A control strategy for merging a single vehicle into a platoon at highway on-ramps

Citation for published version (APA):

Document license:
CC BY

DOI:
10.1016/j.trc.2021.103511

Document status and date:
Published: 01/03/2022

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
A control strategy for merging a single vehicle into a platoon at highway on-ramps

W.J. Scholte a,*, P.W.A. Zegelaar a,b, H. Nijmeijer a

a Mechanical Engineering Department, Eindhoven University of Technology, PO BOX 513, Gemini-Zuid 0.143, 5600 MB Eindhoven, Netherlands
b Ford Motor Company, Research and Advanced Engineering Europe, Aachen, Germany

A R T I C L E   I N F O

Keywords:
Automotive control
Connected and automated vehicles
Cooperative merging
Highway on-ramps

A B S T R A C T

An important topic of research regarding cooperative platoons is merging vehicles into a platoon at highway on-ramps. This paper 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 validated 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 satisfactory considering safety, efficiency, passenger comfort, and disturbance handling.

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 cooperative automated vehicles (CAVs) drive closely behind each other using their on-board sensors and vehicle-to-vehicle 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 experimentally validated. An extension of this control strategy is made in Scholte et al. (2020). This extension allows for gap opening in the platoon to accommodate a new vehicle joining the platoon. The extended control strategy has only one additional term which can be set to zero for regular driving.

A topic of interest in the field of platooning is merging a new vehicle into an existing platoon at a highway on-ramp. Many of the works regarding the highway on-ramp merging of CAVs are summarized in the survey of Rios-Torres and Malikopoulos (2017a). Furthermore, this work discusses the coordination at intersections since these are similar scenarios. The survey identifies the main distinction between control strategies as centralized and decentralized controllers. 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) 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. Since a switch to a CACC algorithm is not considered in these papers, this problem has not been addressed. The problems are briefly demonstrated in Section 4.2 of this paper.

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 cost of this decision 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. Since the CACC controller is active when the vehicles are in the same lane, this may lead to unsafe situations.

The highway merging problem can also be approached with a fuzzy controller, 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 intersections. 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 crossed. 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 from 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 was necessary, and calculated 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 may seem applicable to the highway on-ramp merging scenario. However, one major difference between the two scenarios is the considered difference in the initial states of the vehicles. In the previously mentioned work, the initial states of the vehicles are relatively close to their desired terminal states during experiments and simulations. This is especially true for the initial velocity. 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. 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) provide 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 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.
This paper proposes a decentralized merging control strategy for CAVs in highway on-ramp environments. More precisely, the control strategy is designed for the merging of a single CAV into a CACC platoon. The strategy is briefly explained in Fig. 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 switches to a gap-opening CACC controller. Then it creates a virtual platoon with the new vehicle using a virtual transitional CACC controller such that it can transition to a regular CACC controller. The new vehicle is initially driving individually and switches to a virtual transitional CACC controller to transition into a steady state platoon. At the end of the maneuver, all vehicles involved form a 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 paper are:

1. A novel control strategy for highway on-ramp merging maneuvers is proposed. The control strategy is a combination of CACC feedback, gap opening, 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 paper 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 of the proposed control strategy is validated and demonstrated using simulations. Measurement noise and communication delay are included. Furthermore, scenarios including speed variations of a platoon leader are analyzed.

This paper starts with a definition of the problem in Section 2. This includes the vehicle model and the proposed controller dynamics of the individual vehicles. In Section 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. 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 5.

2. Problem statement

This section discusses the problem of a cooperative highway on-ramp merging scenario in more detail. Furthermore, the gap-opening cooperative vehicle controller is briefly explained. This controller is discussed in more detail in Scholte et al. (2020).

2.1. Highway on-ramp merging scenario

This section describes the merging maneuver. A graphic representation of the initial situation is provided in Fig. 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 $P$ and a set of new vehicles $\mathcal{N}$. It is assumed all vehicles have a map of the environment such that they know their position with respect to the on-ramp. Moreover, we assume
that each vehicle knows 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 performed 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, the transition between controllers for some of the vehicles involved in the maneuver 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.

### 2.2. Vehicle model and vehicle controller design

The CACC control strategy assumes multiple vehicles driving along a 1-dimensional (1D) 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 Fig. 3.

