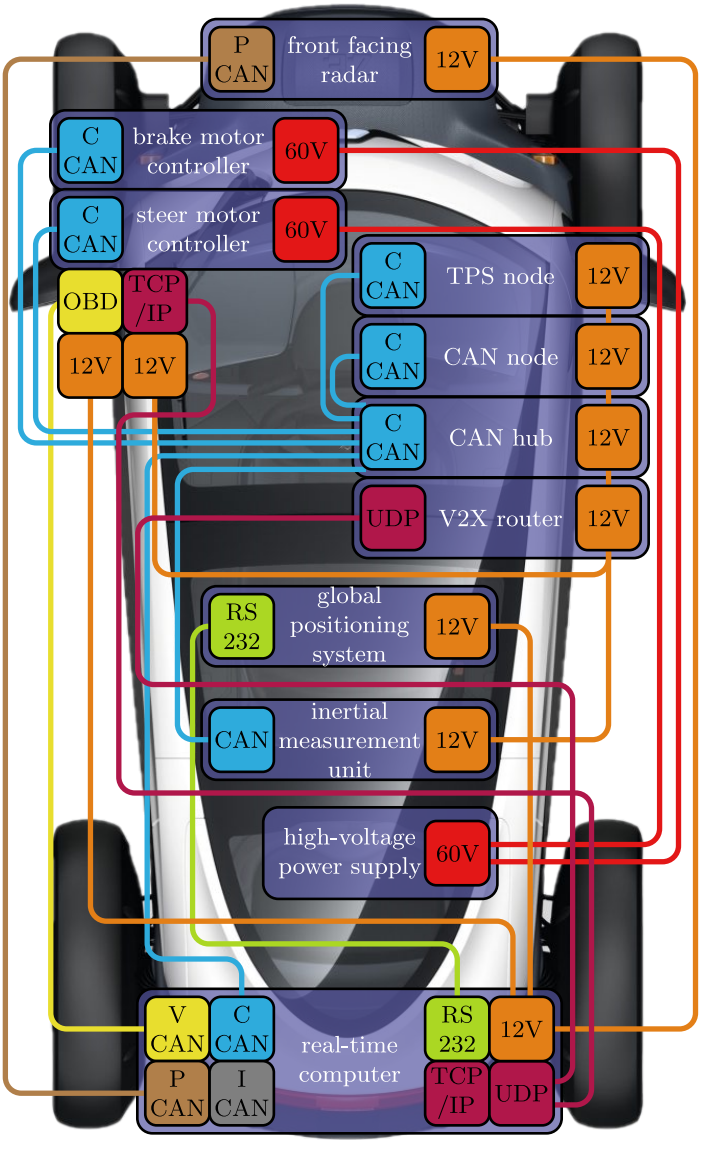


Figure 6.2: Overview of the additional hardware components of the Renault Twizy with their installed locations, power supply, and corresponding communication interfaces: vehicle CAN bus (—), control CAN bus (—), perception CAN bus (—), RS232 (—), ethernet (—) (Hoogeboom, 2020).

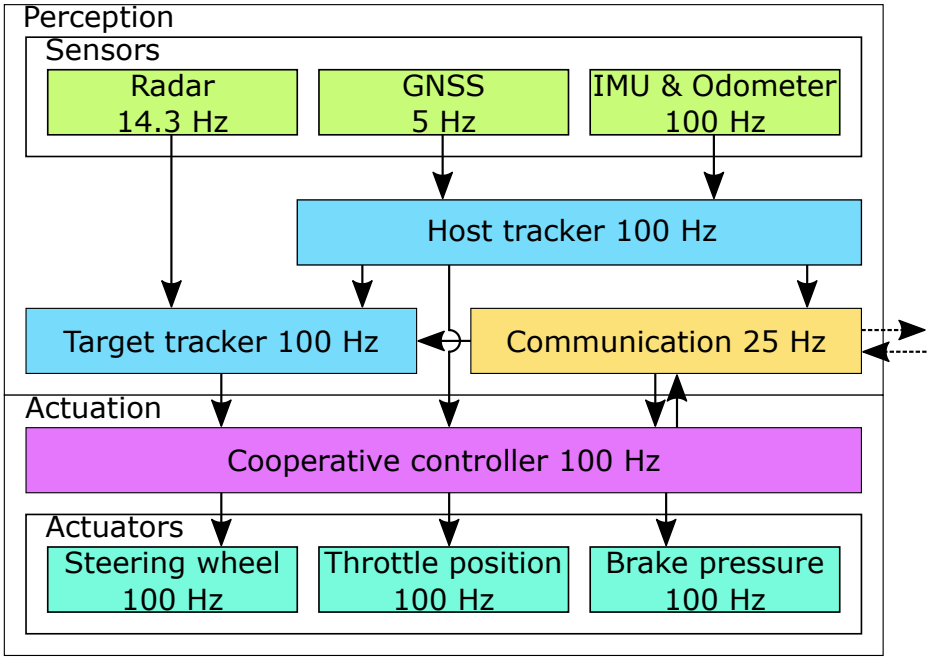


Figure 6.3: An overview of the processes and information flow in the automated driving system.

cles. For these experiments, only the throttle position is used, but the possibility to actuate the steering wheel and brake pressure exists in this setup.

For conventional Cooperative Adaptive Cruise Control (CACC) algorithms, the desired acceleration of the target vehicle's controller is used in the control law (Ploeg et al., 2011). In this experiment, the target vehicle is manually driven. Therefore, the desired acceleration cannot be obtained from a controller. A mapping based on the motor speed, and accelerator and brake pedal positions is used to determine the desired acceleration of the human driver. When the target vehicle broadcasts this desired acceleration, the conventional CACC algorithm can be executed on the following vehicle.

For these experiments the preceding vehicle will be driven manually, the other vehicle is using the proposed controller. The automated following vehicle is actuated using the accelerator pedal. Since the Renault Twizys are electric vehicles, they decelerate when the accelerator pedal is released due to the regenerative braking system. The brake pedal is therefore not required for longitudinal control if decelerations remain small. Lateral control of both vehicles is performed manually.

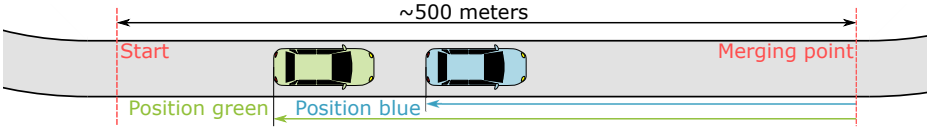


Figure 6.4: An overview of the processes and information flow in the automated driving system.

6.2.2 Experimental Environment

The experiments are performed at a straight stretch of road, a visualization of which is shown in Figure 6.4. A virtual line representing the merging point is placed on the road. The disadvantage of using a point is the chance of lateral misalignment due to GNSS inaccuracy. The vehicles compute their distance to the line which is used as their one-dimensional position coordinate. This process is performed by describing the environment using a local tangent plane coordinate system. The latitude, longitude, and altitude coordinates are transformed to north, east, down (NED) coordinates. These NED coordinates are expressed in meters making them ideal for the proposed computations. Then the virtual line is described using two points with coordinates (N_1, E_1) and (N_2, E_2). The position of the host vehicle is expressed using coordinates (N_h, E_h). Distance q_h is then expressed as

$$q_h = \frac{(N_2 - N_1)(E_1 - E_h) - (N_1 - N_h)(E_2 - E_1)}{\sqrt{(N_2 - N_1)^2 + (E_2 - E_1)^2}}. \tag{6.1}$$

It should be noted that the sign of q_h changes when the virtual line is crossed. If the absolute distance is used, q_h would decrease as the line is approached and increase after it is crossed. The coordinate would thus no longer represent the one-dimensional coordinate used for platooning. Furthermore, it is important the virtual line is perpendicular to the trajectory of the vehicles. If the line is under an angle, the expression $\dot{q}_h = v_h$ is invalid.

The merging controller is manually turned on at approximately 500 meters before the merging point, starting with the individual Model Predictive Control (MPC) controller. It is important to note that the road exists before and after this 500 meter stretch such that the vehicles can have an initial and terminal velocity. Since the controller is started manually some variation occurs between experiments. Furthermore, the initial distance between the vehicles is not controlled and may vary.

6.2.3 Controller Implementation

The MPC method of Section 4.3.3 is difficult to implement directly. Additional delays between the accelerator pedal position and motor speed cause problems

when the desired jerk is directly translated to a control input. Therefore, the MPC approach is combined with a conventional feedback controller. In essence, the MPC approach Section 4.3.3 is used to compute a reference trajectory existing of q_r, v_r, a_r, and j_r. Using these trajectories and the vehicle model presented in Section 2.1.2, the error dynamics are defined as

$$e_q = q_r - \hat{q} \tag{6.2}$$
$$e_v = v_r - \hat{v} \tag{6.3}$$
$$\dot{e}_v = a_r - \hat{a} \tag{6.4}$$
$$\ddot{e}_v = \frac{1}{\tau}\left(u_r - a_r + u - \hat{a}\right) = \frac{1}{\tau}\left(u_r + u - \dot{e}_v\right). \tag{6.5}$$

Here, the hat denotes that these are the estimated values from the sensor fusion module. Variable τ is the driveline constant and obtained through measurements with the vehicle (Hoogeboom, 2020). Furthermore,

$$u_r := a_r + \tau j_r. \tag{6.6}$$

Now by selecting the control input as

$$u = u_r + k_{p,r} e_q + k_{d,r} e_v \tag{6.7}$$

the error dynamics yield

$$\begin{bmatrix} \dot{e}_q \\ \dot{e}_v \\ \ddot{e}_v \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -\frac{k_{p,r}}{\tau} & -\frac{k_{d,r}}{\tau} & -\frac{1}{\tau} \end{bmatrix} \begin{bmatrix} e_q \\ e_v \\ \dot{e}_v \end{bmatrix}. \tag{6.8}$$

By defining appropriate control gains $k_{p,r}$ and $k_{d,r}$ such that $k_{p,r}, k_{d,r} > 0$ and $k_{d,r} > k_{p,r}\tau$ this system is asymptotically stable. During the experiments, the gains were defined as $k_{p,r} = 0.2$ and $k_{d,r} = 0.7$. These values are the same as those used in the CACC controller and have not been tuned specifically for this controller. For this vehicle, τ is estimated to be 0.06 seconds (Hoogeboom, 2020).

The reference trajectory computed by (4.19), (4.20), (4.21), and (4.22) is updated every 0.5 seconds. This provides a stabilizing controller for the individual driving phase of the vehicle. The conventional and transitional CACC controllers can remain unchanged during the experiments. These controllers and the switching between them are the main focus of this research. Therefore, the implementation of the individual controller does not negate the results obtained from these experiments.

6.2.4 Controller Tuning

The user has freedom in the tuning of the presented controllers. First, the variables of the CACC algorithm can be chosen. These are set to be $k_p = 0.2$,

$k_d = 0.7$, $h = 0.5$ seconds, and $r = 10$ meters. It should be noted that standstill distance r is relatively big, this is to give the fallback-ready user enough time to intervene if needed. The values of the controller gains were taken from Ploeg et al. (2011) and no specific tuning is performed. The aim of these experiments is to demonstrate a proof of concept. Therefore, specific tuning to optimize the performance of the vehicles is considered to be outside of the scope of the current work.

As described in Section 4.3.4, the controller transition can be tuned. An important parameter is the available time for the lane change. During the experiments, this time is set to 5 seconds. Consequently, the transition must be completed when the host vehicle estimates it will arrive at the merging point in 5 seconds. The position at which the lane change would be initiated is referred to as q_{lc}. Furthermore, the time for the controller transition can be bounded. During the experiments, the transition is performed in 5 seconds. The upper and lower bound are thus equal. This minimizes the number of trajectories checked which is computationally advantageous. The only restriction on the expected behavior for approval of the transition is that the absolute acceleration is less or equal to 1.5 $\mathrm{m/s^2}$.

6.3 Experimental Results

The performed experiments can be categorized using three scenarios describing the behavior of the preceding vehicle. These are, a constant velocity scenario, an acceleration scenario, and a decelerating scenario. The constant velocity experiment demonstrates a baseline behavior of the proposed algorithm. The acceleration and deceleration scenarios required the preceding vehicle the accelerate and decelerate respectively. This unexpected behavior creates additional challenges for the merging vehicle. To demonstrate the reproducibility of the experiments, three measurements for the acceleration scenario are compared. The automated vehicle forms a platoon in a safe and timely fashion for all performed experiments, demonstrating the proposed control strategy. The experimental results are presented in this section.

6.3.1 Constant Velocity Scenario

To create an understanding of the general behavior of the controller, experiments in which the driver of the preceding vehicle is instructed to drive at a constant velocity were performed. The following vehicle is placed at an unspecified distance behind the preceding vehicle using a cruise controller set to the same speed. When the straight road is reached (see Figure 6.4) and the velocity had settled, the proposed controller is introduced by replacing the cruise controller with the individual MPC-based controller of Section 5.3.

The position error during the experiment is shown in Figure 6.5a. The error in this figure is determined in post-processing using logged data from both vehicles which were synchronized using the GNSS time. The inter-vehicle distance in this analysis is calculated using the global positions. The data from the radar, although used in the real-time controller, is not used in this analysis. A comparison between the two error definitions is given in Appendix C. The error follows the definition of (3.5), where the value of $\gamma(t)$ is zero after the controller transition. The value of $\gamma(t)$ when the transitional CACC controller is employed is indicated in the graph.

The figure shows that position error settles at approximately -1.3 meters. The bias can be explained by the fact that this error is not fed into the controller as explained in Appendix C. It is noteworthy that after the controller transition is completed there are some oscillations of approximately ± 0.7 meters in the position error. This is partly caused by some initial errors when γ and its derivatives are initiated. Moreover, some problems with the actuation of the vehicle contributed to this. The exact problems are discussed when analyzing the acceleration of the vehicle. The oscillations in the position error have largely dampened out when q_{lc} is reached. Overall, these errors are within typical standstill distances, and the maneuver is therefore deemed to be safe.

The velocities during the experiment are shown in Figure 6.5b. The merging vehicle has a higher average velocity throughout the maneuver because it needs to close the gap to the preceding vehicle. The velocity during the MPC phase of the experiment appears to be polynomial which is what is to be expected because the reference trajectory is polynomial. Then, during the transition phase, the velocity increases to close a residual gap. One point of interest is that when the transition is completed, the velocity of the merging vehicle appears to be too high. This can be explained by examining the acceleration profile of the vehicle.

The acceleration of the vehicle is shown in Figure 6.5c and compared with the desired acceleration. It is visible that during the transition phase, the desired deceleration does not match the actual deceleration. In essence, the vehicle cannot decelerate as fast as desired. This is because all braking is done using the regenerative brakes which are seemingly insufficient. The desired deceleration goes up to -2 $\mathrm{m/s^2}$. Even though this is not an extreme deceleration, the regenerative braking did not slow down the car enough. This is indicated by the throttle position plot in Figure 6.5d. During the mismatch in desired and actual acceleration the minimum throttle position is 0%. This is the maximum deceleration possible by regenerative braking. Because of the insufficient braking, the velocity at the end of the transition phase is too high. Consequently, there exists a residual error at the end of the transition phase. The negative error indicates the vehicle is too close to its target, which is to be expected. The acceleration is close to zero from point q_{lc} and starts to increase when the target vehicle increases its velocity. It can thus be stated that, within the available accuracy and given limits of the model, actuation, and sensing, it is possible to

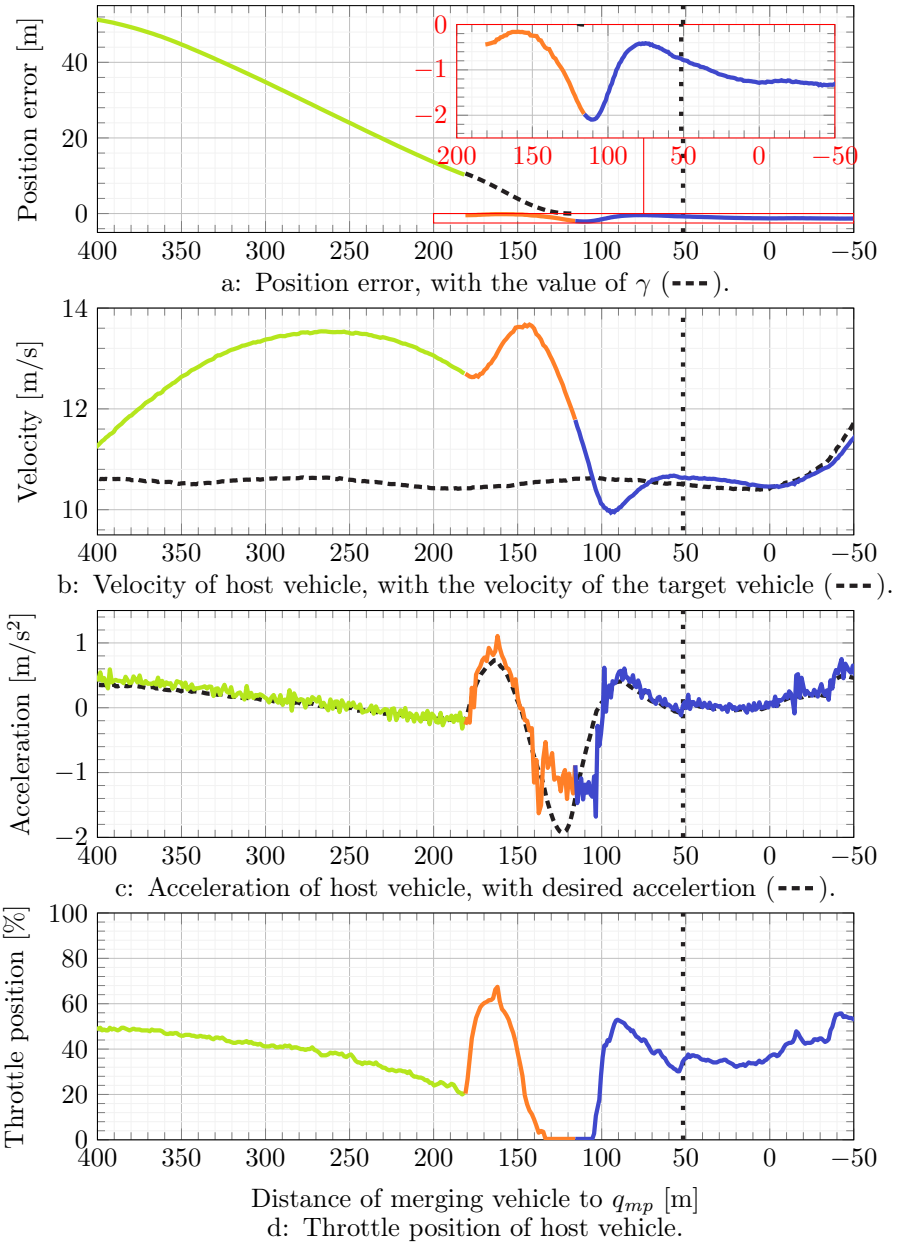


Figure 6.5: The position error during an experiment with the constant velocity scenario. Before (—), during (—), and after (—) the controller transition. The position q_{lc} at which the merging vehicle would initiate its lane change is denoted with the vertical line (· · ·).

form a steady-state platoon when q_{lc} before is reached.

