Technische Universität München
Lehrstuhl für Robotik, Künstliche Intelligenz und Echtzeitsysteme

Motion Planning of Autonomous Vehicles with Safety Guarantees

Silvia Breşug

Vollständiger Abdruck der von der Fakultät der Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:	Prof. Dr. Helmut Seidl
Prüfer der Dissertation: 1.	Prof. Dr.-Ing. Matthias Althoff
2.	Prof. Dr.-Ing. Klaus Diepold

Die Dissertation wurde am 27.07.2020 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 17.02.2021 angenommen.

Ich versichere, dass ich diese Doktorarbeit selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

München, May 27, 2021

Silvia Breşug

Abstract

This thesis proposes a novel framework for motion planning with safety guarantees for autonomous vehicles with a focus on highway traffic scenarios.

To achieve this safety guarantee, we first introduce an architecture for the longitudinal control of the ego vehicle that ensures safety with respect to all relevant vehicles by computing sufficient inter-vehicle distances. The required space between vehicles permits an emergency maneuver that can bring the ego vehicle to a safe state, thus making this approach intrinsically safe. To assess the efficacy and efficiency of our presented methodology, we compare our results with state-of-the-art approaches via numerical simulations.

We then introduce a new approach to generate the over-approximation of all possible occupancies of surrounding traffic participants over time. Whilst full braking of vehicles driving ahead is assumed as a worst-case scenario for the longitudinal motion of the ego vehicle, in order to allow the ego to perform more complex maneuvers, this assumption alone does not suffice anymore. Therefore, the prediction of future occupancies is realized by employing techniques from reachability analysis on an abstract model of a vehicle. The introduced approach is evaluated and validated using real traffic data showing that the recorded trajectories are indeed enclosed by the predicted occupancy sets.

Finally, we introduce a fail-safe motion planner that guarantees safety with a novel control scheme that is similar to a safety net. A long-term optimal trajectory is generated by considering non-formal predictions of the surrounding vehicles. This trajectory is followed only if there exists an emergency maneuver capable of steering the ego vehicle to a safe state. If no further fail-safe maneuver exists, the motion planner switches to a precomputed emergency maneuver that guarantees safety by embedding the overapproximative sets of the surrounding vehicles as system constraints. Furthermore, the motion planner introduces an approach to determine the latest point in time where an emergency maneuver is guaranteed to exist.

Zusammenfassung

Diese Dissertation schlägt ein neuartiges Konzept für die Bewegungsplanung von autonomen Fahrzeugen mit Sicherheitsgarantien vor. Der Fokus liegt dabei besonders auf Autobahnszenarien.

Um die Sicherheitsziele zu erreichen, führen wir zunächst eine Architektur für die Längssteuerung des Ego-Fahrzeugs ein. Diese gewährleistet die Sicherheit aller relevanten Fahrzeuge, indem ausreichend große Abstände zwischen den Fahrzeugen berechnet werden. Aufgrund der Abstände gibt es ein Notfallmanöver, das das Ego-Fahrzeug in einen sicheren Zustand bringen kann, wodurch unser Ansatz intrinsisch sicher wird. Um die Effektivität und Effizienz unserer vorgestellten Methode zu bewerten, vergleichen wir in numerischen Simulationen unsere Ergebnisse mit dem Stand der Technik.

Danach führen wir einen neuen Ansatz ein, der eine Überapproximation aller möglichen Belegungsmengen anderer Verkehrsteilnehmer im Laufe der Zeit berechnet. Während wir für die Längsbewegung des Ego-Fahrzeuges die Vollbremsung vorausfahrender Fahrzeuge als Worst-Case-Szenario annehmen, reicht diese Annahme nicht für komplexere Manöver des Ego-Fahrzeugs aus. Deshalb wird die Prädiktion der zukünftigen Belegungsmengen durch den Einsatz von Techniken aus der Erreichbarkeitsanalyse an einem abstrakten Fahrzeugmodell realisiert. Der vorgestellte Ansatz wird unter Verwendung realer Verkehrsdaten bewertet und validiert. Dabei wird gezeigt, dass die aufgezeichneten Trajektorien tatsächlich in den vorhergesagten Belegungsmengen eingeschlossen sind.

Schließlich stellen wir einen ausfallsicheren Bewegungsplaner vor, der die Sicherheit mit einem neuartigen Regelungsschema garantiert, das einem Sicherheitsnetz ähnelt. Eine langfristig optimale Trajektorie wird erzeugt, indem nicht-formale Prädiktionen der umgebenden Fahrzeuge berücksichtigt werden. Diese Trajektorie wird nur dann verfolgt, wenn es ein Notfallmanöver gibt, das das eigene Fahrzeug in einen sicheren Zustand bringen kann. Falls kein weiteres ausfallsicheres Manöver vorhanden ist, wechselt der Bewegungsplaner zu einem vorberechneten Notfallmanöver, das die Sicherheit garantiert, indem die überapproximierten Mengen der umgebenden Fahrzeuge als Randbedingungen verwendet werden. Außerdem beinhaltet der Bewegungsplaner einen Lösungsansatz zur Identifizierung des spätesten Zeitpunkts zu dem ein Notfallmanöver noch garantiert ist.

Acknowledgements

First and foremost I would like to express my deepest gratitude to my PhD adviser Prof.Dr.-Ing. Matthias Althoff for offering me the opportunity to work under his guidance on this challenging research topic. Clearly, this thesis would not have been possible without his supervision, valuable advice and support he gave me throughout my PhD studies. He is an inspiration to me. Additionally, I gratefully acknowledge the funding received from the Graduiertenkollegs PUMA program, managed by Prof. Dr. Helmut Seidl.

To my colleagues from the Cyber-Physical Systems group - Aaron, Esra, Albert, Bastian, Carmella, Andrea, Ahmed, Dongkun, Sebastian, Markus, Stefanie, Niklas, Stefan, Moritz, Felix - thank you for providing such a great working atmosphere! Special thanks goes to Christian for facilitating the usage of the driving simulator from the BMW Group, to Sebastian for performing the user study, to Zhenzhang Ye for our fruitful collaboration, and to my office mates Mingchuan and Sina. Ute and Amy, thank you for the help provided for various administrative tasks at the Institute of Robotics, Artificial Intelligence and Embedded Systems.

I would also like to thank to my colleagues from Audi AG for their encouragement and support.

My deepest gratitude goes to my parents Ecaterina and Eugen, and my sister Andreea, to whom I am forever indebted for their love, encouragement, and faith they always had in me.

Finally, I would like to thank to Răzvan, for his support and continuous encouragement. I am so lucky to have you by my side!

Silvia Breşug

Munich, July 2020

Contents

List of Tables

List of Figures

Chapter 1

Introduction

One of the biggest technology innovations to date is represented by advancements in autonomous driving, which can bring considerable enhancements towards the quality of life. Approximately 1.35 million people die each year as a result of road traffic crashes[1], out of which human error represents the root cause in more than 90% of these situations [169]. Increased road safety, traffic flow, mobility, and comfort are among the most prominent advantages to be introduced by automated driving, by partially or completely removing driving duties from human drivers [10]. Therefore, autonomous vehicles represent the future of the transportation industry due to their massive potential of improving safety. As a result, autonomous vehicles are expected to become the most viable means of transportation by 2040 and are predicted to account for more than 75% of the cars on the roadway[2].

While this innovative technology brings a great improvement in mobility, it also raises considerable concern since it has a direct social impact [126]. Among the most critical topics raised are data security, liability, engagement with insurance companies after an accident, legal aspects, and one of the foremost topics, safety. The latter topic begs the following questions: *Can the autonomous vehicles improve safety on roads? Can safety be guaranteed for passengers?* These questions represent just a few of the frequently arising concerns regarding this new technology.

Both automobile and technology companies have made remarkable achievements in autonomous driving research and have already demonstrate their technological advancements (see Fig. 1.1), massive potential, as well as their encountered challenges [43, 116, 201]. Due to the integration of advanced driver assistance systems (ADAS), many vehicles available on the market are already equipped to support drivers in various situations such as lane-keeping or emergency braking assistance. These systems have already demonstrated their ability to enhance the comfort and safety of passengers.

[1] https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries

[2] https://site.ieee.org/itss/2014/09/15/you-wont-need-a-drivers-license-by-2040/

In particular, adaptive cruise control systems (ACC) — as described in the corresponding intelligent transportation system standard [82] — enjoys increasing popularity since it automates the longitudinal control of vehicles in uncomfortable or frequently occurring situations, e.g., stop-and-go in traffic jams or long-lasting highway journeys. Various ACC systems have been proposed in the literature over the years, aiming to improve different aspects of existing systems such as driving comfort, traffic flow [79, 80], fuel efficiency [2], and travel time [20].

Figure 1.1: Different prototypes of self-driving vehicles[1].

Since fully autonomous vehicles relieve humans of all driving duties, the safe operation of vehicles in dynamic and uncertain environments must be guaranteed by considering both uncertainties introduced by sensors and actuators and those introduced by the unexpected future behavior of other traffic participants. On the one hand, the uncertainty introduced by sensors and actuators can be decreased by either explicitly modelling this uncertainty in the planning problem, or by using high performance hardware, that produces more accurate results. On the other hand, guaranteeing safety while considering unexpected maneuvers of other traffic participants when planning a trajectory for the ego vehicle still represents a challenging task for the automotive industry.