Each vehicle is assumed to be equipped with a radar 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

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

$$
\dot{d}_i(t) = v_{i-1}(t) - v_i(t),
$$

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

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

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) shows that $q_i$ will converge to $u_i$. The desired acceleration cannot be assumed instantaneously. Thus, the driveline dynamics are simulated using time constant $\tau$. 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 gap opening CACC controller as described in Scholte et al. (2020) is presented. The desired inter-vehicle distance is determined using a constant time gap spacing policy. This distance $d_{i,j}$ is defined as

$$
d_{i,j}(t) = r_i + h_j v_i(t) + \gamma_j(t),
$$
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. The controller error is defined as

\[
e_i(t) = d_i(t) - d_{ij}(t).
\]

(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,

\[
[e_{1j} \quad e_{2j} \quad e_{3j}]^\top = [e_i \quad \dot{e}_i \quad \ddot{e}_i]^\top.
\]

(7)

The error dynamics yield

\[
e_{1j} = d_i - d_{ij} = a_{i-1} - a_i - L_i - r_i - h_i v_i - \gamma_i
\]

(8)

\[
e_{2j} = v_{i-1} - v_i - \gamma_i - h_i a_i
\]

(9)

\[
e_{3j} = a_{i-1} - a_i \left( 1 - \frac{h_i}{\tau} \right) - \gamma_i - \frac{h_i}{\tau} u_i
\]

(10)

\[
\dot{e}_{3j} = - \frac{1}{\tau} e_{3j} - \frac{1}{\tau} e_i + \frac{1}{\tau} u_{i-1},
\]

(11)

where

\[
\xi_i = h_i \dot{u}_i + u_i + \gamma_i + \tau \dot{\gamma}_i.
\]

(12)

A control law for \( \xi_i \) that stabilizes the error dynamics can now be chosen. Using (11) the chosen control law is

\[
\xi_i = k_p e_{1j} + k_d e_{2j} + u_{i-1},
\]

(13)

where scalars \( k_p \) and \( k_d \) are control parameters. Now (12) and (13) yield the control law

\[
\dot{u}_i = \frac{1}{\tau} \left( k_p e_{1j} + k_d e_{2j} + u_{i-1} - u_i - \gamma_i - \tau \dot{\gamma}_i \right).
\]

(14)

It should be noted that the designed trajectory of gap distance \( \gamma_i \) requires \( C^2 \) continuity such that \( \gamma_i \) 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; Scholte et al., 2020).

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 Fig. 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.

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 sequence manager is investigated.

The problem of merging sequence management is outside of the scope of the current work and will be addressed in future work. The initial state of the proposed control strategy in this work 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 will be part of future work 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.

3.2. Controller overview

There are three main steps in the merging maneuver control strategy. These steps are visualized in Fig. 1. In short, the steps are:

1. Vehicle Alignment: When vehicles \( n \) and \( f \) have confirmed their location 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 up 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 vehicle \( n \) and \( f \) have been sufficiently aligned, the lane change is initiated. Vehicle \( n \) merges into the main lane. At this point, it is important for vehicle \( f \) to 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 Fig. 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. In order for vehicle \( n \) to be 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 gap opening CACC controller behind vehicle \( p \). At this moment the vehicle is opening up 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 gap opening controller and used to ensure a timely transition. In essence, the \( \gamma \)-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 \( t_{f,f} \). One constraint is that \( t_{f,n} \) must be before or at \( t_{lc} \). This ensures that a virtual steady-state platoon is obtained during the lateral maneuver. From instance \( t_{0,e} \) up until \( t_{mp} \), 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 control switches to a transitioning CACC algorithm targeting vehicle \( p \) at time \( t_{0,n} \). Then at time \( t_{n,f} \) 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 at each vehicle. To validate the applicability of the proposed control strategy, the information communicated between the vehicles is discussed in Section 3.6.
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 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 Fig. 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 ($\delta_p$) vehicle $n$ travels due to the lane change. $\delta_p$ is calculated before and during the lane change. In Fig. 5, $\delta_q$ 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 $\delta_q$ 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.

3.3.1. 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 its 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 center of the main lane is followed. 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, it is important that the second phase is 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 3.4.

The individual controller aims to have vehicle $n$ arrive at the desired position ($q_{nc}$) and time ($t_{nc}$) at which the designed lane change starts. Position $q_{nc}$ is based on time $t_{nc}$. The value of $t_{nc}$ is chosen such that a reasonable lane change maneuver can be performed before reaching the merging point $q_{mp}$. This value is determined using the time of arrival at $q_{mp}$ ($t_{mp}$) and a predefined time for the lateral movement ($\delta_{lc}$). 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,

$$q_{mp,p} = q_{mp} + L_n + r_n + h_p v_p.$$  

Here $L_n$, $r_n$, and $h_p$ 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 $t_{nc}$ are now calculated using

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

Here

$$t_{nc} = t + \frac{q_{nc} - q_{con}}{v_p}.$$  

The position of the vehicle $n$ is visualized as the longitudinal distance between the position of vehicle $p$ and $n$.

![Fig. 5. The main lane vehicle path (red) and on-ramp vehicle path (blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)](image-url)
where \( r \) 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 \( q_{lc} = \eta_{mp} - \delta_{l,lc} v_p \).