6.3.2 Acceleration Scenario

In this scenario, the preceding vehicle accelerates during the maneuver. The control strategy of the merging assumes a constant velocity of the preceding vehicle. This expected behavior is not matched during this scenario. Therefore, the merging vehicle is required to adjust its desired final states and intended time of the merge. In other words, the merging vehicle is required to adjust its state sooner than initially expected due to the higher speed throughout the maneuver. Furthermore, the acceleration creates a difference between the initial and final velocity of the merging vehicle.

Figure 6.6a that the position error settles at approximately -2 meters. After the controller transition, the error stays within ± 0.8 meters from this value. This behavior is relatively similar to that of the constant velocity experiment. It is concluded that the algorithm can handle the additional challenge posed by excitations of the target vehicle. Furthermore, the error shows little variation after q_{lc} is reached. Indicating that a safe merging maneuver is feasible.

Figure 6.6b shows that the average velocity of the merging vehicle is higher than that of the target vehicle. The higher average velocity is required to close the gap between the vehicles. The peak velocity is higher than that in the constant velocity scenario. This is primarily due to the lower initial velocity in this scenario. A higher peak velocity is required to ensure the average velocity is adequate to close the gap. The velocity of the predecessor is closely matched from q_{lc}. This indicates that a steady-state platoon is established, and the algorithm thus performs well.

The actual and desired accelerations are shown in Figure 6.6c. It is shown that the vehicle has difficulties tracking the desired deceleration during the controller transition. The minimum throttle position of 0%, as shown in Figure 6.6d, is insufficient to decelerate as much as desired. This results in problems similar to those that arose for the constant velocity scenario. Namely, position errors being created and a mismatch in velocity after the transition. Despite the additional challenge caused by the compliance of the setup, the controller manages to form a platoon in a safe and timely fashion.

6.3.3 Deceleration Scenario

To further investigate the performance of the controller during unexpected behavior of the preceding vehicle, experiments involving a braking action are investigated. In these experiments, it is important that the velocity of the merging vehicle is adjusted such that it does not merge with a too high velocity.

The error during the experiment can be found in Figure 6.7a. It is shown that the merging vehicle manages to close the initial gap to the target vehicle.

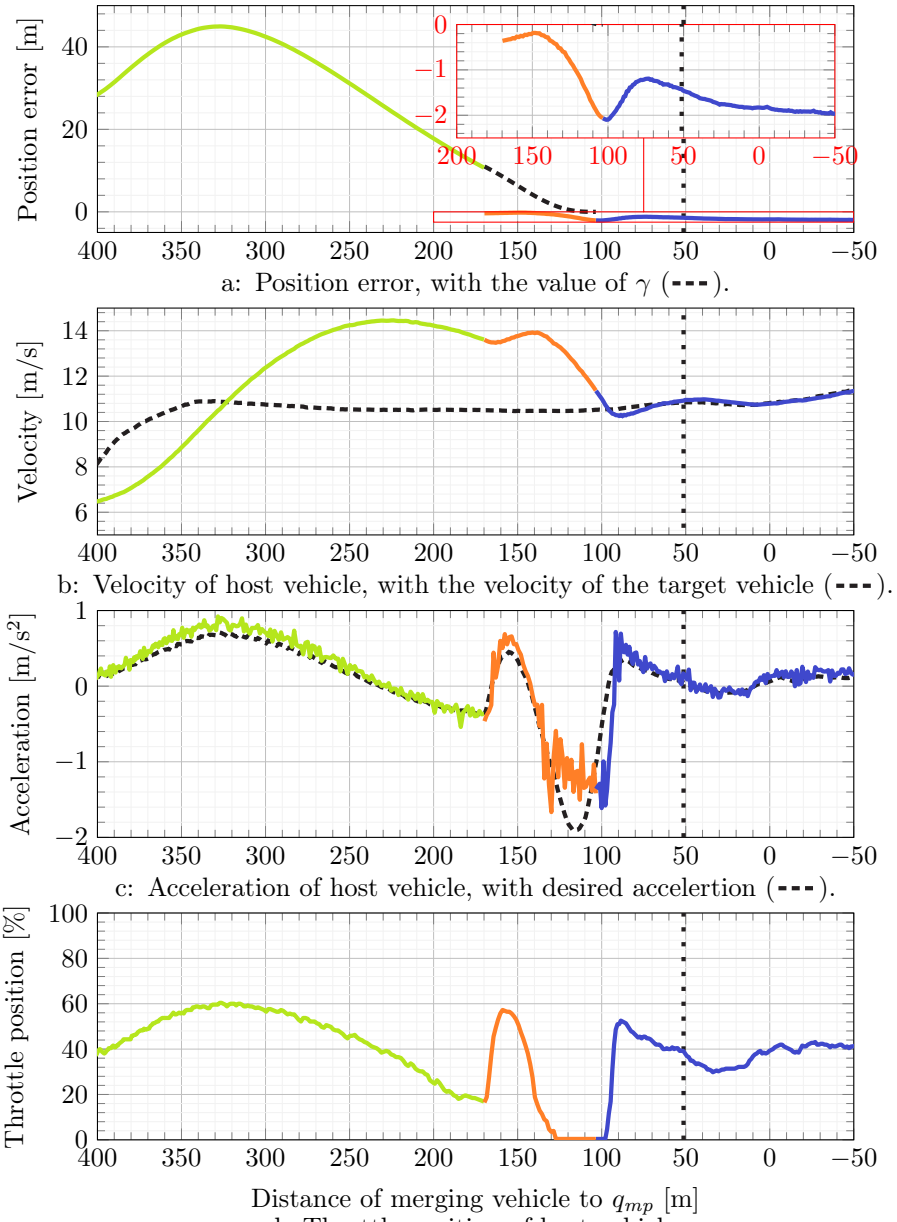


Figure 6.6: The position error during an experiment with the acceleration scenario. Before (——), during (——), and after (——) the controller transition. The position q_{lc} at which the merging vehicle would initiate its lane change is denoted with the vertical line (· · ·).

The error during CACC driving is approximately -1.1 meters. During and after the controller transition the vehicle remains within ±0.5 meters of this value. A notable difference between this measurement and those of the previous scenarios is that no transient behavior is visible during and after the controller transition. The variation in the error appears to have a random nature and may be caused by factors such as measurement noise. Therefore, it can be concluded that the controller behaved well.

The error variation during the steady-state CACC driving appears to be more frequent than that of the previous scenarios. It should be noted that because of the lower velocity, the CACC driving is performed for a longer time. Oscillations caused by disturbances that are equal in the time domain thus appear more frequent in the position-domain. Furthermore, there is some difference between the ease of control of the vehicles at different velocities. Since the properties of the conventional CACC algorithm are well known, these variations are indicative of the performance of the setup rather than of the algorithm.

The velocities in Figure 6.7b clearly show the braking action of the preceding vehicle. The controller transition starts around 240 meters before q_{mp}. Consequently, the initial and desired final states are far apart. Therefore, a big adjustment in the velocity of the merging vehicle occurs during the transition phase. After the transition, the velocity of the new vehicle is only 0.6 m/s higher than that of the preceding vehicle. The transition is thus performed successfully.

Figure 6.7c shows the actual and desired acceleration. It is shown that the vehicle manages to track the desired acceleration well. The lowest measured deceleration of the merging vehicle is -2 m/s^2. However, the throttle position stays above 0% and the vehicle's behavior is not limited by the setup. This precise tracking is part of the reason why the velocity at the end of the transition is matched so accurately. The improved tracking may be due to several factors, the most probable is the lower motor speed during the braking. Overall, this experiment shows the behavior of the proposed transitional controller with a compliant setup. There is hardly any transient behavior visible in the error, velocity, and acceleration plots. Therefore, it can be concluded that the algorithm behaves exactly as expected.

6.3.4 Repeatability

To obtain a better understanding of the performance, the repeatability of the experiments is examined. Three experiments using the acceleration scenario are compared. All these measurements are different from the one presented in Section 6.3.2. Some variation in initial states and the behavior of the preceding vehicle exist. It is important that the maneuver is executed correctly despite these variations. The exact scenarios at an on-ramp will vary a lot and desired behavior must always be met. This section analyzes the differences in behavior between the different runs.

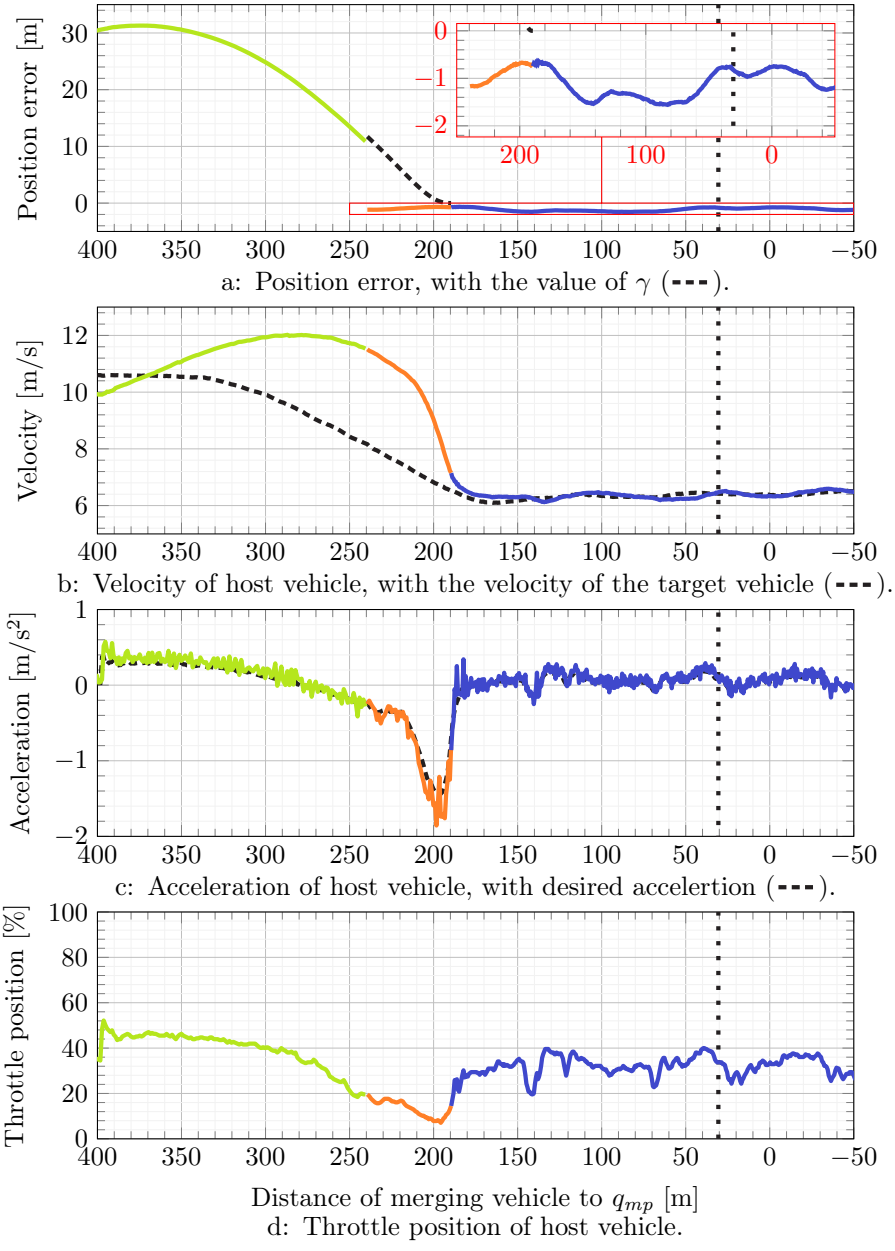


Figure 6.7: The position error during an experiment with the deceleration scenario. Before (—), during (—), and after (—) the controller transition. The position q_{lc} at which the merging vehicle would initiate its lane change is denoted with the vertical line (· · ·).

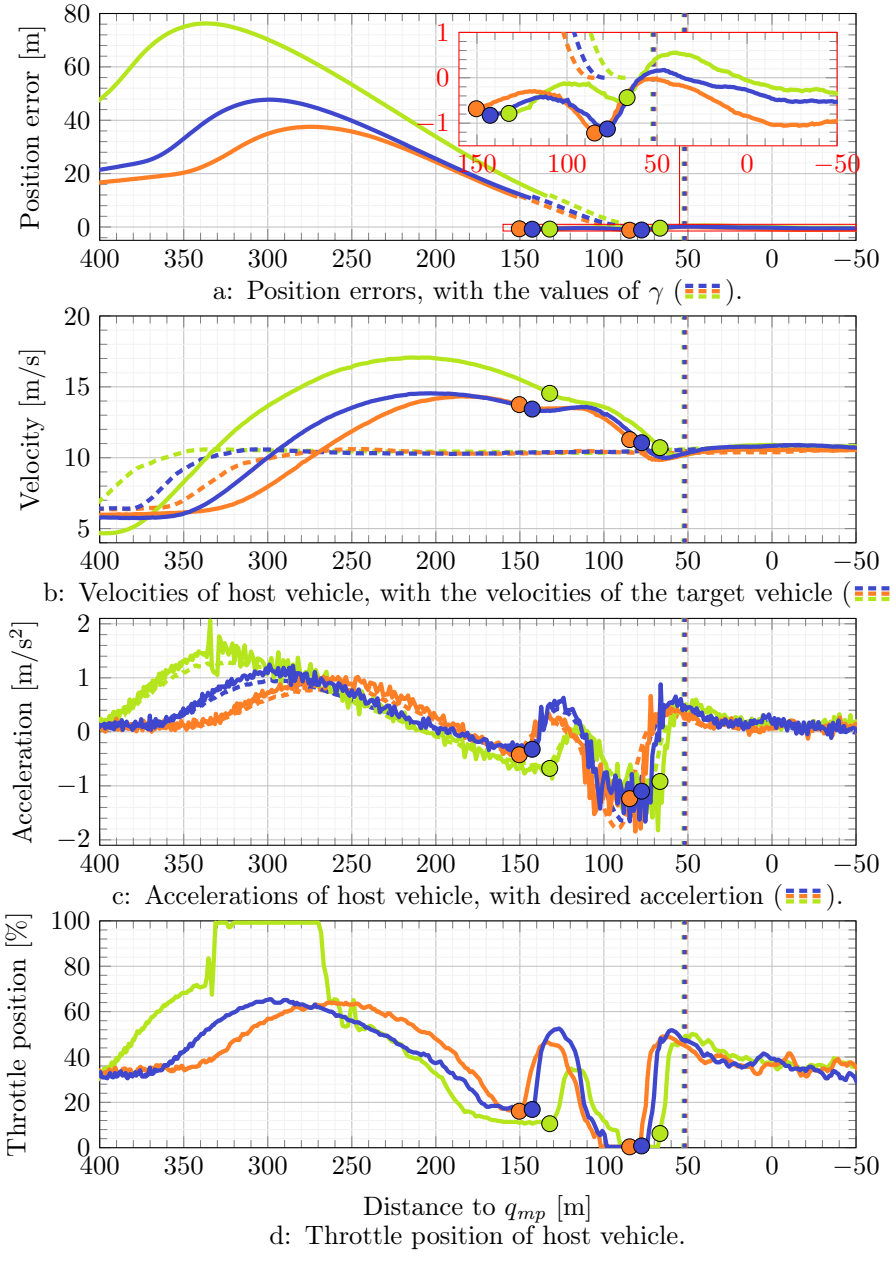


Figure 6.8: The position errors and velocities for three runs with the acceleration scenario. The individual runs are denoted using the colors ( ), ( ) and ( ). The markers indicate the start and end of the controller transition. The vertical lines ( ) indicate position q_{lc} at which the merging vehicle would initiate its lane change.