Figure 1.2: A Google car involved in an accident, due to misinterpretation of behavior of another traffic participant [2].

For instance, one of the first accidents caused by a self-driven Google car (see Fig. 1.2) occurred due to a misinterpretation of the future behavior of the other traffic participants. Still, predicting the actions

[1] These images are the property of Google LLC.

[2] Source: www.bbc.com/news/technology-35800285

of other traffic participants is not a trivial task, even for human drivers — for example, the trained test driver of the Google car could not accurately foresee the actions of the other vehicle involved in the collision.

Nevertheless, a reliable autonomous vehicle must be capable of predicting and coping with unexpected maneuvers of other traffic participants. Thus, the driving decisions of such autonomous systems must guarantee the safety of passengers and other traffic participants, before they are deployed into a mixed traffic situation, that is, with both human-driven and self-driven vehicles, which represents the main theme of this thesis.

1.1 Standards and Legal Background Regarding Safety

Although impressive results have already been achieved by experimental autonomous vehicles, safety remains the most significant challenge the autonomous driving industry is facing. Currently, when an autonomous vehicle is tested on public roads, a safety driver is required in order to bring the vehicle to a safe state in case of a failure. However, in the future, a fully autonomous vehicle must always be able to keep the system in a *safe state*. Absolute safety, i.e., the guarantee that an autonomous vehicle will never be involved in an accident, is not possible [168] since the road is shared with other agents as well (e.g., human-driven vehicles whose driving decisions cannot be controlled). What is attainable, however, is the guarantee that an autonomous vehicle will not cause an accident and will try to avoid or mitigate a possible collision when feasible.

According to the ISO 26262 standard [83], a system is said to operate in a safe mode when there is no unreasonable risk. In this standard, the risk is calculated as a combination of the severity of a personal injury and the probability of occurrence. The challenge, however, is to identify a risk level that is considered acceptable based on the current driving situation, according to moral, ethical, and social aspects [126, Chapter 23]. Nonetheless, this standard tackles only safety hazards caused by software or hardware failures and not the ones provoked by erroneous driving decisions. In early 2019, a new standard has been released — Safety Of The Intended Functionality (SOTIF) [84] — which is concerned with guaranteeing safety in the absence of a fault in the system. However, the SOTIF standard does not provide a solution such that the system can guarantee safety, but instead provides a set of guidelines (e.g., recommendations regarding design or validation, etc.) that help to achieve safety.

One of the most important safety-related aspects regarding driving decisions (for both human-driven and self-driven vehicles) represents the required inter-vehicle distance, often called the *safe distance*, that must be kept with respect to the other traffic participants. In the German Road Traffic Regulation (Straßenverkehrsordnung) [34, Section 4] this distance is defined as follows: “a vehicle moving behind another vehicle must, as a rule, keep a sufficient distance from that other vehicle such that it is able to pull up safely even if it suddenly slows down or stops.” However, the description of this traffic rule regarding safe distance is not a formal one. As a consequence, it is challenging, if not almost impossible, to check the correctness of the resulting behavior of an autonomous vehicle.

Recently, a new research direction — formalizing traffic rules [157, 159] — has gained more attention as this represents the first step towards compliant and safe autonomous cars. For such systems, formal guarantees are compulsory in order to prove their correctness — such as obeying the traffic regulations — before being deployed on roads. However, how to formally guarantee safe motion planning that can correctly react to unexpected changes in the traffic environment is not a trivial task.

1.2 Scope and Objectives

In early 2014, the Society of Automotive Engineers (SAE) established a classification of the level of autonomy [161] for self-driven vehicles (as illustrated in Fig. 1.3) that ranges between 0 (No Automation) and 5 (Full Automation). The first three levels correspond to vehicles requiring drivers. Some driving functions, however, are taken over by the system. Conditional automation corresponds to a system where the driver represents a necessity and must be able to take control of the vehicle at any time. In contrast to the automation levels 1—3, level 4 and 5 (high automation and full automation) correspond to a system where the driver is not required; however, the driver does retain the option of controlling the vehicle, if desired.

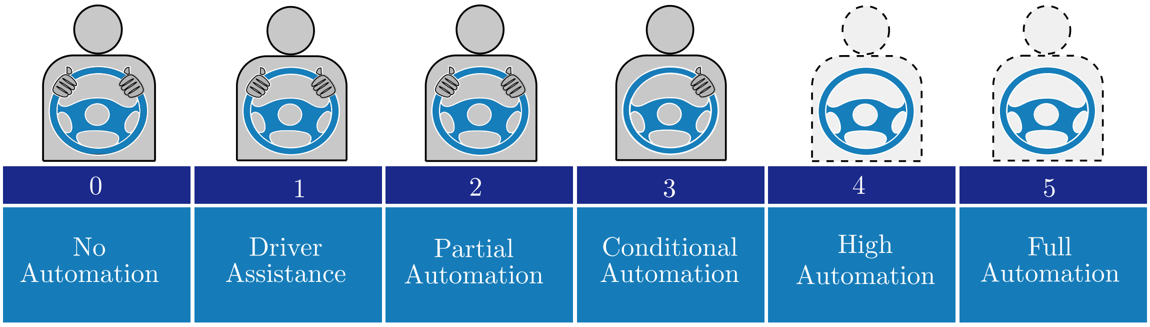


Figure 1.3: Levels of autonomy according to SAE[1].

In this thesis, we focus on the two highest levels of automation (high and full automation), starting from a partially automated system — equipped with Adaptive Cruise Control (ACC) functionality. As mentioned previously, in order to deploy such systems where the driver is not required to take control of the vehicle, safety must be guaranteed in any traffic scenario. However, since an infinite number of traffic scenarios exists, classical test-drives are not enough to prove that an autonomous vehicle will always make the correct decision. Studies have shown that in order to prove the safety of autonomous vehicles, hundreds of millions of test-drives are required [86]. Hence, tens of years of testing would be necessary

[1] Source: www.nhtsa.gov/technology-innovation/automated-vehicles-safety

to validate the reliability of self-driven vehicles. Therefore, a different validation paradigm is essential to guarantee the safety of autonomous vehicles.

This thesis addresses the problem of generating provable safe motion planning for autonomous vehicles in a mixed traffic scenario that includes both human-driven and self-driven vehicles. For this purpose, fail-safe maneuvers are generated through certifiable methods that can bring the ego vehicle to a safe state under various scenarios including the worst-case scenario, where other traffic participants perform unexpected maneuvers that may endanger the ego vehicle. The key element in guaranteeing safety is to consider not only the most likely behavior of other traffic participants, but the entire set of possible future maneuvers when planning a trajectory for the ego vehicle. Furthermore, the generated maneuvers of the ego vehicle must not be too conservative such that traffic flow is hindered.

1.3 Thesis Outline

Throughout this thesis, each chapter contains a relevant literature review, followed by a problem formulation. Then, the proposed solution is provided, together with numerical and experimental results that validate our methods.

Safe Longitudinal Control of Autonomous Vehicles

The current advanced driving assistance systems that provide longitudinal control of vehicles cannot ensure safety in all traffic situations. This is mainly caused by inaccurate assumptions of the behavior of other vehicles driving ahead, which lead to an erroneous estimation of the required safe distance. The main contribution of Chapter 2 is to design a longitudinal controller with safety guarantees for autonomous vehicles, while still satisfying comfort. The first novelty of our approach is in the computation of the required safe distance with respect to the vehicles driving ahead of the ego vehicle. The correct safe distance is calculated such that an emergency maneuver that can bring the ego vehicle to a safe state is guaranteed to exist, even in the worst-case scenario where the leading vehicle performs full brake. In contrast to other approaches that generate only one trajectory based on the most likely behavior of the leading vehicle, we generate two trajectories: one that tries to achieve comfort and efficiency, while trying to keep the safe distance, and a second one that is able bring the ego vehicle to a safe state when the assumptions made of the behavior of other traffic participants no longer hold.

Another novel feature introduced by our approach is that we do not only consider the closest leading vehicle when calculating an acceleration command, but all traffic participants with respect to which the ego vehicle might violate the safe distance or those that can violate the safe distance themselves by performing cutting-in maneuvers. Hence, we prohibit the violation of the safe distance by the ego vehicle, with respect to all vehicles driving ahead.

Two different approaches are proposed for the generation of the acceleration of the ego vehicle with respect to the vehicles that are driving in the ego lane and the ones that change lanes towards the ego lane. For the first category of vehicles (the ones driving in the ego lane), we first calculate the correct inter-vehicle distance that is required in order to safely react, even when a leading vehicle is performing a full brake (as required in [34, Section 4]). Then, a mechanism of engaging a precomputed emergency maneuver, in the situation that the safe distance cannot be kept anymore, is presented. For the second category of selected vehicles (the ones changing lanes towards the ego lane), if the safe distance is violated by cutting-in vehicles, an acceleration profile is calculated for the ego vehicle such that it can clear the violated safe region in a given time horizon. Finally, a decision algorithm is presented that selects which acceleration command among the set of generated ones satisfies the safe distance corresponding to all selected surrounding vehicles. To assess the performance regarding safety and comfort of the proposed framework numerical experiments were conducted using the NGSIM data [65].
Parts of the contributions of Chapter 2 are published in [121].