The trajectory of vehicle \( n \) is determined by minimizing the cost function \( J = \frac{1}{2} \int_{0}^{\tau} s(t)^2 dt \). 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

\[
\begin{align*}
q_i^*(t) &= \frac{c_1 t^7}{7!} + \frac{c_2 t^6}{6!} + \frac{c_3 t^5}{5!} + \frac{c_4 t^4}{4!} + \frac{c_5 t^3}{3!} + \frac{c_6 t^2}{2} + c_7 \\
v_i^*(t) &= \frac{c_1 t^6}{6!} + \frac{c_2 t^5}{5!} + \frac{c_3 t^4}{4!} + \frac{c_4 t^3}{3!} + \frac{c_5 t^2}{2} + c_6 + c_7 \\
a_i^*(t) &= \frac{c_1 t^5}{5!} + \frac{c_2 t^4}{4!} + \frac{c_3 t^3}{3!} + c_5 t + c_6 \\
j_i^*(t) &= \frac{c_1 t^4}{4!} + \frac{c_2 t^3}{3!} + \frac{c_3 t^2}{2} + c_4 t + c_5.
\end{align*}
\]

Where \( j_i(t) \) denotes the jerk profile. Coefficients \( c_1 \) to \( c_6 \) 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), control input \( u_i = \tau j_i^* + a_i \) is shown to make the vehicle follow the desired jerk profile. To account for disturbances a model predictive control (MPC) type controller is used. 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). Since \( \tau, a_i, \) and \( u_i \) are assumed to be known, the jerk can be estimated at any time instance.

The desired terminal states are \( q_{n,lc}(t_i) = q_{lc}, v_{n,lc}^* = v_p, a_{n,lc}(t_i) = 0 \) and \( j_{n,lc}(t_i) = 0 \). 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 (18)–(21), the optimal trajectory can quickly be calculated. Analytical expressions for coefficients \( c_1 \) to \( c_6 \) 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 1D \( q \) coordinate is changed for a 2-dimensional (2D) \( x \) 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

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

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 changes (Krajewski et al., 2018; Papadimitriou and Tomizuka; Venkita et al., 2020).

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

\[
x_{lc} = x_{mp} - v_p T_{lc}
\]

where \( x_a \) 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 \( x \)-direction. Consequently, the time of the maneuver \( (\delta_{l,lc}) \) 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 \( n \) is meeting the main lane at \( x_{mp} \).

To make the transformation from the 2-dimensional space to the 1-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 \( \delta_q \) as

\[
\delta_q(x) = L_{lc}(x) - (x_{mp} - x) \quad \forall \ x_{lc} \leq x \leq x_{mp}.
\]

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 \( \delta_{l,lc} = L_{lc}(x_{lc})/v_p \).

3.3.2. Follower vehicle controller design

Vehicle \( f \) is controlled using the gap opening controller of Section 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 \( y \)-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 \( y \)-trajectory at \( t_0 \) are chosen as arbitrary values such that

\[
[y(t_0) \ y'(t_0) \ y''(t_0)]^T = [y_0 \ y_0' \ y_0'']^T.
\]
This allows the trajectory to be redefined at some time when $\gamma$ 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

$$[\gamma(t_{lc}) \  \dot{\gamma}(t_{lc}) \  \ddot{\gamma}(t_{lc})]^{T} = [\gamma_{lc} \ \ 0 \ \ 0]^{T}. \quad (26)$$

Where

$$\gamma_{lc} = v_{p}(t_{lc})h_{n} + L_{n} + r_{n} \quad (27)$$

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 $\gamma$ 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. The terminal states of vehicle $f$ can thus be assumed to be satisfactory if appropriate values for $\gamma$ and its derivatives are chosen.

To satisfy the 4 initial and 4 terminal conditions a seventh-order polynomial is used. This polynomial is similar to the optimal trajectory in (18)-(21). Here $\gamma$ replaces position $q^*$ and its derivatives replace $v^*$, $a^*$, and $j^*$. An MPC-type controller is used to recalculate and adjust the $\gamma$-trajectory. This allows for disturbances from vehicle $p$ to alter the values of $t_{lc}$ and $v_{p}(t_{lc})$ during the maneuver.

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 coming section.

### 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 CACC controller with $\gamma$-factor behind vehicle $p$ to a 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 $\gamma$-factor in the new CACC controller. Initial values for $\gamma$ and its first and second derivatives are chosen such that the perceived initial error states are zero. Next, a suitable $\gamma$-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 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 ($t_{lc}$) and the final ($t_{lc}$) time of the transition can be computed. If there is a value for $t_{lc}$ for which the expected states satisfy the proposed conditions, the transitional controller is started. When $t_{lc}$ is reached the vehicle switched to a regular CACC controller.