Results from the three experiments are shown in Figure 6.8. Figure 6.8a shows that the position error is small during and after the controller transition. This is the most important result because it means that the merge can be executed safely. When q_{lc} is reached, the transitions are completed and the errors are small. Furthermore, it is shown that the behavior of the error in these three runs is relatively similar in shape. However, the amplitude during the MPC phase varies vastly. The maximum error during the first run (—) is more than 76 meters, during the second run (—) the peak is less than 38 meters. This difference is caused by the variance in initial conditions between the two runs.

Figure 6.8b shows the velocity profiles during the different experiments. Again, the consistency of the behavior is evidenced by the similar shapes of the trajectories. Overall, the velocity profiles are as expected. There are overshoots in the velocities with respect to the final velocity, which are required to close the gap to the preceding vehicle. It is shown that the first run achieves the highest velocity, this is a logical consequence of the large initial position error. The velocity is matched in all instances during the time reserved for the lane change maneuver.

The accelerations are shown in Figure 6.8c. These also follow the expected behavior which is like that of the previously presented experiments. The first run has the lowest peak acceleration during its controller transition. This is likely because its velocity is highest when the transition is initiated. The higher initial velocity requires a lower average acceleration to match the desired final velocity. In the first run the maximum acceleration peaks briefly over 2 m/s^2. Since this is just a single peak it is likely the result of measurement noise rather than the actual acceleration. In general, the accelerations remain within ± 2 m/s^2 which is reasonable behavior.

Lastly, the throttle positions in Figure 6.8d show that especially the first run is experiencing the limits of the setup. During the MPC phase, a maximum throttle position of 100% is reached. During the transition phase, a minimum throttle position of 0% is achieved. The behavior is thus limited by the setup. Even with these limitations, the maneuver is executed satisfactorily. This once more underlines the robustness of the proposed method.

The results in this section have shown the repeatability of the experiments. This is indicative of the robustness of the proposed merging strategy. The exact results vary, but the general behavior is not influenced by factors such as measurement and actuator noise, or variation in initial conditions.

6.4 Conclusions

This chapter has shown the results of experiments conducted to test the control strategy presented in Chapters 3 and 4. The goal of the experiments is to form a two-vehicle platoon while driving before reaching a prespecified position. During the experiments, multiple challenges are posed to the control algorithm. The

main challenges are those introduced by the setup. Namely, the sensor noise and compliance of the actuators. Despite these challenges, the variable gap CACC algorithm of Chapter 3 during the transition phase has similar accuracy as a conventional CACC algorithm. Furthermore, the merging strategy of Chapter 4 also handles these challenges well, despite their influence on the estimation of variables such as the expected time required to reach a certain point.

An additional disturbance is introduced by accelerating or decelerating the preceding vehicle such that predictions regarding future states do not match. This too could be handled by the proposed algorithm of Chapter 4. It is shown that variance introduced in the experiments due to the human actions required does not alter the general behavior of the vehicle. These results, therefore, demonstrate the proposed longitudinal control strategy for merging.

The control strategy for the merge is intended for highway use. However, these experiments are performed at lower velocities. Based on the throttle position graphs, the velocities and accelerations are on the limit of the capabilities of the vehicles used. It is therefore reasonable to assume that highway maneuvers performed by more powerful vehicles are achievable using this strategy. One extension that can be made in future research is the addition of a third vehicle such that the entire merging maneuver can be demonstrated. Since no additional low-level controllers are used by the third vehicle, it is unlikely that additional problems arise during such experiments with respect to longitudinal motions. Algorithms for lateral motion and perception must be implemented before such experiments can be performed.

Chapter 7

Conclusions and Recommendations

Vehicle automation has the potential to increase traffic safety and efficiency. Due to this potential, this research topic is of great interest. A subclass of automated vehicles is that of connected automated vehicles. These vehicles use communication to receive information beyond that observable by their own sensors. Furthermore, the available communication can be used for cooperative driving, where multiple vehicles strive to increase the combined performance. One example of this technique is platooning. Platooning vehicles drive in a string using small inter-vehicle distances to increase traffic efficiency and throughput. This thesis aims to create a control approach for the merging of a new vehicle in such a string of vehicles. This chapter presents the main conclusions obtained in this work. Additionally, recommendations regarding future work on this topic are proposed.

7.1 Conclusions

Platooning is a technique where vehicles drive closely behind each other to form a string. This technique can potentially improve traffic throughput, efficiency, and safety. One problem for practical implementation of the technique is the desire to merge new vehicles into the string while driving. The main aim of this research is to create a control approach for the merging problem. To achieve this aim, the following objectives are defined:

- Design of a control strategy for the cooperative merging of a new vehicle between two platoon vehicles in highway on-ramp environments. This design includes the required inter-vehicle information flow and longitudinal controllers for the involved vehicles. To ensure a steady-state platoon is obtained at the end of the maneuver, the transition to a Cooperative Adaptive Cruise Control (CACC) controller required by some vehicles is given special attention.
- Design of a sequence manager that decides on the desired position of a new vehicle in a platoon. This manager can handle large differences between the initial states of the platoon and new vehicles.
- Experimental testing of the controller transition in spatially constrained environments with full-scale vehicles.

Based on these objectives, the main contribution of this thesis is a control approach for the merging of a single vehicle into a platoon. This approach includes higher-level components such as merging sequencing and lower-level components such as the longitudinal controller. The lower-level component has been validated using experiments with passenger vehicles.

The design of a lower-level longitudinal CACC controller which allows for variable inter-vehicle distances is the first step to achieving the first objective. Based on the first objective, the most important requirement for this controller is that it can open a gap under spatial constraints. The proposed design is presented in Chapter 3. The controller is required to perform the gap opening and closing actions necessary in the merging maneuver. The control strategy is based on a conventional CACC algorithm and has an additional gap term in the definition of its desired distance. This gap term can be varied over time to perform the opening and closing actions. When the additional gap term is set to zero, the control law is the same as that of the conventional CACC controller. Therefore, a smooth transition from and to a conventional CACC controller can be ensured. Feedforward terms in the control law ensure that the desired gap is accurately tracked. This is demonstrated in experiments using mobile robots. The desired inter-vehicle gap is only accurately tracked during experiments where the feedforwards terms are included. When designing a control strategy for the merging maneuver, the longitudinal controller must be able to

accurately track the desired trajectory to coop with the spatial constraints. The proposed strategy possesses this ability and is deemed suitable for the merging maneuver.

The proposed longitudinal controller of Chapter 3 is implemented in a control strategy for the merging maneuver in Chapter 4. The strategy considers the control of the gap-opening platoon vehicle and the new vehicle after communication is established. Initially, the platoon vehicle performs a gap-opening maneuver and the new vehicle uses an individual controller aiming for the opening gap. At some point, both vehicles need to switch their controller. The platoon vehicle switches from the gap-opening controller to a CACC controller targeting the new vehicle. The new vehicle switches from the individual controller to the CACC controller. To ensure a smooth and timely transition to the conventional CACC controller, these switches are performed with a transition strategy that uses the variable gap CACC controller of Chapter 3. The gap term and feedforward terms are initialized such that the residual gap is zero. Then, the gap and feedforward terms are brought to zero such that the conventional CACC controller is obtained. One potential safety concern arises when the following vehicle has switched its control target. The preceding vehicle in its own lane is then not directly considered anymore. Since a collision may occur due to the longitudinal movement of the vehicles, a collision-avoidance controller must be used to ensure safety. This controller prevents the gap-opening platoon vehicle from having a rear-end collision with the preceding vehicle after it switched its control target to the new vehicle. The control strategy is demonstrated using multiple simulations with sensor noise and communication delays. The simulations have varying platoon perturbations, disturbances, and noise profiles. The maneuver is executed safely for all simulations, emphasizing its performance. The first objective is now achieved by the combined work of Chapters 3 and 4.

The second objective, concerning merging sequencing, is tackled in Chapter 5. The merging sequence determines where in the platoon the new vehicle will merge. This decision determines the required trajectories of the vehicles involved in the merging maneuver. The sequence prescribes which platoon vehicle is required to open a gap, and thus which subsequent platoon vehicles experience the perturbation. Evidently, the trajectory of the new vehicle is also dependent on its position in the platoon. There are multiple possible methods for merging sequence management. The available sequencing methods can be categorized as distance-based, time-based, or optimization-based. A newly proposed optimization-based method is presented that aims to minimize the perturbations for subsequent platoon vehicles. Using simulations, the method is analyzed and compared to benchmark methods of all three categories. The proposed method does not significantly outperform the benchmark time-based or optimization-based method. This is likely due to measurement noise and communication delay present in the simulations. The optimization-based methods use predictions of future trajectories of the vehicles. These trajectory predictions

can be distorted by the disturbances, resulting in errors in the cost predictions of different sequences. Furthermore, compared to the optimization-based methods, the time-based method can more easily be implemented, is less computationally heavy, and appears to be less affected by the noise. Though based on the presented results some may argue that the optimization-based methods produce slightly favorable sequences, the additional advantages of the time-based method indicate that the sequences produced by the optimization-based methods must be significantly beneficial before this type of method is used. The usage of optimization-based strategies is therefore not recommended at this point, and future research using a time-based method as a benchmark is required before a final decision is made. It can be noted that the merge is executed successfully and safely in all simulations. This confirms that the previously designed merging maneuver functions well and can handle a wide range of scenarios. Moreover, the time-based and optimization based methods investigated in this chapter are all able to handle large differences between the initial states of the platoon and new vehicles. While more research on which strategy is the best amongst these three potential strategies is required, it can therefore be concluded that the second objective is achieved.

Experiments demonstrating the performance of the proposed merging strategy of Chapters 3 and 4 are presented in Chapter 6. The experiments consider the formation of a two-vehicle platoon using passenger vehicles. The first vehicle is human-driven, and the second vehicle uses the proposed automated controller to merge behind it. Throughout the experiment, the new vehicle thus uses the individual and CACC controllers, and the algorithm to transition between them. The experiments show that the designed approach can handle the disturbances experienced by the setup. The experiments are repeated using different inputs of the human-driven platoon leader. The platoon formation is successfully completed in every experiment and the repeatability is shown. Due to the limited number of vehicles, the behavior of the gap-opening vehicle is not demonstrated. Since it uses a combination of the same controllers, it is credible that its control strategy can also handle the disturbances of a full-size platform. With these experiments, the third and final objective is achieved. To conclude, the proposed control strategy can be applied to passenger vehicles. The experimental results suggest that the merging maneuvers are executed in a safe and timely manner.

7.2 Recommendations

This thesis presents a strategy for the merging of a new vehicle into an existing platoon. Recommendations regarding future work are presented in this section. First, recommendations considering the on-ramp merging application are given. These recommendations either aim to improve the performance of the current solution or to ensure acceptable vehicle behavior which is required before this technology can be brought to market. An example of the latter category is the

behavior of the vehicles in rare but dangerous situations. Then some recommendations regarding the extension of the presented work to other applications.

In Chapter 3 the proposed variable gap CACC controller is based on the controller of Ploeg et al. (2011). The main idea behind the proposed controller is making the desired inter-vehicle distance variable and ensuring this is tracked by studying the error dynamics. Therefore, the proposed method may be used on other platooning algorithms. This has potential benefits on the vehicle behavior or the requirements on the gap term. For example, using the current approach the gap term is required to be C^2 continuous due to the necessary derivatives in the feedforward terms. However, using the algorithm of Lefeber et al. (2020), the position error states are differentiated one less time to determine the control law. Possibly, the requirement on the gap term can then be lowered to C^1 continuity. This gives more freedom in designing the trajectory and possibly gives a smoother transition since fewer states are required to initialize the gap term. An analysis of implementing the variable gap in the different platooning strategies may potentially improve vehicle behavior. This recommendation is primarily aimed at performance improvement of the current method. It should also be considered that the platoon may be heterogeneous because in a highway situation not all vehicles are necessarily the same. The platooning algorithm of Ploeg et al. (2011) is intended for homogeneous platoons. Other platooning algorithms, such as Lefeber et al. (2020), do consider heterogeneous platoons specifically. This extension may therefore provide some additional safety benefits when the system is brought to market.

A control strategy for the merging maneuver is proposed in Chapter 4. This strategy shows successful results, but its usage can be extended to a wider range of scenarios. One possibility is the merging of multiple vehicles in the platoon. One can imagine that if for a string of vehicles, all vehicles open a gap at the same time, the last vehicle will come to a standstill. A proper analysis of the merging of multiple vehicles is thus required to ensure safety. This scenario likely requires little adaptation of the strategy since it can handle excitations by the preceding vehicle. However, the knowledge of gaps opening at different places in the platoon may be used to enhance the performance of the strategy. This extension includes the merging of two platoons into one, such as done during the 2016 Grand Cooperative Driving Challenge (GCDC). It will be interesting to compare the proposed approach to the work presented as part of that challenge. Furthermore, the strategy can be altered to allow for gaps in the platoon that fit multiple vehicles. Another interesting scenario is that of mixed-traffic. This is a scenario in which human-driven or unconnected vehicles are present in the environment. Since 100% of market penetration of Connected Automated Vehicles (CAVs) cannot directly be assumed. There is a wide range of research available for mixed-traffic scenarios in general. For example, to ensure cut-in maneuvers in platoons are handled safely. Considering nearby unconnected vehicles in the control strategy can possibly enhance the safety of the maneuver. Fur-

thermore, one potential risk is the loss of communication during the maneuver. Safety must be ensured during this situation. Since the loss of communication of some vehicles essentially creates a mixed-traffic situation, this special case can be considered while investigating the mixed-traffic situation. Before the technology is brought to market, it is important to guarantee safety even in these rare situations. The performance may be pushed slightly by allowing the vehicle alignment to end after the lane change is initiated but before it enters the main lane. In essence, the current strategy has a sequential philosophy where the vehicles must be aligned before the lane change is initiated to ensure the lane change is completed before the vehicle enters the main lane. When investigating this possibility, it is important to consider the safety of the maneuver.

Multiple sequencing methods are analyzed in Chapter 5. The current analysis shows the potential of time-based and optimization-based methods. However, despite the large number of simulations, the results are not conclusive. Further investigation into this topic is required to select the preferred merging sequencing method. The analysis can be extended in multiple ways. First, longer traffic flow simulations can be used to investigate the throughput at highway entrances. This type of simulation is often used in literature and is important because the entrances can cause bottlenecks in the traffic flow. However, to the best of the author's knowledge, a comparison between time-based and optimization-based methods using such simulations has not yet been published. The results from this thesis suggest that such a comparison would be interesting. Another extension to the current analysis is using an experimental setup. It is attempted to add realism to the simulations by introducing aspects such as sensor noise and communication delay. However, a discrepancy between simulation and reality is inevitable. Especially because of the effect of noise on the optimization-based methods, the additional disturbances from an experimental setup are interesting. These experiments require a large number of vehicles. For practical reasons, experiments using small mobile robots can be used, although full-size vehicles are advantageous from a quality point of view. By artificially adding sensor noise and communication delays on top of that already present in the experimental setup a wide range of problems can be investigated. These experiments are required to ensure an acceptable merging sequence is produced in any situation.

Experiments using the proposed merging strategy are presented in Chapter 6, these experiments must be expanded to validate the performance of the proposed method. An additional vehicle can be used to test the merging between two vehicles. This will require an expansion of the current demonstrator platform which consists of two vehicles. Few revelations regarding the longitudinal controllers of the gap-opening platoon vehicle are expected since the components are tested individually in the current experiments. The three-vehicle experiment requires a lateral movement of the new vehicle. Such movement was not included in the current experiments and can provide additional insights. The lateral movement can be controlled manually or with a controller from literature. All aspects of

the merging maneuver are then included in the experiments. Furthermore, the braking can be activated on the demonstrator platform. This can improve the trajectory following of the vehicles and allows for experiments with more severe braking scenarios.