Overapproximative Occupancy Set Computation of Traffic Participants

To introduce the ability of an autonomous system to perform complex maneuvers that are provable safe, the entire range of possible future behavior of other traffic participants must be considered. In contrast to most existing work, where probabilistic-based approaches are used to predict collision probability with other traffic participants, we propose a formally guaranteed method to compute the overapproximative occupancy of other traffic participants. In contrast to the longitudinal control with safety guarantees approach where it suffices to consider the worst-case behavior of other traffic participants in order to prevent rear-end collisions, the entire range of physical possible maneuvers a vehicle can perform corresponding to the current road network is considered in Chapter 3. In other words, this is a requirement such that the ego vehicle can autonomously perform other maneuvers rather than just keeping the current lane.
Due to the nonlinear dynamics of other traffic participants and the consideration of corresponding constraints (e.g., excluding behaviors that violate traffic regulations, such as leaving the road boundary), standard techniques for reachability analysis cannot be applied to calculate a set-based prediction of other traffic participants. In [7] it was shown that the set-based occupancy prediction of the surrounding vehicles can be calculated in a formal manner. In this chapter, the previous work is extended by providing a framework that can rigorously compute the overapproximative occupancy prediction of the surrounding traffic participants on an arbitrary road network. To validate the proposed method, two sets of evaluations were performed. First, we have compared the corresponding predicted occupancy sets against the real trajectories of vehicles taken from the NGSIM data [65]. Then, we have demonstrated that our overapproximative results are tight by comparing the results with a high-fidelity model.
The contributions from Chapter 3 are published in [10].

Fail-Safe Motion Planning

One of the most critical challenges facing motion planners in autonomous driving is guaranteeing that the ego vehicle will not reach an inevitable collision state, which could be caused by either an inaccurate assumption of the behavior of dynamical obstacles (i.e., the other traffic participants) or by a faulty response to changes in the traffic scenario. Chapter 4 addresses the problem of developing a fail-safe motion planner capable of generating optimal trajectories that can guarantee safety at all times. The optimality is achieved by considering the most probable maneuvers of the other traffic participants. Similar to the longitudinal motion planning with safety guarantees, the trajectory is followed, only if it is verified as safe. A trajectory is verified as safe, only if its occupancy sets do not intersect the corresponding occupancy of the other traffic participants, and, at the end of this trajectory, there exists an emergency maneuver that can safely bring the ego vehicle to a standstill.

While verifying the existence of such an emergency maneuver at each time step guarantees safety, it may be computationally too expensive, and oftentimes it is not required. Therefore, we provide an algorithm that calculates the maximum time horizon during which the ego vehicle can safely follow a given optimal trajectory, with the guarantee that a safe emergency maneuver exists at the end of this time horizon.

To demonstrate the efficiency of our presented algorithm, we test it against real traffic data taken from NGSIM data [65]. In doing so, we slightly modify the traffic situations to further challenge the motion planner. Then, we calculate the latest point in time where an emergency maneuver is guaranteed to exist. Chapter 4 is based on contributions that are published in [120] and [122].

Chapter 2

Safe Longitudinal Control of Autonomous Vehicles

One of the main challenges associated with advanced driving assistance systems is the guarantee that the vehicle can be brought to a safe state even when faced with unexpected, dangerous driving situations. In particular, when the safe distance is erroneously approximated based on an incorrect assumption of the future behavior of other traffic participants, this may lead to a possible imminent collision state if a dangerous situation occurs (e.g., the leading vehicle performs an unexpected emergency maneuver). We first present a method to calculate the correct inter-vehicle distance, which guarantees that if the distance is maintained, an emergency maneuver exists such that the ego vehicle can be brought to a safe state. The second novelty compared to the most existing work on motion planning is that we do not only consider a single target vehicle when calculating the required safe distance, but we select a set of relevant traffic participants that may affect the safety of the ego vehicle. In this way, we prevent imminent collisions resulting from situations where the target vehicle is rapidly changing[1].

The remainder of this chapter is organized as follows: In Sec. 2.1, a review of the related literature is presented, together with the contributions brought by this chapter. Then, in Sec. 2.2, preliminary definitions and notations used throughout the chapter are introduced. After the problem statement is formulated, the method for computing safe longitudinal control is presented by sequentially introducing different traffic scenarios based on the relevant selected traffic participants (see Sec. 2.3). Lastly, in Sec. 2.4, the experimental results based on recorded traffic data are presented to demonstrate the performance and improvement of the novel longitudinal planning approach compared to the previous work.

[1] For instance, the closest leading vehicle changes the current lane due to a slow or static obstacle which forces the ego vehicle to suddenly react to a new target vehicle.

2.1 Introduction

Adaptive Cruise Control (ACC) systems can be particularly useful to improve safety by partially removing driving duties from human drivers. Although most ACC systems try to consider safety explicitly, they fail to ensure safety in complex scenarios, such as sudden emergency braking maneuvers of preceding vehicles or cut-ins of vehicles from adjacent lanes. However, these kinds of situations do occur in real traffic and their resulting collisions can cause severe or fatal injuries, unless the human driver intervenes such that a collision is avoided or mitigated.

Thus, to achieve autonomy levels 4 and 5 (as illustrated in Fig. 1.3), the required step forward is to guarantee vehicle safety during the longitudinal motion of an autonomous vehicle in arbitrary traffic scenarios. Except for the typical driving scenario where the ego vehicle must keep the desired velocity and the desired distance with respect to the leading vehicle, other traffic situations may occur where the safe distance is jeopardized. For example, the safe distance may be violated by surrounding vehicles that perform cut-in maneuvers, or when the closest leading vehicle rapidly changes due to a slower vehicle driving ahead that forces the ego vehicle to perform an emergency maneuver. Therefore, if other traffic participants perform dangerous maneuvers that can endanger the safety of the ego vehicle, an appropriate reaction is required to avoid or mitigate a potential collision. In the following, a review of the related literature is presented.

2.1.1 Related Work

Systems that provide longitudinal control of a vehicle (such as, cruise control, adaptive cruise control, or cooperative adaptive cruise control systems[1]) are already receiving a lot of attention from automobile companies. Each of these systems is characterized by key aspects that define the functionality that the corresponding system is intended for, as illustrated in Fig. 2.1.

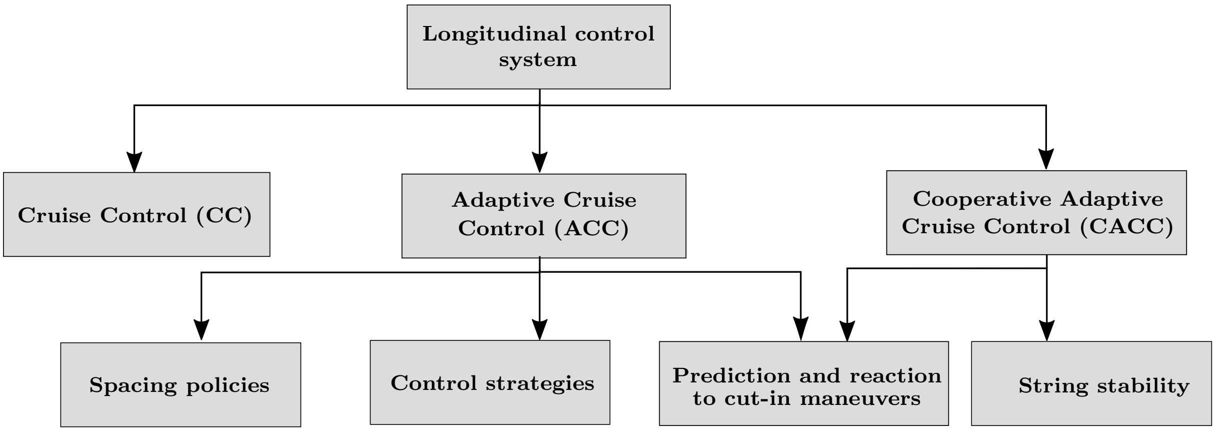

Longitudinal control system
Cruise Control (CC)
Adaptive Cruise Control (ACC)
Cooperative Adaptive Cruise Control (CACC)
Spacing policies
Control strategies
Prediction and reaction to cut-in maneuvers
String stability

Figure 2.1: Key aspects of various longitudinal control systems.

[1] Cooperative adaptive cruise control systems are mainly used for truck platooning.

Cruise Control. Computing a longitudinal acceleration command to maintain a desired velocity/inter-vehicle distance has been exhaustively studied. Systems like conventional cruise control (CC), which automatically control the speed of a vehicle to a user-defined set value, are widely used in vehicles nowadays with the aim of achieving fuel economy but also improving driving comfort on highways [67,148].

Whilst cruise control systems have simpler requirements compared to other longitudinal control functionalities, adaptive cruise control is a more complex system that relies on a precise set of specifications that must be satisfied. Next, we present some of the key aspects that define an adaptive cruise control system, that are also in the focus of this chapter.

Adaptive Cruise Control (ACC). To go one step further than CC, to improve not only driving comfort, but also safety on roads, active safety systems have been developed. The goal of active safety systems (e.g., adaptive cruise control, lane departure prevention, brake assist, or electronic stability control) is to avoid collisions, and when this is not possible, to mitigate the effects of a possible collision. Previous studies have shown that these systems are indeed able to improve safety on roads, by decreasing the number of traffic accidents [156].