### Algorithm 1: General overview of the controller switching logic

```
while Initial controller do
    Compute $\gamma(t_{0}), \dot{\gamma}(t_{0}), \ddot{\gamma}(t_{0})$ using current states;
    Compute $t_0$ 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 $t_{i,min} \leq t_{s,i} - t_{0,i} \leq t_{i,max}$;
    Compute multiple expected trajectories for switching using a discretization of the set of $t_{s,i}$ values;
    if $\exists$ 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 $t_{i,min} + t_{0,i} \leq t_{k}$ then
            Start transition controller with $t_{s,i} = t_{k}$
```

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

### 3.4.1. 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 $\gamma$-factor is used. This allows the vehicle to transition from an initial state to the desired state in a predefined time.
Using this controller a trajectory of $\gamma$ is designed such that the desired trajectories (18)–(21) are followed. Since the gap opening controller is stable we assume the errors $e_{1,n}$, $e_{2,n}$, and $e_{3,n}$ to be zero. Using (8)–(10) this yields

$$u^n = \frac{1}{h} \left( q_p - q^n - L_n - r_n - \dot{\gamma} \right) \tag{28}$$
$$a^n = \frac{1}{h} \left( v_p - v^n - \dot{\gamma} \right) \tag{29}$$
$$j^n = \frac{1}{h} \left( a_p - a^n - \dot{\gamma} \right). \tag{30}$$

Now, the $\gamma$-trajectory is obtained using differential equation

$$\ddot{\gamma} = \frac{a_p - \dot{\gamma}}{h} - \frac{v_p - \dot{\gamma}}{h^2} + \frac{q_p - q^n - L_n - r_n - \dot{\gamma}}{h^3}. \tag{31}$$

It should be noted that (35), (36), and (15) can be used to determine $\gamma$, $\dot{\gamma}$, 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 $q^n_p$ and $j^n_p$ 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 $t_{f,p}$ or earlier. The terminal velocity, acceleration, and jerk are set to match those of vehicle $p$. After the transition, the vehicle proceeds using the 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, the assumption $u_p$ is zero for any future time instances is used. Now, the states of the preceding vehicle are defined as $x_p = [q_p \ v_p \ a_p]^T$. Using (1), (3), and (4) the dynamics can be written as $\dot{x}_p = A_p x_p$ where

$$\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
0 & 0 & -1/\tau_p
\end{bmatrix}.$$

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

$$x_p(t) = e^{A_p(t-t_0)}x_p(t_0). \tag{33}$$

This results in the equations

$$a_p(t) = a_p(t_0)e^{-t_0/\tau_p} \tag{34}$$
$$v_p(t) = v_p(t_0) + a_p(t_0) \left( \tau_p - \tau_p e^{-t_0/\tau_p} \right) \tag{35}$$
$$q_p(t) = q_p(t_0) + v_p(t_0) (t-t_0) + a_p(t_0) \left( \tau_p e^{-t_0/\tau_p} + \tau_p (t-t_0) - \tau_p^2 \right) \tag{36}.$$
\[ a_{n,\text{min}} \leq a_n(t) \leq a_{n,\text{max}} \quad \forall t_{0,n} < t \leq t_{s,n} \tag{41} \]

\[ j_{n,\text{min}} \leq j_n(t) \leq j_{n,\text{max}} \quad \forall t_{0,n} < t \leq t_{s,n} \tag{42} \]

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 \( \gamma \) is introduced. Such that

\[ \gamma_{n,\text{min}} \leq \gamma(t) \leq \gamma_{n,\text{max}} \quad \forall t_{f,\text{min}} < t \leq t_{f,\text{max}}. \tag{43} \]

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

\[ t_{n,\text{min}} \leq t_{n} - t_{0,n} \leq t_{n,\text{max}} \quad \& \quad t_{s,n} \leq t_{l_c}. \tag{44} \]

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

One of the main drawbacks of this strategy lays in (33). This equation requires a knowledge of \( a_p \) which is not directly measurable. Currently, the transmitted desired acceleration \( \bar{a}_p \) 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 \( \bar{a}_p \) is investigated in Appendix B and found to be relatively small.

### 3.4.2. 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 \( \bar{a}_n \) 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 CACC controller without \( \gamma \) factor, the assumption \( \bar{a}_n \) is zero is used.

The control strategies of vehicle \( n \) aim to follow the optimal trajectory as described in (18). This knowledge is used by vehicle \( f \) to design its \( \gamma \)-trajectory. Vehicle \( n \) transmits a vector \( \mathbf{C} \) that contains the coefficients \( c_1 \) to \( c_8 \) of \( \ddot{q}_n \). Regarding the transmission, it is important to note the communication delay. The time of transmission is added to the message such that the current values of the control coefficient can be computed by shifting the polynomial in time. These coefficients are used to predict \( q_n, v_n, \) and \( a_n \). Now differential equation

\[ j_f^\prime = \frac{a_n - \dot{\gamma}}{h} - \frac{v_n - \dot{\gamma}}{h^2} + \frac{q_n - q_f^\prime - L_f - r_f - \gamma}{h^3}. \tag{45} \]

is used to determine the \( \gamma \)-trajectory. This ensures that the trajectory of (18) is followed and the predicted behavior of vehicle \( f \) can be investigated. The equivalent conditions of (40)–(43) 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 (44) is defined as

\[
\begin{cases}
  t_{f,\text{min}} \leq t_{f} \leq \min \left( t_{s,n}, t_{f,\text{max}} \right), & \text{if } \exists t_{s,n} \\
  t_{f,\text{min}} \leq t_{f} \leq \min \left( t_{l_c}, t_{f,\text{max}} \right), & \text{if } \exists t_{l_c}, \text{ and } t < t_{l_c} \\
  t_{f,\text{min}} \leq t_{f} - t_{0,f} \leq t_{f,\text{max}}, & \text{otherwise.} 
\end{cases}
\tag{46}
\]