When designing new experiments, the sequencing method of Chapter 5 can be incorporated if enough vehicles are available. Dependent on the size of the platoon, the initial positions of the vehicles must be relatively well controlled to ensure that a position in the platoon is selected. An important environmental aspect is the inclusion of the on-ramp such that the maneuver is started before the platoon is in the line-of-sight of the new vehicle. This can be done artificially by not considering on-board perception sensors up to a certain point. Similarly, because the testing environment may be so small that all vehicles can communicate at any point, the communication range may be limited artificially by excluding information from vehicles based on their two-dimensional distance. By controlling the platoon leader, disturbances found throughout this thesis may be included in these experiments. Such disturbances include accelerating and braking actions, and sinusoidal inputs on the acceleration. Moreover, the experiments must also include safety-critical scenarios, such as communication loss and mixed-traffic scenarios, once the theory for those situations is completed. These experiments must be performed before the product is brought to market.

Lastly, future work can focus on applying the presented work for other cooperative driving problems. First, is the problem of nonsignalized intersection crossing. In literature, a popular approach is using virtual platoons. One issue here is ensuring the timely convergence to a steady-state platoon. This can be solved using the transitional controller of Chapter 4. The second problem is that of platoon leaving. When a vehicle exits the platoon, the subsequent vehicle is required to transition its controller and close the gap if it wants to maintain in the platoon. The proposed transitional controller can be a good solution for this as it avoids aggressive accelerations expected from a conventional CACC controller. In literature, the Artificial Potential Field (APF) approach is another solution to avoid aggressive gap closing. This solution requires the APF platooning algorithm to be used continuously. The APF approach has advantages and disadvantages compared to a conventional CACC algorithm. A comparison between these two gap-closing solutions is therefore an interesting extension, which can be performed in simulations and experiments.

APPENDIX A

Supplementary Analyses for the Merging Maneuver

A.1 Analysis of Changing the Initial Position of the New Vehicle

In this appendix, the influence of the initial position of vehicle n is investigated. During the analysis, the vehicle is moved half a platoon position forward and backward. This is because the current work does not present a method for sequence management. The shift in position is representative of the possible initial positions the vehicle may assume without expecting a change in the proposed sequence. However, due to the lack of sequence management, other initial parameters cannot be investigated at this moment. This will be subject to the analysis in Chapter 5 when a merging sequence strategy can be used.

To keep this appendix brief, only the results of the acceleration scenario are presented here. This scenario causes the largest excitations. It is assumed all vehicles in the platoon are L_p long with a standstill distance and headway time of r_p and h_p. The distance between two adjacent positions in the platoon is thus $v_p h_p + L_p + r_p$. The initial position of vehicle n is therefore moved $q_{\delta,n} = 0.5(v_p h_p + L_p + r_p)$ forward or backward.

In Figure A.1 the effect of changing the initial position is shown. It should be noted that the behavior of vehicle p is identical in both simulations. The behavior of vehicle f is identical up to the controller transitions for both disturbances. After the controller transition is initiated, the trajectory of vehicle f starts to deviate. This is partly due to the difference in controller transition timing. The trajectory of vehicle n causes vehicle f to find a suitable γ-trajectory at a different time. For example, when vehicle n is moved backward the initial distance between the two vehicles is smaller. A suitable transitional trajectory can thus be found earlier.

It is shown that the trajectory of vehicle n is greatly influenced by the change

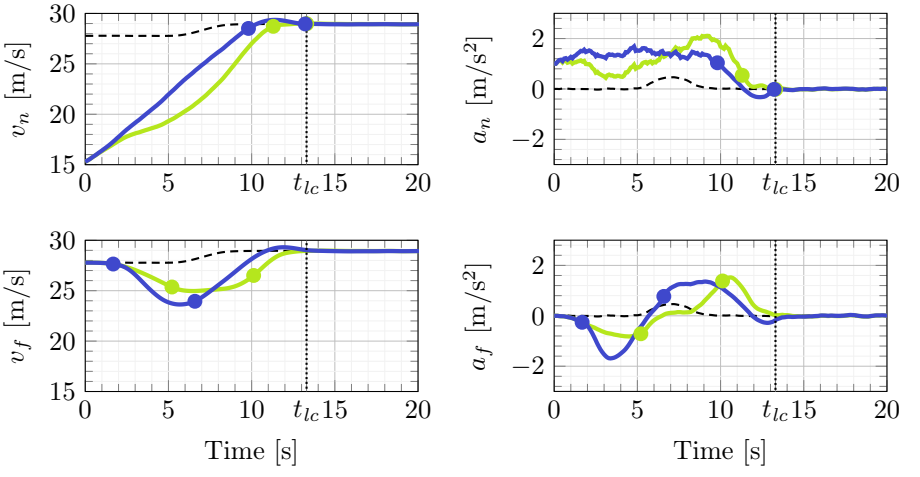


Figure A.1: The velocity and acceleration of vehicles n and f during the acceleration scenario. Vehicle n is moved $q_{\delta,n}$ forward (——) or backwards (——). The circles indicate the controller transitions. The velocity and acceleration of the preceding vehicle is indicated (----).

of initial position. When the vehicle is moved forwards its acceleration is lower at the first part of the simulation to reach the correct position. Then, the acceleration increases to reach the correct velocity, resulting in a higher maximum acceleration. The amplitude of the maximum acceleration is just over 2 m/s^2. This should be just within the range of a high-end passenger vehicle. This underlines the importance of selecting the correct merging sequence and motivates the work performed in Chapter 5.

A.2 Analysis of Disturbances in the Estimation of the Preceding Vehicle's Acceleration

One challenge for the proposed solution is that vehicles require the acceleration of their new CACC target during the controller transition. Vehicles cannot measure the acceleration of other vehicles and thus must estimate it. This appendix investigates the influence of errors in this estimation on the behavior of vehicles n and f. An additional error of ± 1 m/s^2 is added to the estimated acceleration of other vehicles in the control of vehicle n. This influences the controller transition as it affects the initial determination of γ and its derivatives. The effect of this disturbance on the error dynamics can be seen in Figure A.2. The effect on the velocity and acceleration is less significant.

An error of ± 1 m/s^2 is relatively large compared to the working range of a

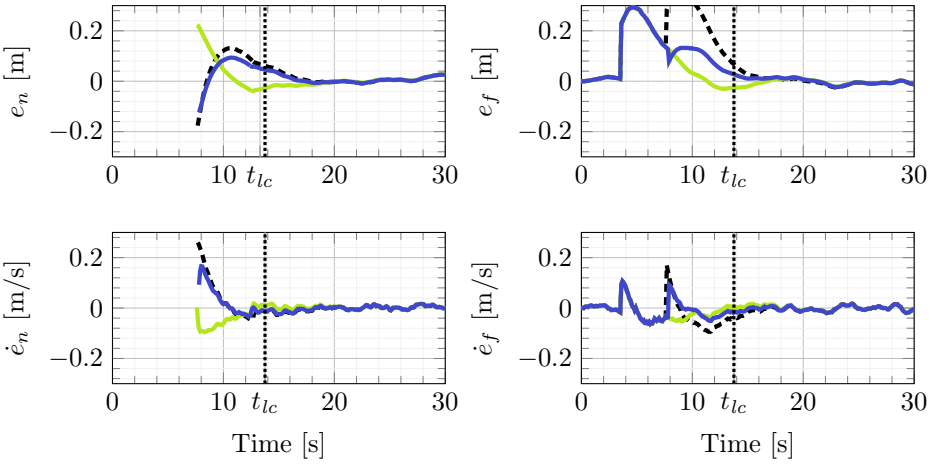


Figure A.2: The error and its derivative for vehicles n and f during the constant velocity scenario. With a positive (——) and negative (——) disturbance on the perceived acceleration. Errors without the disturbance (- - - -) are shown for reference.

passenger vehicle. Figure A.2 shows that the effect of this disturbance on the error is relatively small. The controller can handle this disturbance well and over time the error signals of the two simulations converge. The position error stays within centimeter range and thus no dangerous situation occurs.

The controller is challenged more by combining the initial position movement of vehicle n with the disturbance in error estimation. One of the most extreme cases is the acceleration scenario when vehicle n is moved forward. The results are shown in Figure A.3. The resulting errors remain in centimeter range and are deemed safe as it is much smaller than a typical standstill distance. An error in the estimation of a_p thus has a sufficiently small influence. However, the system may be improved by using an appropriate estimator of the preceding vehicle's states.

A.3 Analysis of the Behavior for Continuous Velocity Changes by the Platoon Leader

In merging situations, the velocity of the platoon leader may change frequently. To investigate the effect of such behavior, this appendix includes the results of two simulations where the platoon leader has a sinusoidal disturbance. The desired acceleration of the leader has an amplitude of 1 m/s^2 and a frequency of 1 or 2 rad/s. Due to the platoon dynamics, the amplitude of the 2 rad/s disturbances dampens more throughout the platoon than that of the 1 rad/s

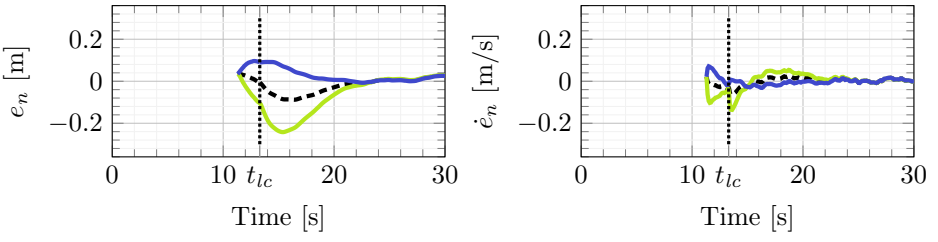


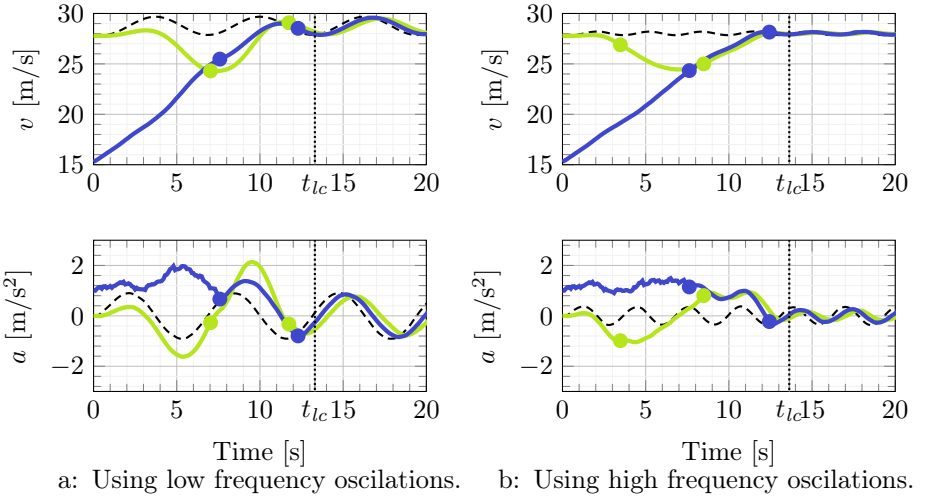
Figure A.3: The error and its derivative for vehicle n during the deceleration scenario when the initial position is moved $q_{\delta,n}$ forwards. With a positive (——) and negative (——) disturbance on the perceived acceleration. Errors without the disturbance (- - - -) are shown for reference.

disturbance. Both simulations thus pose different challenges.

The velocities and accelerations of the vehicles during these simulations are shown in Figure A.4. The velocity profiles show that the subsequent vehicles mimic the behavior of the platoon leader relatively well. High accelerations are encountered for the 1 rad/s scenario. This is due to the high acceleration of the preceding vehicle, on top of which the gap opening and vehicle alignment are executed. The peak acceleration of approximately 2 m/s^2 is reasonable considering that an acceleration of 1 m/s^2 is surpassed in the constant velocity scenario. Furthermore, for the 2 rad/s disturbance the acceleration profiles are similar in amplitude to that of the constant velocity scenario.

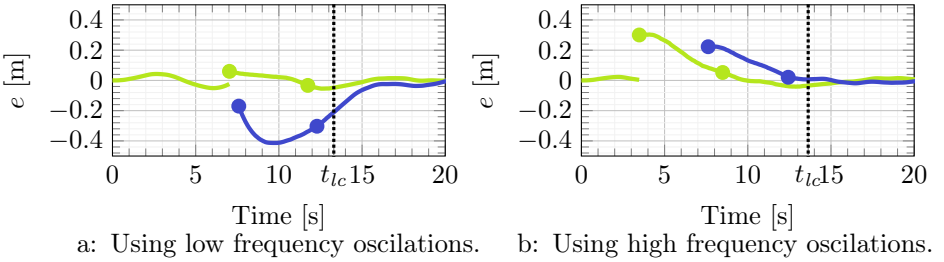
The position errors during the maneuver are shown in Figure A.5. The error becomes largest in the 1 rad/s scenario with approximately 0.4 meters for the new vehicle. The error dampens to 0.2 meters at t_{lc}. For the 2 rad/s scenario the control algorithm appears better capable to handle the disturbance and the errors remain smaller. Overall, these errors remain within reasonable limits.

It can be concluded that even when the preceding vehicle does not follow the trajectory expected by the control algorithm, the controller functions well. It is demonstrated that when a sinusoidal trajectory with a large amplitude is performed by the preceding vehicle, the control algorithm performs the maneuver as expected. Combined with the acceleration and deceleration scenarios, these simulations show that the algorithm can handle disturbances from the platoon leader.



a: Using low frequency oscilations. b: Using high frequency oscilations.

Figure A.4: The velocity and acceleration of vehicles n (——) and f (——) during the sinusoidal oscillation scenarios. The circles indicate the controller transitions. The velocity and acceleration of the preceding vehicle is indicated (----).



a: Using low frequency oscilations. b: Using high frequency oscilations.

Figure A.5: The position error of vehicles n (——) and f (——) during the sinusoidal oscillation scenarios. The circles indicate the controller transitions.

APPENDIX B

Supplementary Material for the Sequence Manager

B.1 Sequencing algorithms

This appendix provides the pseudo-code for the benchmark sequencing algorithm. Their main philosophies are explained in Section 5.4.

Algorithm B.1: Distance-based sequence manager.

Result: PreID, FolID
PreID $= \emptyset$; `// Index of the preceding platoon vehicle`
FolID $= \emptyset$; `// Index of the following platoon vehicle`
leadID $= \min(\mathcal{C}_k)$; `// Determine platoon leader`
foreach $i \in \mathcal{C}_k \backslash leadID$ **do**
  if $q_{new} > q_i$ **then**
    PreID $= i - 1$;
    FolID $= i$;
    break;
  else if $i = \max(\mathcal{C}_k)$ & $q_{new} \geq q_i - L_{new} - r_{new} - h_{new} v_{lead}$ **then**
    PreID $= i$; `// New vehicle merges behind platoon`
  end
end

Algorithm B.2: Time-based sequence manager.

Result: PreID, FolID

PreID = $\emptyset$; // Index of the preceding platoon vehicle

FolID = $\emptyset$; // Index of the following platoon vehicle

leadID = $\min(\mathcal{C}_k)$; // Determine platoon leader

t_f^* = OptimalFinalTime(x_{new},q_{mp},v_{lead},τ_{lc}) ;

$t_{lead,mp} = (q_{mp} - q_{lead}) / v_{lead}$;

foreach $i \in \mathcal{C}_k \backslash leadID$ **do**

  $t_{i,mp} = t_{i\text{-}1,mp} + (r_i + L_i) / v_{lead} + h_i$;

  if $t_f^* + \tau_{lc} < t_{i,mp}$ **then**

    PreID = $i - 1$;

    FolID = i ;

    break;

  else if $i = \max(\mathcal{C}_k)$ & $t_f^* + \tau_{lc} < t_{i,mp} + (r_{new} + L_{new}) / v_{lead} + h_{new}$ **then**

    PreID = i ; // New vehicle merges behind platoon

  end

end

Algorithm B.3: Conventional optimization sequence manager.