In this chapter, however, we will solely focus on adaptive cruise control. Adaptive cruise control is a system that provides longitudinal control of the ego vehicle such that a required safe distance is kept with respect to the leading vehicle, as defined in the corresponding ISO standard 156222 [82], by assuming that pedestrians and non-motorized vehicles are prohibited on the road (e.g., highway traffic environment). In addition to the primary purpose of keeping the safe distance, ACC can improve traffic flow and driving comfort [79, 80]. Besides improving traffic flow and comfort, ACC systems can also reduce fuel consumption [2] and trip time [20]. An extensive survey on ACC systems can be found in [182] and [197].

Spacing policies for ACC. How to compute the required safe inter-vehicle distance is of utmost importance when designing an ACC system, because if it is not properly defined, a collision with the leading vehicle may be imminent. In the German Road Traffic Regulations [34, Sec. 4] it is stated that "[...] a vehicle moving behind another vehicle must, as a rule, keep a sufficient distance from that other vehicle to be able to pull up safely even if it suddenly slows down or stops." However, this definition is not a formal one. Although developing ACC systems is a mature research field, how to guarantee safety if the leading vehicle performs a sudden emergency maneuver, while avoiding jerky maneuvers is not a trivial task.

In [128], an intelligent cruise control system is presented, where a constant time headway is kept. In order to cope with emergency situations, a strategy for switching from cruise control to emergency control was developed [128]. When a hazardous situation occurs, the emergency controller is activated, comparing the safe braking distance for both the target and ego vehicle. On the one hand, if the leading vehicle fully brakes until it stops, but the ego vehicle brakes with a less deceleration power compared to the leading vehicle, a collision may be imminent if the current inter-vehicle distance is not large enough. On the other hand, if the ego vehicle strongly decelerates to reestablish the violated safe distance, this would cause high jerk values, and thus, uncomfortable driving, although the situation may quickly resolve.

Control strategies for ACC. In previous works, different approaches for verifying collision avoidance by using various space control policies have been presented [125,162,178,200]. Several acceleration profiles utilized to maintain a given safe inter-vehicle distance, have also been proposed in [138,195]. However, in order to maintain a safe inter-vehicle distance and comfortable drive, there must be a trade-off between these two aspects.

Another approach for the control of a vehicle such that a given safe inter-vehicle distance is satisfied uses correct-by-construction control software synthesis [142,143]. Within this paradigm, the specifications for the adaptive cruise control system are given as Linear Temporal Logic (LTL) formulae. To design a controller that satisfies the given specification, a discrete abstraction of the system is performed. However, the finite abstraction computation is expensive, and the size of the resulting graph used to design the controller is exponential in terms of the dimensions of the considered system and the length of the LTL behavior specification.

To increase safety and traffic throughput, game theory-based methods have been investigated as well [118, 181]. In these approaches, each vehicle is considered an agent and the controller design is seen as a game between the actions of each agent and the disturbances introduced by the environment. Nevertheless, this approach has exponential complexity in the number of agents.

The ACC problem can also be addressed using control barrier functions [15,129]. These functions are used to penalize any violation of the constraints that arise from ACC specifications. Briefly, this method exploits the control barrier function property that as the value of the function approaches infinity, the points grow closer to the boundaries of the safe region (i.e., the safe distance becomes too short). Yet, finding an appropriate control barrier function is not a trivial task and it is limited by system dimensions.

A widely used method to tackle adaptive cruise control problems is the Model Predictive Control (MPC) framework, which uses its capability of handling multiple constraints in a receding horizon fashion [22,38,112,127,136,172]. At each time instant, a finite-time open-loop optimal control problem is solved. Then, the solution to the optimization problem provides the optimal control input from which only the first part is executed before new sensor values are available. A benchmark setup is proposed in [39], which assesses different model predictive control methods used for ACC. An overview of constraint MPC can be found in [119]; for a comprehensive survey on MPC with constraints, the reader is referred to [127].

Hereafter, we primarily focus on the previous work on ACC that uses MPC since this work is most closely related to our approach. In [22], a two-mode ACC system that utilizes MPC, in which controllers shift between speed control (transitional operation) and distance control (steady-state operation), is presented. The optimization problem is solved subject to desired inter-vehicle distance and acceleration limitation, which are incorporated as constraints. In [112], in order to increase tracking capabilities and fuel economy, an optimal control law is applied. There, a constant time headway spacing policy is used to keep a safe distance between vehicles. The aim of the control problem addressed in [38] is to minimize the distance between two vehicles. It is assumed that the ego vehicle receives the future reference state of the leading vehicle at each sample time. However, if the leading vehicle suddenly brakes, the ego vehicle may not stop within the given safe distance.

This leads to another approach on ACC, developed in [183], where a collision avoidance strategy is integrated. The desired acceleration of the ego vehicle is computed based on the forward spacing error and velocity. However, no analysis on the variation of the spacing error is performed as the focus was on the variation of the velocity.

Cooperative Adaptive Cruise Control. To benefit from new technologies that enable communication between vehicles, the idea of cooperative adaptive cruise control was developed [145, 172]. *String stability*, i.e., the capacity of minimizing the tracking errors in the upstream direction of convoys, which is one of the most crucial properties of a platoon, is addressed in [37,153,198,200,204]. An essential component in a cooperative architecture (platoon) represents inter-vehicle communication, i.e., all entities within the cooperative team know the future trajectory of the others. However, if the communication is lost (e.g., due to environmental conditions) and one of the vehicles performs an unexpected maneuver, such as fully braking, a collision is likely inevitable.

Prediction and reaction to cut-in maneuvers. Up until now, only typical traffic scenarios have been taken into account, where the ego vehicle considers only the closest leading vehicle, driving in the same lane. However, this information is not enough to guarantee safety for an autonomous vehicle, even when concerned with travel only in the longitudinal direction. For example, in the situation where another traffic participant performs a lane change towards the ego lane aiming for the inter-vehicle gap between the ego vehicle and the leading vehicle, the ego vehicle should promptly react such that if the cutting-in vehicle fully brakes, a collision could be avoided or at least mitigated.

Therefore, in order to consider a cut-in maneuver when controlling the ego vehicle, the cutting-in maneuver first must be recognized. Thus, methods that reliably detect if a surrounding vehicle performs a cut-in maneuver are required. Previous work that considers cut-in maneuver detection has been carried out in [21, 88, 109, 133, 202]. The most common approaches use neural networks [55], hidden Markov models [47], Bayesian networks [58, 110], and support vector machines [17]. In [170], a survey on vision-based vehicle future behavior analysis is presented.

The importance of considering cut-in maneuvers performed by human-driven vehicles, especially in platooning scenarios, is investigated in [18]. There, a comparison between the performance of two different approaches, ([155, Chapter 7] and [152]) is performed by examining different aspects, e.g., control strategies and available information. In [81], the effects of cut-in maneuvers in mixed traffic are analyzed, which mainly focuses on position error disturbances and fuel consumption. There already exists work that considers integrating the intended cut-in maneuver into calculation of the required acceleration of the ego vehicle in order to increase the performance of the ACC systems regarding smoothness and safety. For example, in [135], an approach that incorporates cut-in predictions into the longitudinal control of the ego vehicle is presented. If a cut-in maneuver is detected, the cut-in vehicle is now considered as a leading vehicle and the ego vehicle updates its acceleration corresponding to the current traffic situation. However, no further explanation of how the acceleration should be updated is presented. If a possible cut-in is detected (given the current traffic scenario, it is expected that a surrounding vehicle would perform a cut-in maneuver), then the velocity of the ego vehicle decreases until the ego vehicle is slightly slower than the potential cut-in vehicle. However, this may lead to an increase in the number of cut-in maneuvers.

The approach described in [91] considers a situation where a surrounding vehicle may perform a cut-in maneuver to enter a vehicle platoon. As a result, a hard brake may be required in order to react to the sudden lane change. This problem is addressed by developing a stochastic model predictive controller with the aim of minimizing the spacing error. There, the probability of a cut-in maneuver is calculated, which, in turn, is used to update at each time instance the input acceleration. However, it is not clear how the safe distance can still be satisfied or how fast the violated area can be cleared in an emergency situation. Similar to [91], in [203], the probability of a cut-in maneuver is calculated under the assumption that the leading vehicles, as well as the cutting-in vehicle, are driving with constant acceleration. The calculated probability value is then fed to a MPC-based controller.

The scenario where a vehicle cuts in front of a convoy formation is considered in [115]. A model predictive approach is used to control the vehicle convoy by predicting the lane change trajectory of the cutting-in traffic participant. Although the cost function used to generate the control inputs for the vehicle convoy integrates the prediction of the other traffic participant behavior, safety cannot be guaranteed due to the possibility of unexpected maneuvers of the surrounding vehicles.

Often, achieving efficient and comfortable driving, while still guaranteeing safety with respect to the future possible behavior of other traffic participants are contradictory requirements. In general, current longitudinal control schemes only consider rather simple traffic scenarios, e.g., the situation when the first leading vehicle performs an emergency maneuver. If the first leading vehicle is performing a lane change because there is a static obstacle ahead, it is possible that no feasible control input exists such that the ego vehicle is able to avoid a collision. Therefore, not only the closest leading vehicle must be considered when generating an input for the ego vehicle, but also vehicles driving further out in the ego lane.

Moreover, the driving situation where the inter-vehicle distance is suddenly reduced due to another third vehicle performing a cut-in maneuver, that aims for the inter-vehicle distance between the ego and leading vehicle has to be considered such that the ego vehicle can restore the safe distance as soon as possible.