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 unpredictable behavior of vehicle \( f \). Its \( \gamma \)-trajectory will thus be updated when the communicated terminal time of the validity of \( \mathbf{C} \) changes more than a predefined bound.

### 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 \( p \) may thus be undesirably 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 (RP) or an attractive potential (AP) when the trailing vehicle is too close or too far respectively. The RP causes the trailing vehicle to slow down and the AP causes it to accelerate. When the trailing vehicle is following a vehicle on another
lane, the RP 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 RP is zero. However, when the trailing vehicle is getting too close, the RP causes it to slow down. The CACC control law of (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 \( y \)-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

\[
u_f = \min(u_{f,p}, u_{f,n}).
\] (47)

If the merger 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 the vehicle.

### 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 2D coordinates into a one-dimensional \( q \) coordinate. Moreover, important locations, such as \( q_{map} \), 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 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 \( v_f \) and \( q_p \) as shown in (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 \( L_{at}, r_{at} \) and \( h_u \) in (27). Furthermore, \( \delta_{ic} \) is required to compute \( t_{ic} \). Alternatively, vehicle \( n \) can send its predicted \( t_{ic} \) 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 2D dimensional frame. Therefore, vehicle \( n \) is required to send \( \delta_q \) based on its planned trajectory. This will allow all vehicles to make the 2D to 1D 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. Validation of the control strategy

In this section, the control strategy is validated 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 traditional 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.
4.1. 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 (1)–(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 $r = 0.1$ 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/s² 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.

Reasonable values for the proposed merging control strategy were selected. First, the approximated duration of the lane change $T_{lc}$ is 5 s. This is similar to the behavior of a human driver (Thiemann et al., 2008; Toledo and Zohar, 2007). Next, $t_{n,\text{min}} = t_{f,\text{min}} = 2$ s and $t_{n,\text{max}} = t_{f,\text{max}} = 5$ s is considered for the controller transitions. Furthermore, the $\gamma$-trajectory must be such that it satisfies $a_{i,\text{max}} = -a_{i,\text{min}} = 1.2$ m/s², $j_{i,\text{max}} = -j_{i,\text{min}} = 0.8$ m/s³ and $t_{i,\text{min}} = -0.1$ m \( \forall i \in \{n, f\} \). It should be noted that not all available conditions were used. This tuning could be specified more for implementation in a vehicle when more specific behavior of the vehicles is desired.

To accurately simulate a real-world environment, sensor noise was added. 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/s² 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) 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 (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 s was added to all communicated messages (Hoogeboom, 2020).

This section will continue with a comparison between the proposed strategy and a traditional 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. The results are used to analyze the noise sensitivity.

4.2. Control strategy comparison

This section focuses on the comparison of the proposed strategy to a traditional MPC-based strategy. This controller is similar to that proposed in other literature (Rios-Torres and Malikopoulos, 2017b; Ntousakis et al., 2016). As explained in Section 3.3, at every time step a trajectory based on Eqs. (18) to (21) is planned that fulfills the initial and final conditions. This is done by calculating the values for constants $c_1$ to $c_5$. 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 the planning horizon spans from the current time to $t_{f,\text{max}}$. Therefore, the planning horizon spans from the current time to $t_{f,\text{max}}$ and thus varies throughout the maneuver. For the MPC-based strategy, the control of vehicle $n$ is like that of the proposed method. The only difference is that the MPC-based method does not use the proposed $\gamma$-CACC controller for its transition. The MPC controller is thus used until $t_{f,\text{max}}$ when the control strategy is directly switched to the CACC controller. For vehicle $f$ the $\gamma$-CACC controllers for gap opening and controller transition are replaced with the same MPC controller as vehicle $n$. The difference being the terminal conditions of the trajectory such that all vehicles are appropriately spaced at the end of the platoon. A more general assessment of the proposed method is given in Section 4.3.4.

One important challenge implementing this MPC-based strategy generally does not elaborate on the handling of the small planning horizons when $t_{f,\text{max}}$ is approached. Problems with a small planning horizon for this MPC-based method were encountered in Ntousakis et al. (2016). There, the problem was solved by running an ACC controller on 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 (14). The desired control input is thus not directly computed. To avoid extreme excitations, a saturation of $\pm 1.5$ m/s² was added on $u_i$ during for the MPC controller. Using (4) of the vehicle model it is shown that this limits the acceleration of the vehicle. 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 \( \pm 1 \text{ m/s}^2 \) is deemed suitable.