Result: PreID, FolID
PreID = $\emptyset$; `// Index of the preceding platoon vehicle`
FolID = $\emptyset$; `// Index of the following platoon vehicle`
leadID = $\min(\mathcal{C}_k)$; `// Determine platoon leader`
$\mathbf{X}_{lead}$ = PredictLeadStates($\vec{x}_{lead}$);
$\vec{c}(k) = \text{rms}(\vec{a}_{lead})\ \forall\ k \in \mathcal{C}_k$;
foreach $i \in \mathcal{C}_k$ **do**
  $\mathbf{X}_{new}$ = MergeBehind($\mathbf{X}_i, \vec{x}_{new}$);
  $\vec{c}(i) = \vec{c}(i) + \text{rms}(\vec{a}_{new})$;
  if $|\vec{a}_{new}| < \tau_a$ & $|\vec{j}_{new}| < \tau_j$ & $i \neq \max(\mathcal{C}_k)$ **then**
    $\mathbf{X}_{i+1}$ = GapOpeningPredict($\mathbf{X}_i, \vec{x}_{i+1}$);
    $\vec{c}(i) = \vec{c}(i) + \text{rms}(\vec{a}_{i+1})$;
    if $|\vec{a}_{i+1}| < \tau_a$ & $|\vec{j}_{i+1}| < \tau_j$ & $i+1 < \max(\mathcal{C}_k)$ **then**
      foreach $j \in \mathcal{C}_k$ & $j > i+1$ **do**
        $\mathbf{X}_j = \text{lsim}(\mathbf{X}_{j\text{-}1}, \vec{x}_j)$;
        $\vec{c}(i) = \vec{c}(i) + \text{rms}(\vec{a}_j)$;
      end
    else
      $\vec{c}(i) = \infty$;
    end
    $\mathbf{X}_{i+1} = \text{lsim}(\mathbf{X}_i, \vec{x}_{i+1})$;
    $\vec{c}(k) = \vec{c}(k) + \text{rms}(\vec{a}_{i+1})\ \forall\ k \in \mathcal{C}_k$ & $k > i$
  else
    $\vec{c}(i) = \infty$;
  end
end
if $\exists \vec{c}(i) \neq \infty$ **then**
  PreID= $\min_i(\vec{c}(i))$;
  if *PreID* $\neq \max(\mathcal{C}_k)$ **then**
    FolID = PreID+1 ;
  end
end

B.2 Resequencing algorithms

This appendix includes the resequencing algorithm of the proposed sequence manager and the three benchmarks. Their design is relatively similar to the corresponding sequence algorithms. For this reason, they are omitted from the main chapter.

Algorithm B.4: Last vehicle optimization resequence cost calculation.

Result: c_f, c_b, c_c

leadID = $\min(\mathcal{C}_k)$; // Determine platoon leader
$\mathbf{X}_{lead} = \text{PredictLeadStates}(\vec{x}_{lead})$;
foreach $i \in \mathcal{C}_k$ & $leadID < i < PreID$ **do**
  $\mathbf{X}_i = \text{lsim}(\mathbf{X}_{i\text{-}1}, \vec{x}_i)$;
end
$\mathbf{X}_{new,f} = \text{MergeBehind}(\mathbf{X}_{pre,f}, \vec{x}_{new})$;
if $|\vec{a}_{new,f}| < \tau_a$ & $|\vec{j}_{new,f}| < \tau_j$ **then**
  $\mathbf{X}_{pre,f} = \text{GapOpeningPredict}(\mathbf{X}_{pre\text{-}1}, \vec{x}_{pre})$;
  $\mathbf{X}_{fol,f} = \text{GapClosingPredict}(\mathbf{X}_{pre,f}, \vec{x}_{fol})$;
  if $|\vec{a}_{pre,f}| < \tau_a$ & $|\vec{j}_{pre,f}| < \tau_j$ & $|\vec{a}_{fol,f}| < \tau_a$ & $|\vec{j}_{fol,f}| < \tau_j$ **then**
    foreach $i \in \mathcal{C}_k$ & $i > FolID$ **do**
      $\mathbf{X}_i = \text{lsim}(\mathbf{X}_{i\text{-}1}, \vec{x}_i)$;
    end
    $c_f = \max(|\vec{a}_{\max(\mathcal{C}_k)}|)$;
  else
    $c_f = \infty$;
  end
else
  $c_f = \infty$;
end
if $FolID \neq \emptyset$ **then**
  $\mathbf{X}_{pre,b} = \text{lsim}(\mathbf{X}_{pre\text{-}1}, \vec{x}_{pre})$;
  $\mathbf{X}_{fol,b} = \text{GapClosingPredict}(\mathbf{X}_{pre,b}, \vec{x}_{fol})$;
  $\mathbf{X}_{new,b} = \text{MergeBehind}(\mathbf{X}_{fol,b}, \vec{x}_{new})$;
  if $|\vec{a}_{new,b}| < \tau_a$ & $|\vec{j}_{new,b}| < \tau_j$ & $|\vec{a}_{fol,b}| < \tau_a$ & $|\vec{j}_{fol,b}| < \tau_j$ & $FolID < \max(\mathcal{C}_k)$ **then**
    foreach $i \in \mathcal{C}_k$ & $i > FolID$ **do**
      if $i = FolID + 1$ **then**
        $\mathbf{X}_i = \text{GapOpeningPredict}(\mathbf{X}_{i\text{-}1}, \vec{x}_i)$
      else
        $\mathbf{X}_i = \text{lsim}(\mathbf{X}_{i\text{-}1}, \vec{x}_i)$;
      end
    end
    $c_b = \max(|\vec{a}_{\max(\mathcal{C}_k)}|)$;
  else if $|\vec{a}_{new,b}| < \tau_a$ & $|\vec{j}_{new,b}| < \tau_j$ & $|\vec{a}_{fol,b}| < \tau_a$ & $|\vec{j}_{fol,b}| < \tau_j$ **then**
    $c_b = \max(|\vec{a}_{new,b}|)$;
  else
    $c_b = \infty$;
  end
else
  $c_b = \infty$;
end
$c_c = \text{CalculateCurrentCost}(\mathbf{X}_{pre,b}, \vec{x}_i \forall i \in \mathcal{C}_k \text{ \& } i > \text{PreID}, \vec{x}_{new})$;

Algorithm B.5: Distance-based resequence manager.

Result: PreID, FolID
if $q_{new} > q_{pre} + \tau_{dist}$ & $PreID \neq \min(\mathcal{C}_k)$ **then**
  PreID = PreID-1 ; `// Move new vehicle forward`
  FolID = PreID ;
else if $q_{new} < q_{fol} - \tau_{dist}$ & $FolID \neq \emptyset$ **then**
  PreID = FolID ; `// Move new vehicle backward`
  if $FolID \neq \max(\mathcal{C}_k)$ **then**
    FolID = FolID+1 ;
  else
    FolID = $\emptyset$;
  end
end

Algorithm B.6: Time-based resequence manager.

Input: PreID, FolID, $t_{pre,mp}$, $t_{fol,mp}$, t_f^*
Result: PreID, FolID
if $t_f^* + \tau_{lc} < t_{pre,mp} - \tau_{time}$ **then**
  $\mathbf{X}_{new} = \text{PredictBehaviorMoveForward}(\vec{x}_{new}, \vec{x}_i \forall i \in \mathcal{C}_k)$;
  if $|\vec{a}_{new}| < \tau_a$ & $|\vec{j}_{new}| < \tau_j$ & $PreID \neq \min(\mathcal{C}_k)$ **then**
    PreID = PreID-1 ; `// Move new vehicle forward`
    FolID = PreID ;
  end
else if $t_f^* + \tau_{lc} > t_{fol,mp} + \tau_{time}$ & $FolID \neq \emptyset$ **then**
  $\mathbf{X}_{new} = \text{PredictBehaviorMoveBackward}(\vec{x}_{new}, \vec{x}_i \forall i \in \mathcal{C}_k)$;
  if $|\vec{a}_{new}| < \tau_a$ & $|\vec{j}_{new}| < \tau_j$ **then**
    PreID = FolID ; `// Move new vehicle backward`
    if $FolID \neq \max(\mathcal{C}_k)$ **then**
      FolID = FolID+1 ;
    else
      FolID = $\emptyset$;
    end
  end
end

Algorithm B.7: Conventional resequence cost calculation.

Result: c_f, c_b, c_c
leadID = $\min(\mathcal{C}_k)$; // Determine platoon leader
$\mathbf{X}_{lead} = \text{PredictLeadStates}(\vec{x}_{lead})$;
$c_k = \text{rms}(\vec{a}_{lead}) \; \forall \; k \in \{f, b\}$;
foreach $i \in \mathcal{C}_k$ & $leadID < i < PreID$ **do**
  $\mathbf{X}_i = \text{lsim}(\mathbf{X}_{i\text{-}1}, \vec{x}_i)$;
  $c_k = c_k + \text{rms}(\vec{a}_i) \; \forall \; k \in \{f, b\}$;
end
$\mathbf{X}_{new,f} = \text{MergeBehind}(\mathbf{X}_{pre,f}, \vec{x}_{new})$;
if $|\vec{a}_{new,f}| < \tau_a$ & $|\vec{j}_{new,f}| < \tau_j$ **then**
  $\mathbf{X}_{pre,f} = \text{GapOpeningPredict}(\mathbf{X}_{pre\text{-}1}, \vec{x}_{pre})$;
  $\mathbf{X}_{fol,f} = \text{GapClosingPredict}(\mathbf{X}_{pre,f}, \vec{x}_{fol})$;
  $c_f = c_f + \text{rms}(\vec{a}_{new,f}) + \text{rms}(\vec{a}_{pre,f}) + \text{rms}(\vec{a}_{fol,f})$;
  if $|\vec{a}_{pre,f}| < \tau_a$ & $|\vec{j}_{pre,f}| < \tau_j$ & $|\vec{a}_{fol,f}| < \tau_a$ & $|\vec{j}_{fol,f}| < \tau_j$ **then**
    foreach $i \in \mathcal{C}_k$ & $i > FolID$ **do**
      $\mathbf{X}_i = \text{lsim}(\mathbf{X}_{i\text{-}1}, \vec{x}_i)$;
      $c_f = c_f + \text{rms}(\vec{a}_i)$;
    end
  else
    $c_f = \infty$;
  end
else
  $c_f = \infty$;
end
if $FolID \neq \emptyset$ **then**
  $\mathbf{X}_{pre,b} = \text{lsim}(\mathbf{X}_{pre\text{-}1}, \vec{x}_{pre})$;
  $\mathbf{X}_{fol,b} = \text{GapClosingPredict}(\mathbf{X}_{pre,b}, \vec{x}_{fol})$;
  $\mathbf{X}_{new,b} = \text{MergeBehind}(\mathbf{X}_{fol,b}, \vec{x}_{new})$;
  if $|\vec{a}_{new,b}| < \tau_a$ & $|\vec{j}_{new,b}| < \tau_j$ & $|\vec{a}_{fol,b}| < \tau_a$ & $|\vec{j}_{fol,b}| < \tau_j$ & $FolID < \max(\mathcal{C}_k)$ **then**
    $c_b = c_b + \text{rms}(\vec{a}_{new,b}) + \text{rms}(\vec{a}_{pre,b}) + \text{rms}(\vec{a}_{fol,b})$;
    foreach $i \in \mathcal{C}_k$ & $i > FolID$ **do**
      if $i = FolID + 1$ **then**
        $\mathbf{X}_i = \text{GapOpeningPredict}(\mathbf{X}_{i\text{-}1}, \vec{x}_i)$
      else
        $\mathbf{X}_i = \text{lsim}(\mathbf{X}_{i\text{-}1}, \vec{x}_i)$;
      end
      $c_b = c_b + \text{rms}(\vec{a}_i)$;
    end
  else
    $c_b = \infty$;
  end
else
  $c_b = \infty$;
end
$c_c = \text{CalculateCurrentCost}(\mathbf{X}_{pre,b}, \vec{x}_i \forall i \in \mathcal{C}_k \; \& \; i > \text{PreID}, \vec{x}_{new})$;

B.3 Accelerating Leader Scenario

To further investigate the behavior of the sequencing method when experiencing perturbations of the platoon leader, the accelerating leader scenario is used. In the simulations, the platoon leader accelerates by up to 1 m/s^2 during the maneuver. The simulation is repeated with ten different trajectories of the platoon leader where the start of the acceleration is moved by 0.5 seconds. Two of the resulting velocity and acceleration trajectories are shown in Figure B.1. Each trajectory is simulated using ten different noise signals. The results are shown in Figure B.2.

The overall performance of the different sequencing methods is similar to that of the previous analyses. Figure B.2a shows that the distribution in sequences is comparable to that of the decelerating leader scenario. The distance-based and time-based methods result in a single sequence, while the optimization-based methods select up to four different sequences. All methods result in a low position error, as shown in Figure B.2b. The effect of the sequencing algorithms on the minimum and maximum velocity is relatively small as shown in Figure B.2c. Despite the increase in maximum velocity of the platoon leader, no unfeasibly high velocities for the new or following vehicle are encountered. The accelerations in Figures B.2d and B.2e show a slight advantage for the time-based method. When the results of the constant velocity and decelerating leader scenarios are considered, these additional simulations underline the sensitivity of the sequencing methods to the environment.

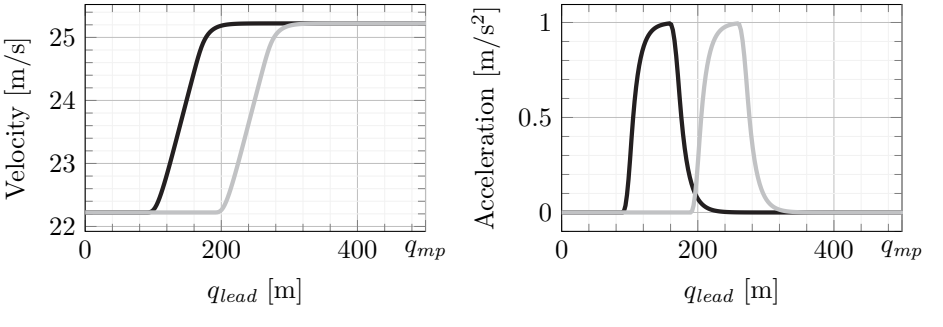
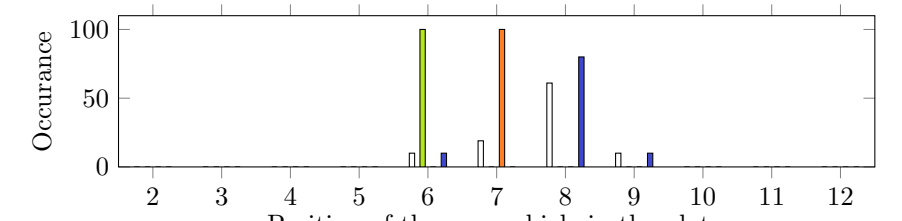
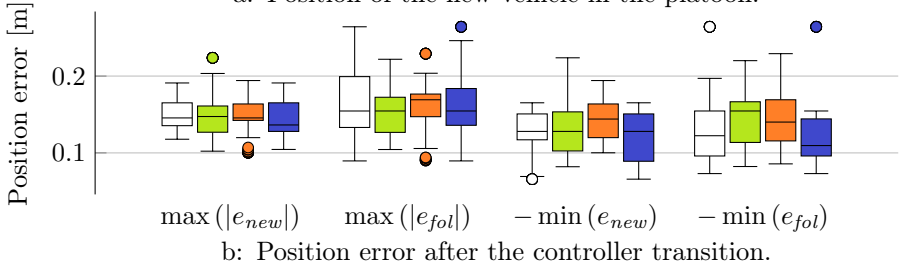


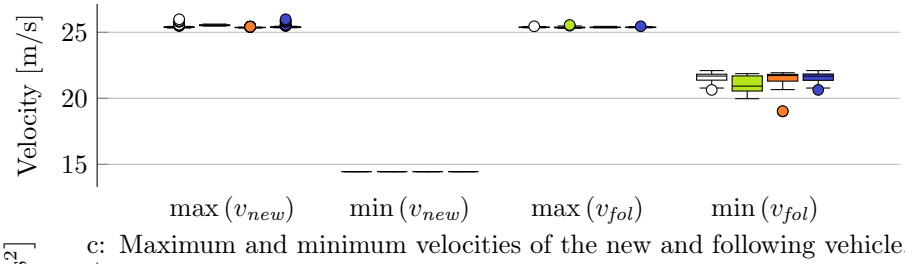
Figure B.1: The velocity and acceleration trajectories of platoon leader during the simulations with the earliest (—) and latest (—) acceleration in the accelerating leader scenario.



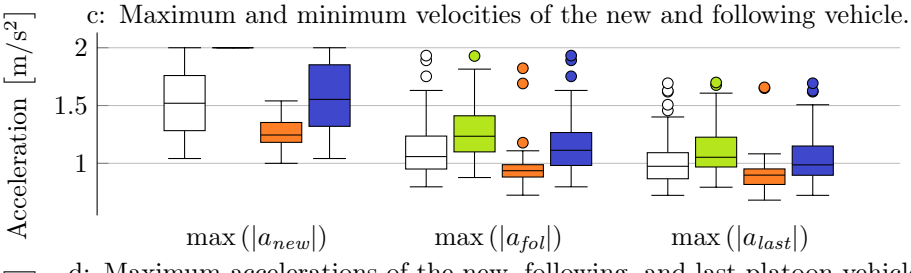
a: Position of the new vehicle in the platoon.