2.1.2 Contributions

The following contributions are introduced in this chapter:

- One of the major contributions of this chapter is the design of a control scheme similar to a safety net, which consists of a *nominal controller*, supervised by an *emergency controller*, which is activated only when the nominal control is no longer feasible. This novel control scheme can guarantee safety at all times by considering unexpected behavior of the other traffic participants. Comfort is also achieved by introducing a new braking profile for the emergency maneuver.
- In order to guarantee *safety*, a correct *safe inter-vehicle distance* based on different emergency braking profiles is computed. Multiple braking profiles used for generating the emergency maneuvers are analyzed, and a discussion is held on how the emergency profile design influences the safe inter-vehicle distance and the jerk.
- In addition to the typical leading-follower vehicle setup, where the ego vehicle must keep a safe inter-vehicle distance from the vehicle driving ahead in the same lane, other traffic participants that are driving in the adjacent lanes are also considered. If a surrounding vehicle initiates a lane change aiming at the gap between the ego vehicle and its leading vehicle, leading to the violation of the safe distance, an acceleration command is calculated for the ego vehicle to clear the violated area.
- Several tests considering multiple scenarios, including a comparison with a standard ACC, were performed in order to assess our approach.

This chapter is based on previous work that was already published in [121].

2.2 Preliminaries and Problem Formulation

Here, some preliminaries and notations used throughout this chapter are presented.

We refer to the driving sector, which is constrained by a left and right boundary, as a *lane*. We denote the ego vehicle by E, i.e., the vehicle to be controlled. Throughout this thesis, the term of “ego vehicle” and “host vehicle” will be used interchangeably. The surrounding vehicles $V_{\square,i}$ represent the other traffic participants, driving in the proximity of the ego vehicle, where $\square \in \{\text{left}, \text{right}, \text{lead}\}$, $i \in \{1, 2, \cdots, N_\square\}$, and $N_\square$ is the number of the detected surrounding vehicles in different lanes, depending on the sensor capabilities of the ego vehicle. The lower the subscript index i, the closer the vehicle is with respect to the ego vehicle. Note that this thesis does not include any sensor performance analysis.

In the following, a highway traffic scenario is considered. In this chapter, only the longitudinal movement of the ego vehicle along a given lane is examined. We assume that the desired lane is given by a tactical planner. For now, the lane change maneuvers of the ego vehicle are not considered.

Definition 2.1 (Occupancy of a vehicle)**.** The occupancy $\Gamma(V, t) : \{E, V_{\square,1}, \cdots, V_{\square,\mathrm{N}}\} \times \mathbb{R}_+ \to P(\mathbb{R}^2)$, represents the rectangle that encloses the body of a vehicle V, at time t. The occupancy $\Gamma(V, t)$ is computed based on the position and the orientation at time t, and the size (length and width) of the vehicle; Uncertainties in the measurements can be considered as well. □

In the following definitions, the □ subscript from the notation $V_{\square,i}$ will be omitted, for readability purposes.

Definition 2.2 (Inter-vehicle distance)**.** The inter-vehicle distance $\delta(V_\mathrm{i}, V_\mathrm{j}, t)$, $i \neq j$, between vehicles V_i and V_j at time t, is defined as the minimum displacement between their corresponding occupancies. □

Definition 2.3 (Safe distance)**.** The safe distance $d_\mathrm{safe}(V_\mathrm{i}, V_\mathrm{j}, t)$ represents the minimum inter-vehicle distance that must be kept between vehicles V_i and V_j, $i \neq j$, such that a collision can still be avoided through braking in the event that the vehicle driving ahead initiates an emergency maneuver. □

2.2.1 Assumptions

Throughout this chapter, the following assumptions are made:

1. No communication between the vehicles (V2V) or vehicle to infrastructure (V2I) is assumed. Therefore, the ego vehicle does not know the future velocity and acceleration of the surrounding vehicles. The entire set of data regarding the environment and the other traffic participants are acquired and then predicted by the ego vehicle.

2. The current information regarding the other traffic participants is acquired through the ego vehicle's sensors (e.g., see [87]). Note that the performance analysis of a vehicle's sensor (e.g., LIDAR, laser, camera) is beyond the scope of the work presented in this thesis. For information regarding the performance of sensors, the reader is referred to [170].

3. An efficient state estimation method is available, and it provides an accurate measurement of the ego state.

4. The maximum deceleration capability of other traffic participants can be estimated (for example, by differentiating them into different classes, as motorbike, truck, car, etc., or by considering the friction coefficient of the road surface the vehicle is driving on[1].)

5. In this chapter, we assume that a high-level planner is available, and it requests keeping the current lane for the ego vehicle.

6. In the event that a surrounding vehicle initiates a cut-in maneuver (i.e., a lane change maneuver aimed at the gap between the ego vehicle and another vehicle driving ahead) that violates the safe inter-vehicle distance, the ego vehicle does not accelerate until the safe distance is restored.

2.2.2 Vehicle System Dynamics

In the following, the mathematical model for a vehicle following setup is derived, as illustrated in Fig. 2.2. For simplicity, we illustrate only one leading vehicle, V_{lead}.

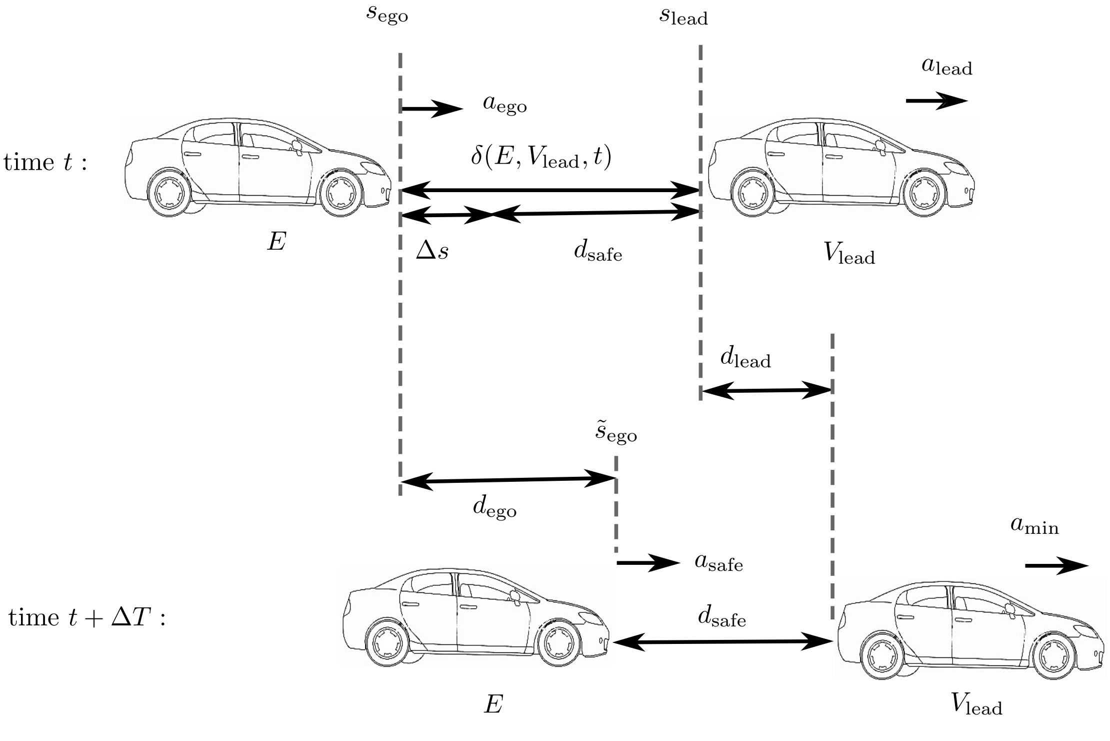


Figure 2.2: Vehicle following setup.

Both the ego and leading vehicle are described by their absolute position, s_{ego} and s_{lead}. The position s_{ego} represents the front of the ego vehicle and s_{lead} the rear of the leading vehicle. The current absolute velocity v_{ego} and v_{lead} and the absolute acceleration a_{ego} and a_{lead} are acquired through the ego vehicle's

[1] The deceleration capability of a vehicle driving on a road covered in ice is reduced compared to the situation where the vehicle is driving on a dry road.

sensor set. The measured inter-vehicle distance between the ego and leading vehicle, $\delta(E, V_{\text{lead}}, t)$, is calculated as $\delta(E, V_{\text{lead}}, t) = s_{\text{lead}} - s_{\text{ego}}$.