Fig. 6 shows the position of the vehicles over time. In this simulation, \( a_{\text{mp}} \) is defined as 0. The start of the lane change (\( t_{\text{lc}} \)) is indicated on the \( x \)-axis. It is shown that both strategies align the vehicles at \( t_{\text{lc}} \). The main visible difference between the two strategies is the behavior of vehicle \( f \) before \( t_{\text{lc}} \). It is shown that vehicle \( f \) drives more forward when using the MPC-based strategy. This is visible at for example 8 s into the simulation. It can thus be concluded that there is a difference in the behavior between the two strategies. Furthermore, after \( t_{\text{lc}} \), vehicle \( f \) appears to be further back using the MPC-based approach. This indicates that the strategies affect the behavior of the platoon after \( t_{\text{lc}} \). 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 Fig. 7 are examined. It is shown that the velocities and accelerations after \( t_{\text{lc}} \) remain close to constant for the proposed strategy. However, for the MPC-based strategy, excitations are found after \( t_{\text{lc}} \). These excitations are caused by initial errors in the CACC controller at \( t_{\text{lc}} \). This behavior is undesired since one of the requirements is to achieve a steady-state platoon at \( t_{\text{lc}} \). 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 caused the gap not to be adequately opened at the start. Then a more severe braking action was required at approximately 7 s into the simulation to open the gap. The saturation on \( u_i \) was hit limiting the acceleration of the vehicle. Then the vehicle accelerates to ensure the desired velocity is met at \( t_{\text{lc}} \). However, the attempt is unsuccessful resulting in a too low velocity at \( t_{\text{lc}} \). 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 Fig. 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 for the MPC-based occur after \( t_{\text{lc}} \), which is especially problematic because the vehicles may be in the same lane. The 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 \( t_{\text{lc}} \) is reached. This once more underlines the advantage of the proposed method.

To comment on the driver comfort, the jerk trajectories are examined in Fig. 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 \( \pm 3 \text{ m/s}^3 \) is deemed acceptable (Hoberock, 1977; Shladover, 1991). The peak jerk for the MPC-based strategy reaches peaks of around \( 22 \text{ m/s}^3 \) for vehicle \( n \). This is near \( t_{\text{lc}} \) when the planning horizon is short. Similar behavior was 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 near constant acceleration. However, a jerk of \( \pm 4 \text{ m/s}^3 \) is surpassed by vehicle \( f \) which is outside of the comfort bounds. The jerk of the proposed method stays within approximately \( \pm 1 \text{ m/s}^3 \) and thus well within the comfort bounds. Regarding passenger comfort, the proposed method is thus also favorable.

A summary of the comparison is given in Table 2. The proposed strategy outperforms the MPC-based strategy for most performance indicators. Most importantly, the errors for the proposed method are small which shows the additional safety provided by the proposed method. It should be noted that only errors after \( t_{\text{lc}} \) 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 2 shows that with the proposed strategy the bounds \( a_i \leq \pm 1.2 \text{ m/s}^2 \) and \( j_i \leq \pm 0.8 \text{ m/s}^3 \) are surpassed. This underlines the importance of choosing conservative values for these limits.

![Fig. 6](image_url) The position of vehicle \( p (\quad) \), \( n (\quad) \), and \( f (\quad) \) during a merge maneuver for the proposed controller (top), and the MPC-based strategy (bottom).
Fig. 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 (- - - -).

Fig. 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.

Fig. 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.

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, the proposed strategy is thus applicable for real-world scenarios and beneficial over existing MPC-based algorithms.
Table 2
A comparison of the two strategies, the errors after time instance $t_{lc}$ are examined.

<table>
<thead>
<tr>
<th></th>
<th>Proposed strategy</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$e_f$</td>
<td>$e_s$</td>
<td>$e_f$</td>
<td>$e_s$</td>
<td>$a_f$</td>
<td>$a_s$</td>
</tr>
<tr>
<td>Max</td>
<td>0.067</td>
<td>0.061</td>
<td>0.019</td>
<td>0.017</td>
<td>1.147</td>
<td>1.426</td>
</tr>
<tr>
<td>Min</td>
<td>−0.025</td>
<td>−0.009</td>
<td>−0.041</td>
<td>−0.028</td>
<td>−0.876</td>
<td>−0.047</td>
</tr>
<tr>
<td>RMS</td>
<td>0.017</td>
<td>0.019</td>
<td>0.013</td>
<td>0.010</td>
<td>0.430</td>
<td>0.676</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>MPC-based strategy</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$e_f$</td>
<td>$e_s$</td>
<td>$e_f$</td>
<td>$e_s$</td>
<td>$a_f$</td>
<td>$a_s$</td>
</tr>
<tr>
<td>Max</td>
<td>5.061</td>
<td>0.041</td>
<td>1.526</td>
<td>0.147</td>
<td>2.049</td>
<td>1.426</td>
</tr>
<tr>
<td>Min</td>
<td>−0.100</td>
<td>−0.685</td>
<td>−0.998</td>
<td>−0.615</td>
<td>−1.499</td>
<td>−0.712</td>
</tr>
<tr>
<td>RMS</td>
<td>2.284</td>
<td>0.328</td>
<td>0.525</td>
<td>0.152</td>
<td>0.882</td>
<td>0.689</td>
</tr>
</tbody>
</table>