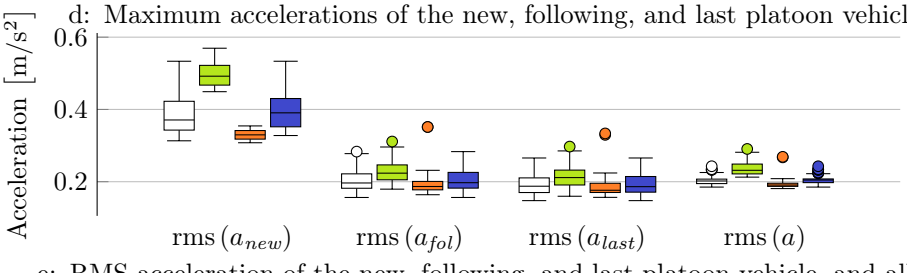
b: Position error after the controller transition.



c: Maximum and minimum velocities of the new and following vehicle.



d: Maximum accelerations of the new, following, and last platoon vehicle.



e: RMS acceleration of the new, following, and last platoon vehicle, and all vehicles.

Figure B.2: The results for 100 simulation parameterizations using the accelerating leader scenario for the last vehicle optimization (□), distance-based (■), time-based (■), and conventional optimization (■) sequencing methods.

B.4 Continuously Changing Velocity Scenario

The previous analyses focused on a maximum of one velocity change by the platoon leader. This section investigates the behavior when the platoon leader is continuously changing its velocity. This is done by adding a sinusoidal input on the desired acceleration of the platoon leader. The amplitude of this signal is 1 m/s^2 and the frequency ranges from 0.5 to 2 rad/s with steps of 0.1 rad/s. This results in a total of sixteen trajectories of the platoon leader, four of which are plotted in Figure B.3. Due to the driveline dynamics, the lower frequency signal results in higher velocities and accelerations. Each trajectory is simulated with ten different noise signals. The results are summarized in Figure B.4.

The distribution in sequences is similar to that of decelerating and accelerating leader scenarios, as shown in Figure B.4a. The position errors are slightly larger due to the continuous unexpected behavior. However, as shown in Figure B.4b, they remain within centimeter range, hence the maneuvers are executed safely. Figure B.4c shows a larger spread in maximum velocities compared to the other scenarios. This is partly caused by the spread in maximum velocity of the platoon leader of the different simulations in this scenario as shown in Figure B.3. Similar to the other scenarios, a higher maximum acceleration of the new vehicle when using the distance-based algorithm is shown. Overall, the velocity results of this scenario are unsurprising. The accelerations in Figures B.4d and B.4e show a disadvantage for the distance-based method which mainly affects the new vehicle. This is explained by the fact that the distance-based method places the new vehicle the furthest forward. Consequently, additional accelerations are required to reach its desired position. The differences between the methods regarding the other vehicles are relatively small. This is likely due to the perturbations of the platoon leader, which heavily influence the trajectories of all vehicles.

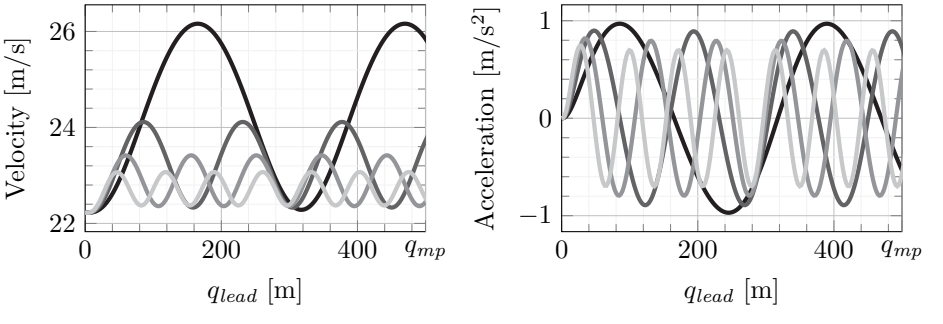
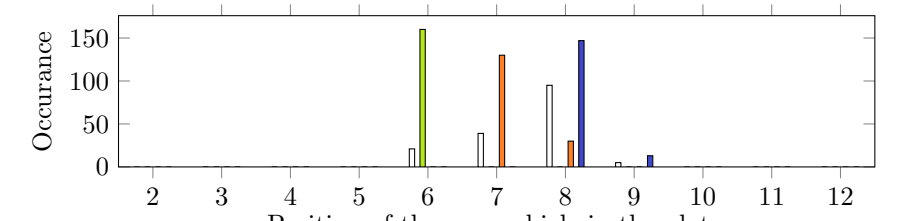
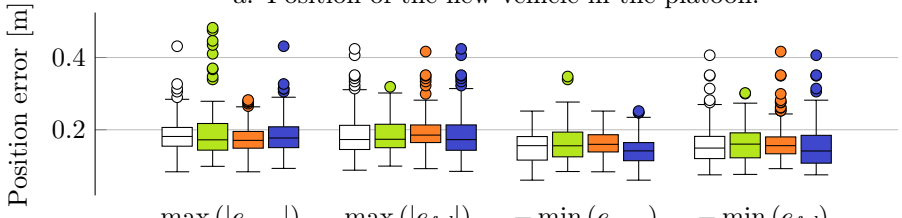


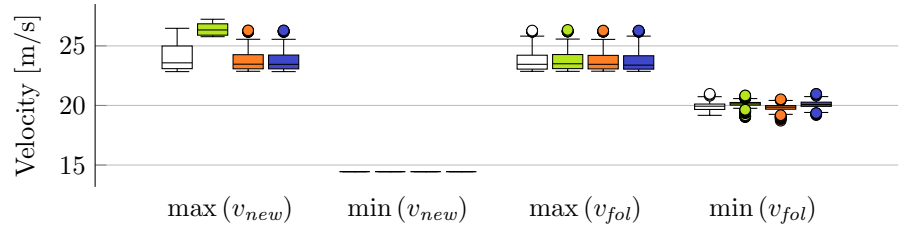
Figure B.3: Four velocity and acceleration trajectories of the continuously changing velocity scenario for 0.5 (—), 1 (—), 1.5 (—), and 2 (—) rad/s.



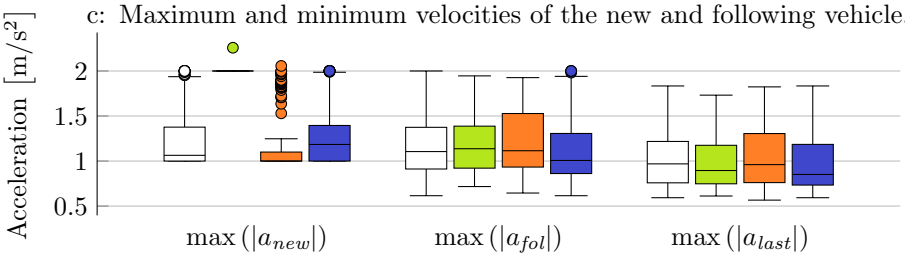
a: Position of the new vehicle in the platoon.



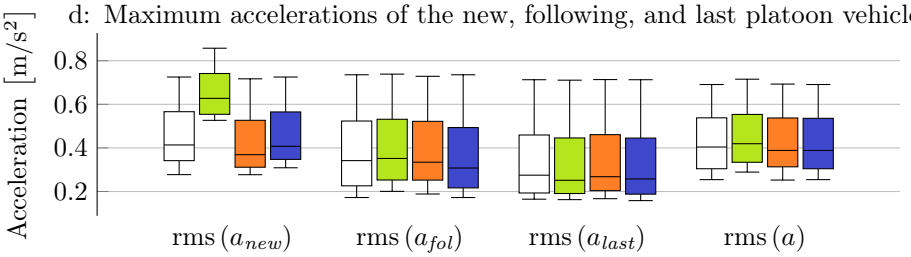
b: Position error after the controller transition.



c: Maximum and minimum velocities of the new and following vehicle.



d: Maximum accelerations of the new, following, and last platoon vehicle.



e: RMS acceleration of the new, following, and last platoon vehicle, and all vehicles.

Figure B.4: The results for 160 simulation parameterizations using the continuous velocity changing scenario for the last vehicle optimization (□), distance-based (■), time-based (■), and conventional optimization (■) sequencing methods.

APPENDIX C

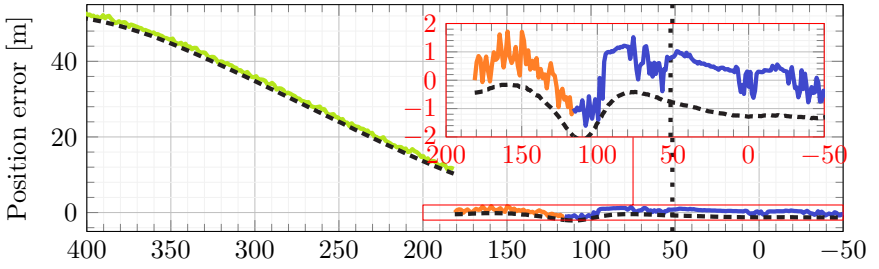
Position Error Comparison for the Experiments

Experiments using two full-scale vehicles are presented in Chapter 6. The position errors in this chapter are based on the post-processed data from both vehicles. This is different from the error as determined by the on-board sensors which is used by the longitudinal controller. To get a better understanding of the merging vehicle's behavior, the position errors as used by the longitudinal controller for all presented experiments are given in this appendix.

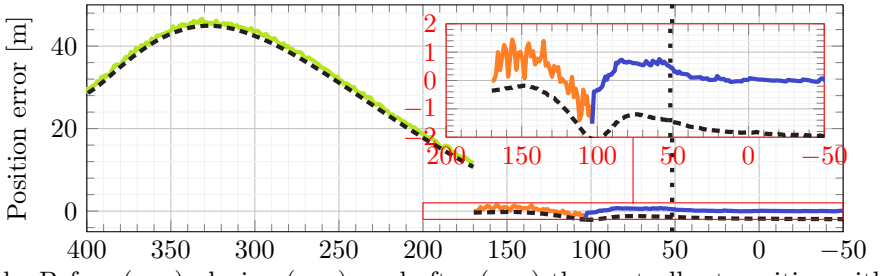
Figure C.1 shows the position error based on the on-board sensors with that of the post-processed data as a reference. The difference between these errors is the computation of the inter-vehicle distance. The on-board sensors use the communicated data and measured radar data to determine the inter-vehicle distance. The post-processed position error is based on the absolute position of both vehicles. One notable difference between the two error values is that the post-processed error always appears to be slightly smaller than that determined by the on-board sensors. The sign of the difference is equal for all experiments but the amplitude changes. Therefore, the difference is likely due to a combination of calibration errors and Global Navigation Satellite System (GNSS) biases. Another apparent difference is that the position error using the on-board sensors is noisier than the post-processed position error. Furthermore, the error based on the on-board sensors appears to jump at certain points in the measurement. This is especially apparent in Figures C.1a and C.1c. These jumps are likely caused by failures to assign the radar data to the target vehicle. If the radar measurement is not recognized as the target vehicle or the target vehicle is outside of the radar range, the communicated data is used to determine the inter-vehicle distance. Due to factors such as calibration errors, GNSS biases, and communication delays, the inter-vehicle distance can be different between the two methods. This difference causes the jumps in the perceived position error.

The figure shows that the controller brings the perceived position error to

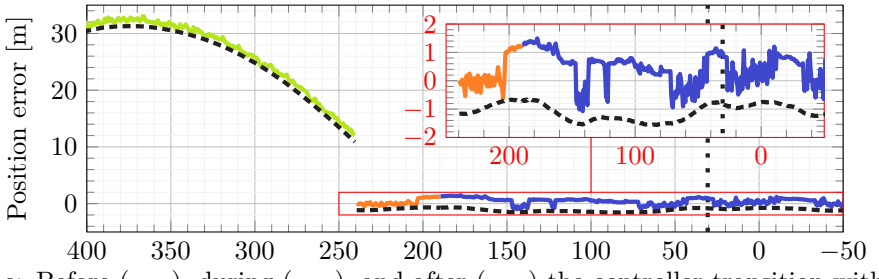
zero. This is especially apparent in Figure C.1b, due to the lack of jumping and low amount of noise in the perceived error. However, in Figure C.1d it is shown that the error does not always manage to reach zero. Especially in the first experiment (—), the post-processed error is closer to zero than that determined using the on-board sensors. This is likely due to the large initial error as discussed in Section 6.3.4 and the limited time available for the error to converge. This underlines the importance of the transitional controller as it illustrates the time required for the error to diminish using the exponentially stable CACC controller. Overall, according to both definitions, the error is well within stand-still distance and the maneuver is deemed safe. Furthermore, the results suggest that improvements in the target tracking module of the vehicles will directly translate to an improvement in the tracking performance of the proposed controller.



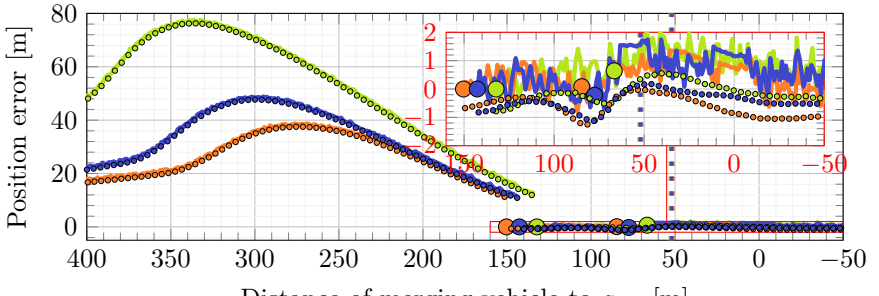
a: Before (—), during (—), and after (—) the controller transition with the constant velocity scenario. Error using post-processed data (- - -).



b: Before (—), during (—), and after (—) the controller transition with the acceleration scenario. Error using post-processed data (- - -).



c: Before (—), during (—), and after (—) the controller transition with the deceleration scenario. Error using post-processed data (- - -).



d: Error using on-board sensors (—) during three experiments with the acceleration scenario. Error using post-processed data (—).

Figure C.1: The position error as measured by the on-board sensors. The position q_{lc} at which the merging vehicle would initiate its lane change is denoted with the vertical lines (- - -) or (- - -).

Bibliography

S. Abuelsamid. Volkswagen adds 'vehicle-to-everything' communications to revamped golf with nxp chips. https://www.forbes.com/sites/samabuelsamid/2019/10/28/volkswagen-includes-nxp-v2x-communications-in-8th-gen-golf/, 2019. Accessed : 18/11/2021.

A. Al Alam, A. Gattami, and K. H. Johansson. An experimental study on the fuel reduction potential of heavy duty vehicle platooning. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pages 306–311, 2010. doi: 10.1109/ITSC.2010.5625054.

A. Alam, B. Besselink, V. Turri, J. Martensson, and K. H. Johansson. Heavy-Duty Vehicle Platooning for Sustainable Freight Transportation: A Cooperative Method to Enhance Safety and Efficiency. *IEEE Control Systems*, 35(6): 34–56, dec 2015. doi: 10.1109/MCS.2015.2471046.

B. Alves Beirigo. *Dynamic Fleet Management for Autonomous Vehicles Learning- and optimization-based strategies*. PhD thesis, Delft University of Technology, 2021.

B. van Arem, J. G. van Driel, and R. Visser. The Impact of Cooperative Adaptive Cruise Control on Traffic-Flow Characteristics. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, 7(4):429, 2006. doi: 10.1109/TITS.2006.884615.

M. Athans. A unified approach to the vehicle-merging problem. *Transportation Research*, 3(1):123–133, apr 1969. doi: 10.1016/0041-1647(69)90109-9.

T. Awal, L. Kulik, and K. Ramamohanrao. Optimal traffic merging strategy for communication- and sensor-enabled vehicles. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, number Itsc, pages 1468–1474. IEEE, oct 2013. doi: 10.1109/ITSC.2013.6728437.

A. Bayuwindra. *Look-ahead tracking controllers for integrated longitudinal and lateral control of vehicle platoons*. PhD thesis, Eindhoven University of Technology, 2019.

A. Bayuwindra, J. Ploeg, E. Lefeber, and H. Nijmeijer. Combined Longitudinal and Lateral Control of Car-Like Vehicle Platooning With Extended Look-Ahead. *IEEE Transactions on Control Systems Technology*, 28(3):790–803, may 2020. doi: 10.1109/TCST.2019.2893830.

M. di Bernardo, P. Falcone, A. Salvi, and S. Santini. Design, Analysis, and Experimental Validation of a Distributed Protocol for Platooning in the Presence of Time-Varying Heterogeneous Delays. *IEEE Transactions on Control Systems Technology*, 24(2):1–1, 2015. doi: 10.1109/TCST.2015.2437336.

A. Boelhouwer. *Exploring, developing and evaluating in-car HMI to support appropriate use of automated cars*. PhD thesis, University of Twente, Netherlands, jan 2021.

A. M. Boggs, R. Arvin, and A. J. Khattak. Exploring the who, what, when, where, and why of automated vehicle disengagements. *Accident Analysis & Prevention*, 136:105406, mar 2020. doi: 10.1016/j.aap.2019.105406.

Bosch Motorsport. Acceleration sensor mm5.10 [online]. https://www.bosch-motorsport.com/content/downloads/Raceparts/en-GB/51546379119226251.html#/Tabs=51582091/. Accessed : 06/04/2022.