A rather simple model is used to design the controller for the ego vehicle:

$$\begin{gathered}
\dot{x} = Ax + Bu,\ A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix}, B = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \\
x = \begin{pmatrix} \Delta s \\ \Delta v \\ a_{\text{ego}} \end{pmatrix}, u = j_{\text{ego}}, \\
\Delta s = \delta - d_{\text{safe}},\ \Delta v = v_{\text{lead}} - v_{\text{ego}},
\end{gathered} \tag{2.1}$$

where the control variable u is the jerk of the ego vehicle j_{ego} (jerk is the time derivative of acceleration). The state and control inputs are only allowed to take values within the following intervals:

$$\begin{gathered}
0 \leq \Delta s \leq \Delta s_{\max}, \\
a_{\min} \leq a_{\text{ego}} \leq a_{\max}, \\
j_{\min} \leq j_{\text{ego}} \leq j_{\max},
\end{gathered} \tag{2.2}$$

where $\Delta s_{\max}$, $a_{\min}$, $a_{\max}$, $j_{\min}$, and $j_{\max}$ are user-specified parameters. An additional constraint is considered for the acceleration of the ego vehicles, in order to achieve string stability [92]:

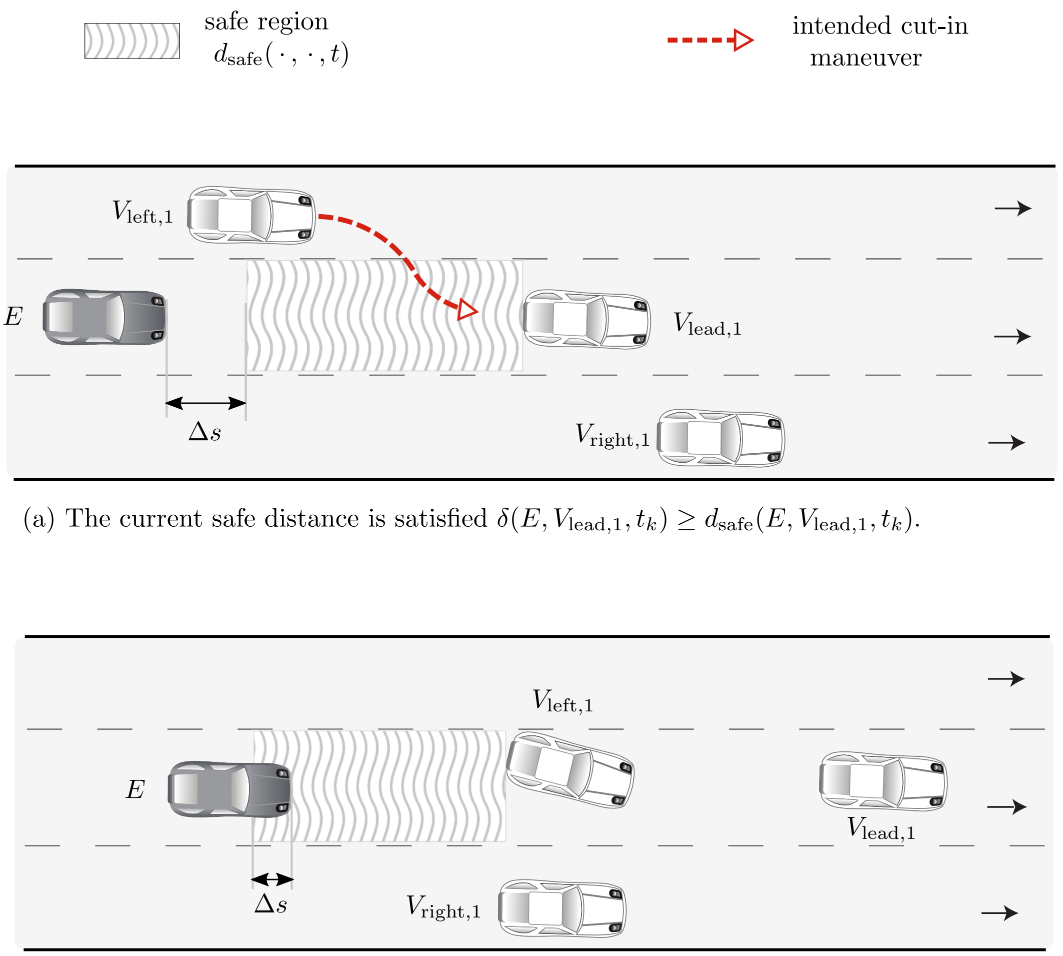
$$a_{\text{ego,k}} \leq \max_{\tau \in [k-H,k]} |a_{\text{lead},\tau}|,\ k \in \{1, \ldots, N_{\text{pred}}\}, \tag{2.3}$$

where N_{pred} is the predicted horizon, H is the size of the time window, and t_k is the current timestamp. In other words, in order to achieve string stability, the acceleration of the ego vehicle should be bounded by the acceleration profile of the leading vehicle over the past time interval $[t_k - H, t_k]$.

2.2.3 Problem Formulation

When focusing on the longitudinal motion of the ego vehicle, three main categories of traffic scenarios can be distinguished depending on the behavior of the other traffic participants (since we assume that the ego vehicle is following the current lane). First, if there is no leading vehicle, typical cruise control (CC) can be applied such that a predefined velocity is maintained by the ego vehicle. When other traffic participants are driving in the proximity of the ego vehicle, two other possible traffic situations may occur where the ego vehicle has to adapt its behavior such that a potential rear-end collision can be avoided.

In the first of these two situations involving other traffic participants, one of the leading vehicles initiates an emergency brake and a collision may be inevitable if the inter-vehicle distance $\delta(E, V_{\text{lead},1}, t)$ is not large enough. Therefore, a correct safe inter-vehicle distance must be computed such that a collision



(a) The current safe distance is satisfied $\delta(E, V_{\text{lead},1}, t_k) \geq d_{\text{safe}}(E, V_{\text{lead},1}, t_k)$.

(b) The safe distance is violated by the cut-in vehicle $\delta(E, V_{\text{left},1}, t_{k+1}) < d_{\text{safe}}(E, V_{\text{left},1} t_{k+1})$.

Figure 2.3: A traffic participant $V_{\text{left},1}$ starting a lane change maneuver towards the inter-vehicle gap between the ego and the leading vehicle.

can be avoided only through braking. The primary issue we deal with here is the design of a control scheme to guarantee that the safe inter-vehicle distance is not violated even in the worst-case scenario where the leading vehicle starts an emergency maneuver. In addition, the generated maneuvers should be comfortable but should not jeopardize safety.

The next considered situation involving other traffic participants represents one where the surrounding vehicles driving in adjacent lanes initiates a lane change towards the current lane and violate the safe distance, as illustrated in Fig. 2.3. Thus, the ego vehicle must cope with the unexpected maneuvers of other traffic participants by restoring the violated safe distance as soon as possible.

To summarize, a unified framework must be designed to control the longitudinal movement of the ego vehicle which guarantees that the ego vehicle will not produce a rear-end collision with the leading vehicle and, in the case of an unexpected cut-in maneuver performed by a vehicle driving in an adjacent

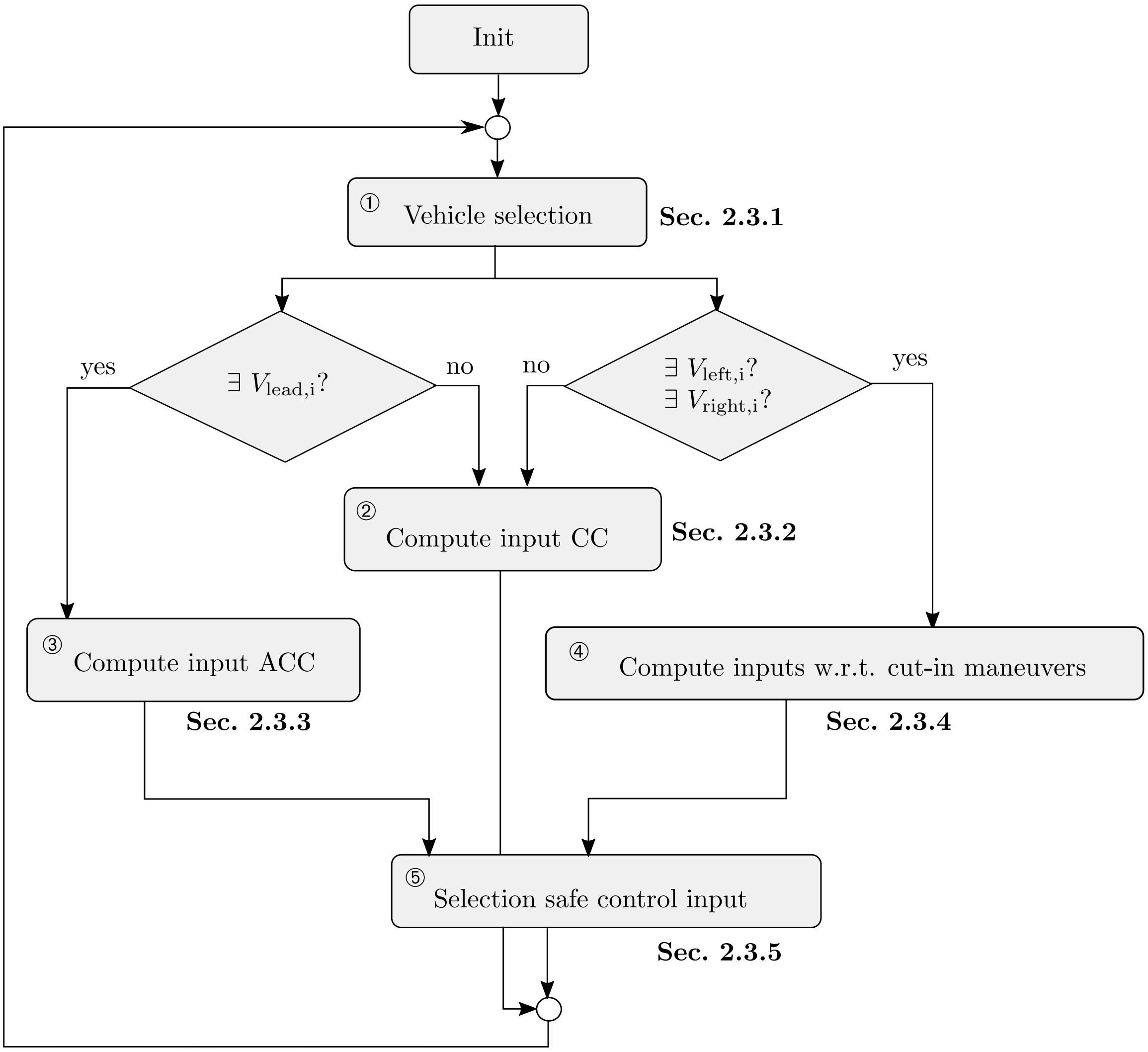

Init
① Vehicle selection
Sec. 2.3.1
yes
$\exists\, V_{\text{lead,i}}$?
no
no
$\exists\, V_{\text{left,i}}$?
$\exists\, V_{\text{right,i}}$?
yes
② Compute input CC
Sec. 2.3.2
③ Compute input ACC
Sec. 2.3.3
④ Compute inputs w.r.t. cut-in maneuvers
Sec. 2.3.4
⑤ Selection safe control input
Sec. 2.3.5

Figure 2.4: Architecture for generating the safe longitudinal control input algorithm.

lane, the ego vehicle shall reestablish the safe distance in a given amount of time.