Fig. 10. The position of vehicle $p$ (---), $n$ (----), and $f$ (-----) during a merge maneuver for the acceleration (top) and deceleration (bottom) scenario.

4.3. Lead vehicle excitation handling

The previous section showed the advantage of the proposed strategy over a traditional MPC-based strategy. This section discusses the performance of the proposed strategy under excitations of the lead vehicle. The main focus of this section 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 analysis of the longitudinal behavior the jerk is examined. This gives insight into the perceived user comfort. The section is concluded with a short analysis of the lateral behavior by examining $\delta_q$.

The position of the vehicles over time for the three scenarios is plotted in Fig. 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 this the velocities and accelerations of the vehicles are examined.

The velocities and accelerations are shown in Fig. 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, it can be seen that the accelerations remain between $\pm 2 \text{ m/s}^2$. 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 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 Fig. 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 vehicles $f$ and $n$ can thus be established first. 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 $a_f$ in Fig. 11(b) shows that the collision avoidance intervention is used in the deceleration scenario. At approximately 7 s into the simulation, vehicle $f$ approaches vehicle $p$ too rapidly and the controllers 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 Fig. 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 $\pm 0.3 \text{ m/s}$, which is an acceptable magnitude.
Fig. 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 (----).

Fig. 12. The error and its derivative for vehicle $n$ (---) and $f$ (--).

Overall, the behavior of the error dynamics is thus deemed desirable. Especially after $t_{lc}$, the error is 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 should be noted that the controller transition aims at an error of zero at the initialization by choosing appropriate values for $\gamma$ 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 Fig. 13. It should be noted that the error definition of vehicle $f$ changes during this maneuver. Between approximately 7 and 9 s, 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 (8), a positive error means that vehicle $f$ is too far behind vehicle $p$. The collision avoidance controller thus successfully keeps a distance to vehicle $p$. The controller interrupted because vehicle $f$ was approaching too rapidly. It is thus 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.

The jerk of the vehicles for both scenarios is shown in Fig. 14. It is shown 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
Fig. 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.

Fig. 14. The jerk of vehicle $n$ (---) and $f$ (---).

Fig. 15. The predicted additional length $\delta_q$ due to the lateral movement during the acceleration (---), constant velocity (---), and deceleration (---) scenario.

sufficiently initialized such that it does not create any peaks in the jerk. It can thus be concluded that the controller transition is relatively smooth. Overall, the passenger comfort during this maneuver is thus deemed 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 $n$ estimates an additional distance $\delta_q$ it needs to drive to achieve this. The behavior of the predicted additional distance $\delta_q$ can be found in Fig. 15. The value of $\delta_q$ 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 $\delta_q$ increases or decreases respectively. This variation in $\delta_q$ is caused by the fact that a time duration is prescribed for the lane change. Thus, the distance traveled at higher velocities is larger. Therefore, the additional distance due to the lateral movement smaller.

The figure also shows that $\delta_q$ is noisy before $t_{lc}$. This is caused by the noisy radar measurements which are used to determine the path. After $t_{lc}$, 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 $\delta_q$ much smoother. Lastly, it should be noted that the lane change takes around 5 s. At a velocity of 100 km/h this results in a trajectory of approximately 139 m. The value of $\delta_q$ and 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 is satisfactory. 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 C 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. The effect of noise on the behavior of the vehicle is investigated in Section 4.5.

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 Fig. 16. At approximately 14 s 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 Fig. 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 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 s) is 15 m. When the gap is closed the resulting overshoot is 0.4 m. This is a reasonable amplitude and after \( t_{lc} \) the error is within 0.2 m. The collision avoidance controller, therefore, appears to behave adequately.

Lastly, to assess the performance of the collision avoidance controller, the velocities and accelerations are briefly analyzed. In Fig. 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. 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.

### 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 4.1. 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 Fig. 19(a). 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 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, $t_{sf}$ has a large variety with a maximum of 3.26 s 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 Fig. 19(b), a summary is provided in Table 3. The acceleration of vehicle $n$ peaks at 1.24 m/s$^2$. Overall, the distribution of the jerk for different noise signals appears relatively uniform. The jerk behavior is therefore acceptable, and the controller is sufficiently unaffected by the noise.