J. Caarls. *Pose estimation for mobile devices and augmented reality*. PhD thesis, Delft University of Technology, 2009.

S. C. Calvert, T. H. van den Broek, and M. van Noort. Modelling cooperative driving in congestion shockwaves on a freeway network. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pages 614–619, 2011. doi: 10.1109/ITSC.2011.6082837.

W. Cao, M. Mukai, T. Kawabe, H. Nishira, and N. Fujiki. Gap Selection and Path Generation during Merging Maneuver of Automobile Using Real-Time Optimization. *SICE Journal of Control, Measurement, and System Integration*, 7(4):227–236, 2014. doi: 10.9746/jcmsi.7.227.

W. Cao, M. Mukai, T. Kawabe, H. Nishira, and N. Fujiki. Cooperative vehicle path generation during merging using model predictive control with real-time optimization. *Control Engineering Practice*, 34:98–105, jan 2015. doi: 10.1016/j.conengprac.2014.10.005.

W. Cao, M. Mukai, and T. Kawabe. Merging trajectory generation method using real-time optimization with enhanced robustness against sensor noise. *Artificial Life and Robotics*, 24(4):527–533, dec 2019. doi: 10.1007/s10015-019-00546-w.

H. Chae, Y. Jeong, S. Kim, H. Lee, J. Park, and K. Yi. Design and Vehicle Implementation of Autonomous Lane Change Algorithm based on Probabilistic

Prediction. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2018-Novem:2845–2852, 2018. doi: 10.1109/ITSC.2018.8569778.

R. E. Chandler, R. Herman, and E. W. Montroll. Traffic Dynamics: Studies in Car Following. *Operations Research*, 6(2):165–184, apr 1958. doi: 10.1287/opre.6.2.165.

N. Chen. *Coordination Strategies of Connected and Automated Vehicles near On-ramp Bottlenecks on Motorways*. PhD thesis, Delft University of Technology, 2021.

N. Chen, M. Wang, T. Alkim, and B. van Arem. A Robust Longitudinal Control Strategy of Platoons under Model Uncertainties and Time Delays. *Journal of Advanced Transportation*, 2018:1–13, 2018. doi: 10.1155/2018/9852721.

N. Chen, B. van Arem, T. Alkim, and M. Wang. A Hierarchical Model-Based Optimization Control Approach for Cooperative Merging by Connected Automated Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–14, 2020. doi: 10.1109/tits.2020.3007647.

A. C. Chiang. *Elements of Dynamic Optimization*. McGraw-Hill, 1992. ISBN 9780070109117.

K.-c. Chu. Decentralized Control of High-Speed Vehicular Strings. *Transportation Science*, 8(4):361–384, nov 1974. doi: 10.1287/trsc.8.4.361.

C. A. Desoer and M. Vidyasagar. *Feedback Systems*. Society for Industrial and Applied Mathematics, 2009. doi: 10.1137/1.9780898719055.

D. Dey. *External communication for self-driving cars: designing for encounters between automated vehicles and pedestrians*. PhD thesis, Eindhoven University of Technology, 2020.

K. C. Dey, L. Yan, X. Wang, Y. Wang, H. Shen, M. Chowdhury, L. Yu, C. Qiu, and V. Soundararaj. A Review of Communication, Driver Characteristics, and Controls Aspects of Cooperative Adaptive Cruise Control (CACC). *IEEE Transactions on Intelligent Transportation Systems*, 17(2):491–509, feb 2016. doi: 10.1109/TITS.2015.2483063.

K. Dresner and P. Stone. A Multiagent Approach to Autonomous Intersection Management. *Journal of Artificial Intelligence Research*, 31:591–656, mar 2008. doi: 10.1613/jair.2502.

L. Eiermann, O. Sawade, S. Bunk, G. Breuel, and I. Radusch. Cooperative automated lane merge with role-based negotiation. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 495–501. IEEE, 2020. doi: 10.1109/IV47402.2020.9304711.

S. Fukuyama. Dynamic game-based approach for optimizing merging vehicle trajectories using time-expanded decision diagram. *Transportation Research Part C: Emerging Technologies*, 120:102766, nov 2020. doi: 10.1016/j.trc.2020.102766.

G. Gunter, D. Gloudemans, R. E. Stern, S. McQuade, R. Bhadani, M. Bunting, M. L. Delle Monache, R. Lysecky, B. Seibold, J. Sprinkle, B. Piccoli, and D. B. Work. Are Commercially Implemented Adaptive Cruise Control Systems String Stable? *IEEE Transactions on Intelligent Transportation Systems*, 22 (11):6992–7003, nov 2021. doi: 10.1109/TITS.2020.3000682.

S. Hallé and B. Chaib-draa. A collaborative driving system based on multiagent modelling and simulations. *Transportation Research Part C: Emerging Technologies*, 13(4):320–345, aug 2005. doi: 10.1016/j.trc.2005.07.004.

P. Hang, C. Lv, C. Huang, Y. Xing, and Z. Hu. Cooperative Decision Making of Connected Automated Vehicles at Multi-Lane Merging Zone: A Coalitional Game Approach. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–13, 2021. doi: 10.1109/TITS.2021.3069463.

J. P. Hespanha. *Linear Systems Theory*. Princeton University Press, Princeton, second edition, dec 2018. ISBN 9781400890088. doi: 10.23943/9781400890088.

L. L. Hoberock. A Survey of Longitudinal Acceleration Comfort Studies in Ground Transportation Vehicles. *Journal of Dynamic Systems, Measurement, and Control*, 99(2):76–84, jun 1977. doi: 10.1115/1.3427093.

R. van Hoek. *Cooperative Trajectory Planning for Automated Vehicles*. PhD thesis, Eindhoven University of Technology, 2021.

R. van Hoek, J. Ploeg, and H. Nijmeijer. Gap Closing for Cooperative Driving in Automated Vehicles using B-splines for Trajectory Planning. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 370–375. IEEE, oct 2020. doi: 10.1109/IV47402.2020.9304732.

R. van Hoek, J. Ploeg, and H. Nijmeijer. Cooperative Driving of Automated Vehicles Using B-Splines for Trajectory Planning. *IEEE Transactions on Intelligent Vehicles*, 6(3):594–604, sep 2021. doi: 10.1109/TIV.2021.3072679.

F. Hoogeboom. *Safety of Automated Vehicles: Design, Implementation, and Analysis*. PhD thesis, Eindhoven University of Technology, 2020.

X. Hu and J. Sun. Trajectory optimization of connected and autonomous vehicles at a multilane freeway merging area. *Transportation Research Part C: Emerging Technologies*, 101:111–125, apr 2019. doi: 10.1016/j.trc.2019.02.016.

Z. Huang, D. Chu, C. Wu, and Y. He. Path Planning and Cooperative Control for Automated Vehicle Platoon Using Hybrid Automata. *IEEE Transactions on Intelligent Transportation Systems*, 20(3):959–974, 2019. doi: 10.1109/TITS.2018.2841967.

R. Hult, F. E. Sancar, M. Jalalmaab, A. Vijayan, A. Severinson, M. Di Vaio, P. Falcone, B. Fidan, and S. Santini. Design and Experimental Validation of a Cooperative Driving Control Architecture for the Grand Cooperative Driving Challenge 2016. *IEEE Transactions on Intelligent Transportation Systems*, 19 (4):1290–1301, apr 2018. doi: 10.1109/TITS.2017.2750083.

I-Cave. I-cave - integrated cooperative automated vehicles. https://i-cave.nl/, 2016. Accessed : 16/07/2021.

S. Jing, F. Hui, X. Zhao, J. Rios-Torres, and A. J. Khattak. Cooperative Game Approach to Optimal Merging Sequence and on-Ramp Merging Control of Connected and Automated Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 20(11):4234–4244, 2019. doi: 10.1109/TITS.2019.2925871.

A. Kanaris, E. Kosmatopoulos, and P. Loannou. Strategies and spacing requirements for lane changing and merging in automated highway systems. *IEEE Transactions on Vehicular Technology*, 50(6):1568–1581, 2001. doi: 10.1109/25.966586.

A. Kesting, M. Treiber, M. Schönhof, F. Kranke, and D. Helbing. Jam-Avoiding Adaptive Cruise Control (ACC) and its Impact on Traffic Dynamics. In *Traffic and Granular Flow'05*, pages 633–643. Springer Berlin Heidelberg, Berlin, Heidelberg, jan 2007. doi: 10.1007/978-3-540-47641-2_62.

R. Kianfar, M. Ali, P. Falcone, and J. Fredriksson. Combined longitudinal and lateral control design for string stable vehicle platooning within a designated lane. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1003–1008. IEEE, oct 2014. doi: 10.1109/ITSC.2014.6957819.

R. Kianfar, P. Falcone, and J. Fredriksson. A control matching model predictive control approach to string stable vehicle platooning. *Control Engineering Practice*, 45:163–173, dec 2015. doi: 10.1016/j.conengprac.2015.09.011.

Y. Kim, J. Guanetti, and F. Borrelli. Compact Cooperative Adaptive Cruise Control for Energy Saving: Air Drag Modelling and Simulation. *IEEE Transactions on Vehicular Technology*, 70(10):9838–9848, oct 2021. doi: 10.1109/TVT.2021.3108537.

R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein. The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for

Validation of Highly Automated Driving Systems. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2118–2125. IEEE, nov 2018. doi: 10.1109/ITSC.2018.8569552.

F. Lampel, F. Uysal, F. Tigrek, S. Orru, A. Alvarado, F. Willems, and A. Yarovoy. System Level Synchronization of Phase-Coded FMCW Automotive Radars for RadCom. In *2020 14th European Conference on Antennas and Propagation (EuCAP)*, pages 1–5. IEEE, mar 2020. doi: 10.23919/EuCAP48036.2020.9135417.

E. Lefeber, J. Ploeg, and H. Nijmeijer. Cooperative Adaptive Cruise Control of Heterogeneous Vehicle Platoons. *IFAC-PapersOnLine*, 53(2):15217–15222, 2020. doi: 10.1016/j.ifacol.2020.12.2304.

T. Li, D. Chen, H. Zhou, J. Laval, and Y. Xie. Car-following behavior characteristics of adaptive cruise control vehicles based on empirical experiments. *Transportation Research Part B: Methodological*, 147:67–91, may 2021. doi: 10.1016/j.trb.2021.03.003.

C.-Y. Liang and H. Peng. Optimal Adaptive Cruise Control with Guaranteed String Stability. *Vehicle System Dynamics*, 32(4-5):313–330, nov 1999. doi: 10.1076/vesd.32.4.313.2083.

J. Liu, W. Zhao, and C. Xu. An Efficient On-Ramp Merging Strategy for Connected and Automated Vehicles in Multi-Lane Traffic. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–12, 2021. doi: 10.1109/TITS.2020.3046643.

J. Los, F. Schulte, M. T. Spaan, and R. R. Negenborn. Collaborative Vehicle Routing when Agents have Mixed Information Sharing Attitudes. *Transportation Research Procedia*, 44:94–101, 2020. doi: 10.1016/j.trpro.2020.02.014.

C. Lu and G. Dubbelman. Learning to complete partial observations from unpaired prior knowledge. *Pattern Recognition*, 107:107426, nov 2020. doi: 10.1016/j.patcog.2020.107426.

X.-Y. Lu and J. K. Hedrick. Longitudinal control algorithm for automated vehicle merging. *International Journal of Control*, 76(2):193–202, jan 2003. doi: 10.1080/0020717031000079418.

X.-Y. Lu, H.-S. Tan, S. E. Shladover, and J. K. Hedrick. Automated Vehicle Merging Maneuver Implementation for AHS. *Vehicle System Dynamics*, 41 (2):85–107, jan 2004. doi: 10.1076/vesd.41.2.85.26497.

J. Ma, X. Li, S. Shladover, H. A. Rakha, X. Y. Lu, R. Jagannathan, and D. J. Dailey. Freeway speed harmonization. *IEEE Transactions on Intelligent Vehicles*, 1(1):78–89, 2016. doi: 10.1109/TIV.2016.2551540.

A. M. I. Mahbub, A. A. Malikopoulos, and L. Zhao. Decentralized optimal coordination of connected and automated vehicles for multiple traffic scenarios. *Automatica*, 117:108958, 2020. doi: 10.1016/j.automatica.2020.108958.

V. Milanés and S. E. Shladover. Handling Cut-In Vehicles in Strings of Cooperative Adaptive Cruise Control Vehicles. *Journal of Intelligent Transportation Systems*, 20(2):178–191, mar 2016. doi: 10.1080/15472450.2015.1016023.

V. Milanés, J. Perez, E. Onieva, and C. Gonzalez. Controller for Urban Intersections Based on Wireless Communications and Fuzzy Logic. *IEEE Transactions on Intelligent Transportation Systems*, 11(1):243–248, mar 2010. doi: 10.1109/TITS.2009.2036595.

V. Milanés, J. Godoy, J. Villagra, and J. Perez. Automated on-ramp merging system for congested traffic situations. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):500–508, 2011. doi: 10.1109/TITS.2010.2096812.

V. Milanés, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, and M. Nakamura. Cooperative adaptive cruise control in real traffic situations. *IEEE Transactions on Intelligent Transportation Systems*, 15(1):296–305, 2014. doi: 10.1109/TITS.2013.2278494.

F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli. The e-puck, a robot designed for education in engineering. *Proceedings of the 9th conference on autonomous robot systems and competitions*, 1(1):59–65, 2009.

A. I. Morales Medina, N. van de Wouw, and H. Nijmeijer. Cooperative intersection control based on virtual platooning. *IEEE Transactions on Intelligent Transportation Systems*, 19(6):1727–1740, 2018. doi: 10.1109/TITS.2017.2735628.

M. R. I. Nieuwenhuijze, T. van Keulen, S. Oncu, B. Bonsen, and H. Nijmeijer. Cooperative Driving With a Heavy-Duty Truck in Mixed Traffic: Experimental Results. *IEEE Transactions on Intelligent Transportation Systems*, 13(3): 1026–1032, 2012. doi: 10.1109/tits.2012.2202230.

H. Nijmeijer, J. van der Sar, and T. P. J. van der Sande. *i-Cave: The future of moving forward.* Eindhoven University of Technology, 2021. ISBN 978-90-386-5400-3. URL https://7656071f.flowpaper.com/ICAVELR/.

J. Nilsson, J. Silvlin, M. Brannstrom, E. Coelingh, and J. Fredriksson. If, When, and How to Perform Lane Change Maneuvers on Highways. *IEEE Intelligent Transportation Systems Magazine*, 8(4):68–78, 2016. doi: 10.1109/MITS.2016.2565718.

I. A. Ntousakis, I. K. Nikolos, and M. Papageorgiou. Optimal vehicle trajectory planning in the context of cooperative merging on highways. *Transportation Research Part C: Emerging Technologies*, 71:464–488, oct 2016. doi: 10.1016/j.trc.2016.08.007.

E. van Nunen, M. R. J. A. E. Kwakkernaat, J. Ploeg, and B. D. Netten. Cooperative Competition for Future Mobility. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1018–1025, 2012. doi: 10.1109/tits.2012.2200475.

E. Onieva, V. Milanés, J. Villagrá, J. Pérez, and J. Godoy. Genetic optimization of a vehicle fuzzy decision system for intersections. *Expert Systems with Applications*, 39(18):13148–13157, dec 2012. doi: 10.1016/j.eswa.2012.05.087.

B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli. A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles. *IEEE Transactions on Intelligent Vehicles*, 1(1):33–55, mar 2016. doi: 10.1109/TIV.2016.2578706.

I. Papadimitriou and M. Tomizuka. Fast lane changing computations using polynomials. In *Proceedings of the 2003 American Control Conference, 2003.*, volume 1, pages 48–53. IEEE, 2003. doi: 10.1109/ACC.2003.1238912.

J. Ploeg. *Analysis and design of controllers for cooperative and automated driving.* PhD thesis, Eindhoven University of Technology, 2014.

J. Ploeg, B. T. M. Scheepers, E. van Nunen, N. van de Wouw, and H. Nijmeijer. Design and experimental evaluation of cooperative adaptive cruise control. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 260–265. IEEE, oct 2011. doi: 10.1109/ITSC.2011.6082981.

J. Ploeg, E. Semsar-Kazerooni, G. Lijster, N. van de Wouw, and H. Nijmeijer. Graceful Degradation of Cooperative Adaptive Cruise Control. *IEEE Transactions on Intelligent Transportation Systems*, 16(1):488–497, feb 2015. doi: 10.1109/TITS.2014.2349498.

J. Ploeg, E. Semsar-Kazerooni, A. I. Morales Medina, J. F. C. M. de Jongh, J. van de Sluis, A. Voronov, C. Englund, R. J. Bril, H. Salunkhe, A. Arrue, A. Ruano, L. Garcia-Sol, E. van Nunen, and N. van de Wouw. Cooperative Automated Maneuvering at the 2016 Grand Cooperative Driving Challenge. *IEEE Transactions on Intelligent Transportation Systems*, 19(4):1213–1226, apr 2018. doi: 10.1109/TITS.2017.2765669.