2.3 Safe Longitudinal Control

In this section, our approach for generating longitudinal control with safety guarantees for the ego vehicle is presented, as illustrated in Fig. 2.4. To ensure that the inter-vehicle distance between the ego and leading vehicle is larger than the desired safe distance, an ACC system with safety guarantees was proposed in our previous work [121]. First, a required safe braking distance is computed for the ego vehicle. Then, an MPC controller tries to generate an acceleration command for the ego vehicle such that the safe distance is satisfied by assuming the most likely behavior of the leading vehicle. Here, a simple prediction that assumes constant velocity is applied. However, more sophisticated prediction algorithms could be used as well. The generated control input is then applied to the ego vehicle only if it is verified as safe, i.e., by applying this control input, the desired safe distance is not violated even in the worst-case scenario of the leading vehicle fully braking.

The general architecture is determined by five main steps: ① First, we select the vehicles that could influence the behavior of the ego vehicle by considering the braking distance of the ego vehicle and the predicted behavior of the other traffic participants (Sec. 2.3.1). ② If there are no selected vehicles, then a typical cruise control is applied to the ego vehicle (Sec. 2.3.2). ③ Otherwise, if there are vehicles driving in the ego lane, an ACC input with respect to each of those vehicles is generated (Sec. 2.3.3). ④ If there are surrounding vehicles driving in adjacent lanes and initiating lane change maneuvers towards the ego lane, a corresponding control input is generated with respect to each cutting-in vehicle, such that the corresponding safe distance is restored as soon as possible (Sec. 2.3.4). ⑤ Lastly, only one control input is selected to be applied to the ego vehicle, such that safety is satisfied with respect to all the considered vehicles (Sec. 2.3.5). Each step will be described in more detail in the following sections.

2.3.1 Vehicle Selection

In this section, a method to select the relevant vehicles that could influence the behavior of the ego vehicle is presented. Since it is assumed for now that the ego vehicle may only move autonomously in the longitudinal direction along the current lane and that no lane change is performed by the ego vehicle[1], the traffic situation could change primarily due to the following types of actions of the surrounding vehicles:

i) Let us first consider the vehicles driving in the same lane as the ego vehicle. If the closest leading vehicle engages an emergency brake, the ego vehicle should be able to avoid a collision. Moreover, if the closest leading vehicle performs a lane change because the next leading object has a lower velocity or is at a standstill (e.g., a stopped vehicle or a construction site), the ego vehicle should be able to avoid a crash with the next leading object as well. This is valid for all objects that are driving or are situated within the stopping distance of the ego vehicle.

ii) Secondly, the other traffic participants driving in adjacent lanes are considered. However, the ones that maintain their current lane do not directly affect the future behavior of the ego vehicle. Therefore, only the vehicles, whose intended maneuver is to perform a lane change towards the ego lane are considered here.

Let us first introduce the stopping distance of the ego vehicle. Given the current velocity v_{ego}, assumption of the maximum deceleration a_{min} of the ego vehicle, and considering the reaction time t_{react} corresponding to the latency of the actuators, the stopping distance of the ego vehicle $d_{\mathrm{stop,ego}}$ can be

[1] To perform a lane change, a vehicle is required to maintain the safe distance while approaching an inter-vehicle gap from an adjacent lane.

defined as follows [158]:

$$d_{\mathrm{stop,ego}} = t_{\mathrm{react}} \cdot v_{\mathrm{ego}} + \frac{v_{\mathrm{ego}}^2}{2 \cdot |a_{\mathrm{min}}|}. \tag{2.4}$$

Next, we define d_{select} as the distance measured from the current position of the ego vehicle within which the relevant vehicles are selected. We set $d_{\mathrm{select}} = d_{\mathrm{stop,ego}}$ since, in this way, we are already considering the worst-case scenario where all vehicles within the stopping distance are taken into account. This means that there will be no collision with any vehicles driving outside this area since a standstill would have been already reached by the ego vehicle, outside the selected distance.

Then, we define the selected relevant surrounding vehicles V_{selected} as follows (see Fig. 2.5):

$$\begin{aligned} V_{\mathrm{selected}} = \Big\{ V_{\square,\mathrm{i}} \mid \ [\delta(E, V_{\square,\mathrm{i}}, t) \leq d_{\mathrm{select}} \ ] \ \wedge \ \ [\ (\square = \text{lead}\ ) \ \vee \ \ (\square \in \{\text{left, right}\} \wedge \\ V_{\square,\mathrm{i}} \text{ performs a lane change towards the ego lane }) \ ] \Big\}. \end{aligned} \tag{2.5}$$

Note that we are not considering any future intended maneuvers of the leading vehicles (i.e., change to the left/right adjacent lane) until they actually reach the target adjacent lanes since we must guarantee that no rear collision with any of the leading vehicles will occur. It is, therefore, better to consider that the leading vehicle(s) will remain within the current lane rather than erroneously predicting a lane change and a rear-end collision to take place.

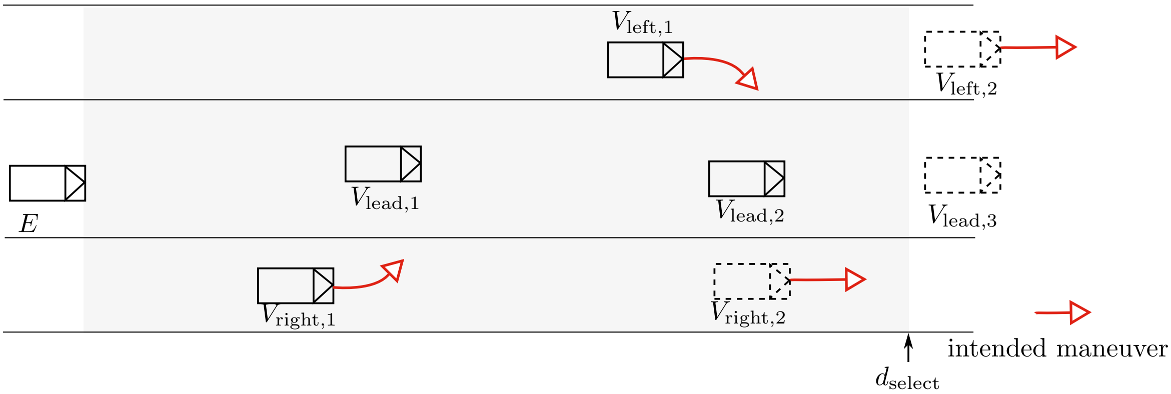


Figure 2.5: Vehicle selection based on the stopping distance of the ego vehicle and on the intended maneuver of the vehicles driving in adjacent lanes. The selected vehicles are represented with a solid line whereas the discarded vehicles are represented with a dashed line.

2.3.2 Cruise Control

After the selection of the surrounding vehicles that could influence the behavior of the ego vehicle, if no surrounding vehicles are selected, then typical cruise control (CC) is applied to the ego vehicle. Although cruise control systems have been already extensively studied both in academia and industry, for completeness a brief introduction to CC systems is subsequently presented. Since no surrounding vehicles are involved here, the dynamics of the ego vehicle can be modeled by a double integrator:

$$\begin{aligned}\dot{x}_{\mathrm{CC}} &= \begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix} x_{\mathrm{CC}} + \begin{pmatrix}0\\ 1\end{pmatrix} j_{\mathrm{ego}},\\ x_{\mathrm{CC}} &= \begin{pmatrix}\Delta v_{\mathrm{ego}}\\ a_{\mathrm{ego}}\end{pmatrix} \text{ and } \Delta v_{\mathrm{ego}} = |v_{\mathrm{ego}} - v_{\mathrm{des}}|,\end{aligned} \tag{2.6}$$

where $|\cdot|$ represents the absolute value and v_{des} is the desired speed. To increase the driving comfort by gradually reaching the desired velocity, the following cost function is introduced:

$$J_{\mathrm{CC}} = \sum_{i=1}^{N_{\mathrm{cc}}} \gamma_{v,i} \cdot \Delta v_{\mathrm{ego}}^2(t_i) + \gamma_u \cdot j_{\mathrm{ego}}^2(t_i), \tag{2.7}$$

where N_{cc} is the time horizon for calculation of the control input for cruise control and $\gamma_{v,i}$, γ_u are the weighting parameters corresponding to Δv_{ego} and j_{ego}, respectively. Finally, the control problem is formulated as follows:

$$\begin{aligned}\min_{a_{\mathrm{ego}}(t_i)} \quad & J_{\mathrm{CC}}\\ \text{subject to} \quad & \forall i \in \{1, \dots, N_{cc}\}:\\ & j_{\min} \leq j_{\mathrm{ego}}(t_i) \leq j_{\max},\\ & \Delta v_{\mathrm{ego}} \leq \Delta v_{\max},\\ & \text{eq. (2.7)}.\end{aligned} \tag{2.8}$$

2.3.3 Adaptive Cruise Control

If among the selected vehicles, there are vehicles driving in the current lane, then the ego vehicle should apply an ACC control input such that the corresponding safe inter-vehicle distance is kept with respect to the leading vehicle.