The position errors for vehicles $f$ and $n$ are shown in Fig. 19(d). Both vehicles have a maximum error of over 0.6 m. However, it should be noted that the error is generally largest just after the controller switch. This is because the estimation of appropriate $\gamma$ values is sensitive to noise. The error dampens over time and therefore the errors will be smaller after time $t_{lc}$.

The position error after $t_{lc}$ is shown in Fig. 19(e). This is an important performance indicator because, at $t_{lc}$, vehicle $n$ initiates its lateral movement. After this time the vehicles may thus 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. Next, an error in the estimation of $a_n$ to initialize $\gamma$ for the controller transition in Appendix B. The appendices show that the proposed control strategy can sufficiently handle these disturbances.

5. Conclusion and future work

This paper proposes a control strategy for the cooperative merging of a single cooperative automated vehicle into a platoon of vehicles. The platoon is assumed to be driving with a traditional CACC algorithm outside of the maneuver. Therefore, the control strategy at a vehicle level is based on the 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 proposed control strategy is validated 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 traditional MPC-based strategies in such circumstances. Furthermore, the response to excitations of a platoon leader is analyzed, which showed satisfactory results. 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 indicated that the proposed control strategy is robust and applicable in real-world scenarios.

In future work, 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, experiments concerning the proposed controller transition approach may be performed in future work. This will better validate the noise handling and real-world applicability of the proposed strategy.

CRediT authorship contribution statement

**W.J. Scholte**: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Visualization. **P.W.A. Zegelaar**: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration. **H. Nijmeijer**: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition.
Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is part of the research program i-CAVE with project number 14893, which is partly financed by the Netherlands Organisation for Scientific Research (NWO), Netherlands.

Appendix A. Analysis of changing the initial position of the new vehicle

In this appendix, the influence of the initial position of vehicle $n$ is investigated. In essence, the vehicle can be 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 future work when a merging sequence strategy can be used.

Due to size constraints of this paper, 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 $q_{b,n} = 0.5(v_p h_p + L_p + r_p)$ forward or backward.

In Fig. 20 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 $\gamma$-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 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 $2 \text{ m/s}^2$. This should be just within the range of a high-end passenger vehicle. This underlines the importance of selecting the correct merging sequence.

Appendix B. 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 in 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 $\pm 1 \text{ m/s}^2$ 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 $\gamma$ and its derivatives. The effect of this disturbance can be seen in Fig. 21. The effect on the velocity and acceleration is less significant.
An error of $\pm 1 \text{ m/s}^2$ is relatively large compared to the working range of a passenger vehicle. Fig. 21 shows that the effect of this disturbance on the error is relatively small. The controller van handle this disturbance well and over time the error signals of the two scenarios 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 Fig. 22. 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.

Appendix C. 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 $1 \text{ m/s}^2$ 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 disturbance. Both simulations thus pose different challenges.

The velocities and accelerations of the vehicles during these simulations are shown in Fig. 23. 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 $2 \text{ m/s}^2$ is reasonable considering that an acceleration of $1 \text{ m/s}^2$ 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 Fig. 24. 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. 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 real world disturbances.
Fig. 23. 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 (- - - -).

Fig. 24. The position error of vehicles n (---) and f (-----) during the sinusoidal oscillation scenarios. The circles indicate the controller transitions.

References


Wouter Scholte is a Ph.D. candidate within the Dynamics and Control group at the Eindhoven University of Technology. He received an M.Sc. degree in automotive technology from the Eindhoven University of Technology in 2018. His research interests include vehicle dynamics, vehicle control, and control systems for cooperative and automated vehicles. His current work is centered around the controller mode transitions in cooperative dual mode automated vehicles. The main focus lays on highway on-ramp merging scenarios.

Peter Zegelaar is Technical Expert Driver Assistance Functions at the Ford Research and Innovation Center in Aachen, Germany. He was active in the field of collision avoidance: automated braking and automated steering to avoid collisions. He is now working on highly automated driving and cooperative driving. Furthermore, he is part-time professor ”Smart Vehicle Technology” at the Eindhoven University of Technology, in the Netherlands.

Henk Nijmeijer (1955) is a full professor at Eindhoven and chairs the Dynamics and Control group. He is an editor of Communications in Nonlinear Science and Numerical Simulations. He is a fellow of the IEEE since 2000 and was awarded in 1990 the IEE Heaviside premium. He is appointed honorary knight of the ‘golden feedback loop’ (NTNU, Trondheim) in 2011. He was an IFAC Council Member in the period 2011–2017. Per January 2015 he is scientific director of the Dutch Institute of Systems and Control (DISC). He is recipient of the 2015 IEEE Control Systems Technology Award and a member of the Mexican Academy of Sciences. He is Graduate Program director of the TU/e Automotive Systems program. He is an IFAC Fellow since 2019.