R. Pueboobpaphan, F. Liu, and B. van Arem. The impacts of a communication based merging assistant on traffic flows of manual and equipped vehicles

at an on-ramp using traffic flow simulation. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pages 1468–1473, 2010. doi: 10.1109/ITSC.2010.5625245.

K. Raboy, J. Ma, E. Leslie, and F. Zhou. A proof-of-concept field experiment on cooperative lane change maneuvers using a prototype connected automated vehicle testing platform. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, 25(1):77–92, jan 2021. doi: 10.1080/15472450.2020.1775085.

R. Rajamani. *Vehicle dynamics and control, Second edition.* Springer, 2012. ISBN 9781461414339.

F. Remmen, I. Cara, E. de Gelder, and D. Willemsen. Cut-in Scenario Prediction for Automated Vehicles. In *2018 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, pages 1–7. IEEE, sep 2018. doi: 10.1109/ICVES.2018.8519594.

J. Rios-Torres and A. A. Malikopoulos. A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps. *IEEE Transactions on Intelligent Transportation Systems*, 18(5): 1066–1077, 2017a. doi: 10.1109/TITS.2016.2600504.

J. Rios-Torres and A. A. Malikopoulos. Automated and Cooperative Vehicle Merging at Highway On-Ramps. *IEEE Transactions on Intelligent Transportation Systems*, 18(4):780–789, apr 2017b. doi: 10.1109/TITS.2016.2587582.

O. Sawade, M. Schulze, and I. Radusch. Robust Communication for Cooperative Driving Maneuvers. *IEEE Intelligent Transportation Systems Magazine*, 10 (3):159–169, 2018. doi: 10.1109/MITS.2018.2842241.

W. Schinkel. *Extended automated driving features: An experimental approach.* PhD thesis, Eindhoven University of Technology, 2021. (in preparation).

W. Schinkel, T. van der Sande, and H. Nijmeijer. State estimation for cooperative lateral vehicle following using vehicle-to-vehicle communication. *Electronics (Switzerland)*, 10(6):1–29, 2021. doi: 10.3390/electronics10060651.

W. J. Scholte, P. W. A. Zegelaar, and H. Nijmeijer. Gap Opening Controller Design to Accommodate Merges in Cooperative Autonomous Platoons. *IFAC-PapersOnLine*, 53(2):15294–15299, jan 2020. doi: 10.1016/j.ifacol.2020.12.2327.

W. J. Scholte, P. W. A. Zegelaar, and H. Nijmeijer. A control strategy for merging a single vehicle into a platoon at highway on-ramps. *Transportation Research Part C: Emerging Technologies*, 136:103511, mar 2022. doi: 10.1016/j.trc.2021.103511.

E. Semsar-Kazerooni, J. Verhaegh, J. Ploeg, and M. Alirezaei. Cooperative adaptive cruise control: An artificial potential field approach. *IEEE Intelligent Vehicles Symposium, Proceedings*, 2016-August(Iv):361–367, 2016. doi: 10.1109/IVS.2016.7535411.

E. Semsar-Kazerooni, K. Elferink, J. Ploeg, and H. Nijmeijer. Multi-objective platoon maneuvering using artificial potential fields. *IFAC-PapersOnLine*, 50 (1):15006–15011, 2017. doi: 10.1016/j.ifacol.2017.08.2570.

A. C. Serban, E. Poll, and J. Visser. A Standard Driven Software Architecture for Fully Autonomous Vehicles. In *2018 IEEE International Conference on Software Architecture Companion (ICSA-C)*, pages 120–127. IEEE, apr 2018. doi: 10.1109/ICSA-C.2018.00040.

A. Severinson. An open platform for research and development in intelligent transportation systems. https://raw.githubusercontent.com/rendits/router/master/doc/whitepaper.pdf. Accessed : 06/04/2022.

S. Sheikholeslam and C. A. Desoer. Longitudinal Control of a Platoon of Vehicles with no Communication of Lead Vehicle Information: A System Level Study. *IEEE Transactions on Vehicular Technology*, 42(4):546–554, 1993. doi: 10.1109/25.260756.

S. E. Shladover. Longitudinal Control of Automotive Vehicles in Close-Formation Platoons. *Journal of Dynamic Systems, Measurement, and Control*, 113(2): 231–241, jun 1991. doi: 10.1115/1.2896370.

S. E. Shladover, C. Nowakowski, X. Y. Lu, and R. Ferlis. Cooperative adaptive cruise control: Definitions and operating concepts. *Transportation Research Record*, 2489:145–152, 2015. doi: 10.3141/2489-17.

Society for Automotive Engineers. *SAE J3016 - Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles.* SAE International, 2021. doi: 10.4271/J3016_202104.

S. Stankovic, M. Stanojevic, and D. Siljak. Decentralized overlapping control of a platoon of vehicles. *IEEE Transactions on Control Systems Technology*, 8 (5):816–832, 2000. doi: 10.1109/87.865854.

C. Thiemann, M. Treiber, and A. Kesting. Estimating Acceleration and Lane-Changing Dynamics from Next Generation Simulation Trajectory Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2088(1):90–101, jan 2008. doi: 10.3141/2088-10.

T. Toledo and D. Zohar. Modeling Duration of Lane Changes. *Transportation Research Record: Journal of the Transportation Research Board*, 1999(1):71–78, jan 2007. doi: 10.3141/1999-08.

u-blox. Evk-m8t evaluation kit user guide [online]. https://content.u-blox.com/sites/default/files/products/documents/EVK-M8T_UserGuide_%28UBX-14041540%29.pdf, 2018. Accessed : 06/04/2022.

A. Uno, T. Sakaguchi, and S. Tsugawa. A merging control algorithm based on inter-vehicle communication. In *Proceedings 199 IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems (Cat. No.99TH8383)*, pages 783–787. IEEE, 1999. doi: 10.1109/ITSC.1999.821160.

M. D. Vaio, P. Falcone, R. Hult, A. Petrillo, A. Salvi, and S. Santini. Design and Experimental Validation of a Distributed Interaction Protocol for Connected Autonomous Vehicles at a Road Intersection. *IEEE Transactions on Vehicular Technology*, 68(10):9451–9465, 2019. doi: 10.1109/TVT.2019.2933690.

J. VanderWerf, S. Shladover, N. Kourjanskaia, M. Miller, and H. Krishnan. Modeling Effects of Driver Control Assistance Systems on Traffic. *Transportation Research Record: Journal of the Transportation Research Board*, 1748(1):167–174, jan 2001. doi: 10.3141/1748-21.

S. R. Venkita, D. Willemsen, M. Alirezaei, and H. Nijmeijer. Switching from autopilot to the driver: A transient performance analysis. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 234(5):1346–1360, apr 2020. doi: 10.1177/0954407019878540.

F. Walker. *To trust or not to trust? Assessment and calibration of driver trust in automated vehicles.* PhD thesis, University of Twente, feb 2021.

Z. Wang, G. Wu, P. Hao, K. Boriboonsomsin, and M. Barth. Developing a platoon-wide Eco-Cooperative Adaptive Cruise Control (CACC) system. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1256–1261. IEEE, jun 2017. doi: 10.1109/IVS.2017.7995884.

Z. Wang, G. Wu, and M. Barth. Distributed Consensus-Based Cooperative Highway On-Ramp Merging Using V2X Communications. In *SAE Technical Papers*, volume 2018-April, 2018. doi: 10.4271/2018-01-1177.

Z. Wang, Y. Bian, S. E. Shladover, G. Wu, S. E. Li, and M. J. Barth. A Survey on Cooperative Longitudinal Motion Control of Multiple Connected and Automated Vehicles. *IEEE Intelligent Transportation Systems Magazine*, 12(1):4–24, 2020a. doi: 10.1109/MITS.2019.2953562.

Z. Wang, Y. Wu, and Q. Niu. Multi-Sensor Fusion in Automated Driving: A Survey. *IEEE Access*, 8:2847–2868, 2020b. doi: 10.1109/ACCESS.2019.2962554.

Z. Wang, K. Han, and P. Tiwari. Digital Twin-Assisted Cooperative Driving at Non-Signalized Intersections. *IEEE Transactions on Intelligent Vehicles (Early Access)*, pages 1–1, 2021. doi: 10.1109/TIV.2021.3100465.

S. Wei, Y. Zou, X. Zhang, T. Zhang, and X. Li. An Integrated Longitudinal and Lateral Vehicle Following Control System with Radar and Vehicle-to-Vehicle Communication. *IEEE Transactions on Vehicular Technology*, 68(2): 1116–1127, 2019. doi: 10.1109/TVT.2018.2890418.

I. Wilmink, G. Klunder, and B. van Arem. Traffic flow effects of Integrated full-Range Speed Assistance (IRSA). In *2007 IEEE Intelligent Vehicles Symposium*, pages 1204–1210. IEEE, jun 2007. doi: 10.1109/IVS.2007.4290282.

J. C. Zegers, E. Semsar-Kazerooni, J. Ploeg, N. van de Wouw, and H. Nijmeijer. Consensus-based bi-directional CACC for vehicular platooning. In *2016 American Control Conference (ACC)*, volume 2016-July, pages 2578–2584. IEEE, jul 2016. doi: 10.1109/ACC.2016.7525305.

Y. Zhou, M. E. Cholette, A. Bhaskar, and E. Chung. Automated On-Ramp Merging and Gap Development with Speed Constraints – A State-Constrained Optimal Control Approach. In *2018 Annual American Control Conference (ACC)*, volume 7, pages 4975–4982. IEEE, jun 2018. doi: 10.23919/ACC.2018.8430796.

Y. Zhou, E. Chung, A. Bhaskar, and M. E. Cholette. A state-constrained optimal control based trajectory planning strategy for cooperative freeway mainline facilitating and on-ramp merging maneuvers under congested traffic. *Transportation Research Part C*, 109:321–342, 2019. doi: 10.1016/j.trc.2019.10.017.

List of publications

Peer-reviewed journal articles

- W. J. Scholte, T.P.J. van der Sande, P. W. A. Zegelaar, and H. Nijmeijer. Experimental Demonstration of Platoon Formation using a Cooperative Merging Controller. *In preparation, 2022*
- W. J. Scholte, P. W. A. Zegelaar, and H. Nijmeijer. A control strategy for merging a single vehicle into a platoon at highway on-ramps. Transportation Research Part C: Emerging Technologies, 136:103511, mar 2022. doi: 10.1016/j.trc.2021.103511.
- V. Rodrigo Marco, J. Kalkkuhl, J. Raisch, W. J. Scholte, H. Nijmeijer, and T. Seel. Multi-modal sensor fusion for highly accurate vehicle motion state estimation. Control Engineering Practice, 100:104409, jul 2020. doi: 10.1016/j.conengprac.2020.104409.

Peer-reviewed conference articles

- W. J. Scholte, P. W. A. Zegelaar, and H. Nijmeijer. Gap Opening Controller Design to Accommodate Merges in Cooperative Autonomous Platoons. IFAC-PapersOnLine, 53(2):15294-15299, 2020. doi: 10.1016/j.ifacol.2020.12.2327.
- W. J. Scholte, V. Rodrigo Marco, and H. Nijmeijer. Experimental Validation of Vehicle Velocity, Attitude and IMU Bias Estimation. IFAC-PapersOnLine, 52(8):118-123, 2019. doi: 10.1016/j.ifacol.2019.08.058.

Dankwoord

De bodem van mijn koffiekopje is weer zichtbaar en het 4-jarige PhD traject zit erop. Zoals koffie de aanzet is voor een productieve werkdag, is de PhD het begin van een (hopelijk) leuke carrière. Het echte werk kan nu beginnen, maar niet voordat ik de personen bedank dankzij wie ik mij de afgelopen vier jaar heb kunnen ontwikkelen. Wegens de beperkte ruimte is het helaas niet mogelijk iedereen hier persoonlijk te bedanken, maar een aantal van jullie noem ik graag in het bijzonder.

Henk, bedankt voor het wekken van mijn nieuwsgierigheid en het bieden van deze mogelijkheid. Jouw kritische blik tijdens mijn afstuderen en de voorkeur voor een niet-lineaire waarnemer zorgden bij mij voor het enthousiasme om een PhD te doen. Het door jou georganiseerde i-Cave project, waar ik in terecht ben gekomen, was een ideale plek, mede vanwege het hands-on werk met de Twizys. Diezelfde kritische blik en het bijbehorende scherpe commentaar (op de theorie en het schrijven) hebben deze dissertatie naar een hoger niveau getild.

Peter, bedankt voor al je input en praktische kijk op de zaken. De gesprekken met jou waren altijd interessant, zeker wanneer het een inkijkje in de industrie gaf. Door jou industriële ervaring kwamen er verschillende belangrijke aspecten naar voren in mijn werk, die we vervolgens goed uitgezocht hebben. Daarnaast was het (zeker tijdens het thuiswerken) ook gewoon fijn om wekelijkse meetings te hebben die soms voor het overgrote deel niet over het project gingen.

Tom, bedankt voor al je steun en begeleiding. Jouw intensieve begeleiding voor het i-Cave project en alles daarnaast hebben mij zeer geholpen. Je enthousiasme voor het vakgebied en sporten zijn zeer aanstekelijk. Je betrokkenheid bij de vele i-Cave demonstraties hebben ervoor gezorgd dat dit project als geheel succesvol is geworden.

I would like to thank the other members of my PhD committee, Bart van Arem, Frank Willems, and Paolo Falcone for providing me with constructive feedback on this thesis and taking part in my PhD defense ceremony. Furthermore, I would like to extend this gratitude to the chair Maarten Steinbuch.

Next, I would like to thank everyone involved in the i-Cave project. This includes all the partners and the project members from other sub-projects who

introduced me to different and interesting aspects of connected automated vehicles. Vooral Frans, Robbin, en Wouter bedankt, samen hebben we menig demo en experiment met de Twizys bezorgt. Dit praktische aspect van mijn PhD is iets wat mij heel goed bij zal blijven. Natuurlijk was het werken met de Twizys niet mogelijk geweest zonder Wietse, Erwin, en Gerard. Jullie heel erg bedankt voor jullie inzet in het AES lab en het medemogelijk maken van deze prachtige setup. Also, I would like to thank Bayu, Bram, Dylan, and Roy for helping with the experiments. Furthermore, I would like to thank my students Anne, Prateek, and Bram for their fantastic work.

I would also like to thank all my DSD colleagues for the company during the punctual coffee breaks, lunch walks, 24-hour meetings, futsal matches, and Benelux meetings. I would especially like to thank all of the office mates I have had through the years. In het bijzonder wil ik Robert en Alex bedanken. Met jullie heb ik het langst een kantoor gedeeld. Deze ervaring heeft mij veel geleerd over de grenzen van het toelaatbare bij professionele interacties met collega's. Sven, Koen, and Manuel, thank you for all the joy in the last phases of my PhD. Ook wil ik Geertje en Anouk bedanken voor al hun organisatorische werk binnen onze groep en het mede-creëeren van een geweldige werksfeer.

Daarnaast wil ik de vele mensen die hebben bijgedragen aan de mooie momenten buiten de universiteit bedanken. Te beginnen met iedereen van de Cambrinus stam. Het varen, de feestjes, en overige activiteiten waren altijd een welkome afwisseling van de dagelijkse sleur. Natuurlijk wil ik ook Gino, Guido, Hugo, Merijn, en Timo bedanken. De afspraken voelde altijd vertrouwt, en ik hoop ooit nog eens niet laatste te worden bij het karten.

Ook wil ik mijn familie bedanken. Mark en Timo, bedankt voor het geven van een goed voorbeeld en alle adviezen en hulp. Assandra en Marissa, bedankt dat jullie mijn broers gelukkig maken. Pap en Mam, bedankt voor het creëren van een stabiele basis en dat jullie er altijd voor mij zijn geweest. Opa, bedankt voor het aanwakkeren en aanmoedigen van mijn interesse in techniek. Oma, bedankt voor de gezeligheid, belletjes en mij te leren dom deurdoen.

Di, you have been with me throughout the entire process and even before. We've had amazing travels and experiences throughout the years. I am looking forward to continuing our adventure. Thank you for everything, I love you ♡.

Wout(er)

About the author

Wouter Scholte was born on December 28, 1992, in Nieuwegein, the Netherlands. After finishing his secondary education in 2010 at Anna van Rijn College in Nieuwegein, the Netherlands, he studied Automotive Technology at the Eindhoven University of Technology, in the Netherlands. In 2018 he graduated 'with great appreciation' within the Dynamics and Control group on Vehicle Motion Estimation. His graduation project was performed at Daimler AG in Böblingen under the supervision of Henk Nijmeijer, Igo Besselink, and Vicent Rodrigo Marco.

In February 2018 he started his PhD research in the Dynamics & Control group at the Department of Mechanical Engineering of the Eindhoven University of Technology under the supervision of Henk Nijmeijer and Peter Zegelaar. This research was part of the 'Integrated Cooperative Automated VEhicles' (i-CAVE) project and was supported by the Dutch Organization for Scientific Research (NWO). The main results of this research are presented in this dissertation.