However, maintaining a safe inter-vehicle distance only with respect to the closest leading vehicle $V_{\mathrm{lead},1}$ (as illustrated in Fig. 2.3) is not sufficient to ensure safety. If the closest leading vehicle $V_{\mathrm{lead},1}$ is changing its lane due to another vehicle that is driving slower, or due to a static obstacle, the ego vehicle may not be able to find any acceleration input such that a safe distance is kept with respect to the next leading vehicle $V_{\mathrm{lead},2}$, or no emergency maneuver could exist such that a collision is avoided.

In contrast to a typical ACC, where the longitudinal control input is calculated only with respect to the closest leading vehicle, we generate a longitudinal control input with respect to each vehicle driving ahead of the ego vehicle in the same lane such that safety is guaranteed. Choosing which control input to be applied among the generated ones is later explained in Sec. 2.3.5. In this section, for the purpose of notation simplicity, we exemplify the ACC design for only one leading vehicle V_{lead}. Nevertheless, generation of the ACC control input for each of the leading vehicles is done in the same manner.

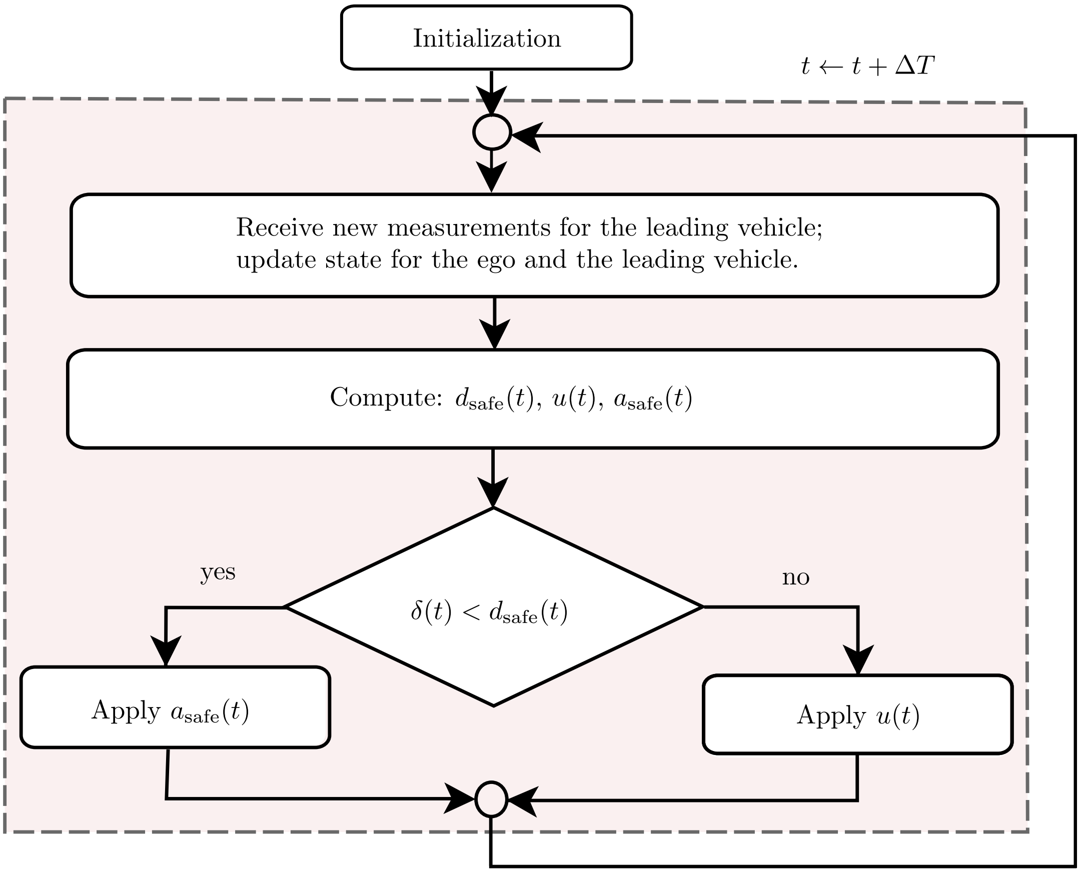

Initialization
$t \leftarrow t + \Delta T$
Receive new measurements for the leading vehicle; update state for the ego and the leading vehicle.
Compute: $d_{\text{safe}}(t)$, $u(t)$, $a_{\text{safe}}(t)$
yes
no
$\delta(t) < d_{\text{safe}}(t)$
Apply $a_{\text{safe}}(t)$
Apply $u(t)$

Figure 2.6: Control scheme of our proposed ACC concept.

To guarantee collision avoidance, we embed standard controllers for adaptive cruise control into a framework that ensures that safe distance calculated with respect to the leading vehicle is satisfied at all times. The main idea behind this is to always have a safe braking trajectory available that is able to bring the ego vehicle to a safe stop, even when the leading vehicle suddenly fully brakes. As long as the standard controller, which we refer to as the *nominal controller*, is able to keep an inter-vehicle distance $\delta(E, V_{\text{lead}}, t) \geq d_{\text{safe}}(E, V_{\text{lead}}, t)$, the nominal controller stays in action. If, however, there exists no feasible input $u(t)$ such that $\delta(E, V_{\text{lead}}, t) \geq d_{\text{safe}}(E, V_{\text{lead}}, t)$, a precomputed emergency maneuver $a_{\text{safe}}(t)$ is engaged so that safety is not jeopardized. This can be guaranteed since the precomputed emergency control $a_{\text{safe}}(t)$ is calculated such that a collision can be avoided, even in the worst-case scenario when the leading vehicle fully brakes. The main control scheme is illustrated in Fig. 2.6.

In the following section, Sec. 2.3.3.1, several braking profiles with respect to the length of d_{safe} and the jerk[1] values of the braking trajectory are studied. A more gradual engagement of brakes decreases jerk and thus increases comfort, while enlarging the required safe distance d_{safe}. Controllers for tracking

[1] Jerk represents the time derivative of acceleration; it is a typical measure for driving comfort.

the pre-computed braking trajectory are not discussed as the focus of this work in on the novel aspect of guaranteeing collision avoidance. During the emergency maneuver, if the inter-vehicle distance becomes again safe, i.e., $\delta(E, V_{\text{lead}}, t) > d_{\text{safe}}(E, V_{\text{lead}}, t)$ since d_{safe} has shortened due to the fact that the preceding vehicle has not engaged brakes to the expected extent, the control is taken back by the nominal controller. Since (i) we choose braking profiles such that they initially engage mildly and (ii), in almost all cases, control quickly goes back to the nominal controller, passengers would not realize that an emergency trajectory was engaged.

In this work, we use model predictive control (MPC) as the nominal controller, since MPC provides optimal solutions while attempting to meet constraints—this, however, is not always achieved due to assumptions on the behavior of the leading vehicle, that are not always correct. Our MPC is computed based on the assumption that the leading vehicle moves with constant velocity, which is a reasonable assumption to optimize ride comfort, but safety cannot be ensured since the leading vehicle may suddenly brake. Therefore, an emergency controller must be applied when a critical situation occurs. The control output of the MPC is denoted by $u(t)$. Our cost function for the MPC is rather standard and can be formulated as a quadratic programming (QP) problem where all matrices have appropriately chosen dimensions:

$$\min_u J(x(t), u(t)) = x_{N|t}^T P x_{N|t} + \\ + \sum_{i=0}^{N-1} \left(x_{i|t}^T Q x_{i|t} + u_{i|t}^T R u_{i|t} \right), \text{subject to: (2.2)-(2.3)},$$

where:

- $x_{i|t}$ and $u_{i|t}$ are the state and input at time i, based on the state measurement at time t, respectively,
- $J(\cdot,\cdot)$ is the cost function, N is the prediction horizon,
- matrix $Q \geq 0$ weights the state vector,
- matrix $R > 0$ penalizes the control input,
- terminal cost P is chosen to guarantee stability.

To summarize the proposed control scheme, first an optimal control output $u(t)$ is generated under the assumption that the leading vehicle is driving with constant velocity v_{lead}. At each sample time we verify if the safety distance d_{safe} is satisfied after applying $u(t_k)$ for one time step. This means we check if there exists an emergency maneuver a_{safe} that can bring the ego vehicle to standstill while avoiding any collision, even in the worst-case scenario when the leading vehicle brakes with full deceleration $a_{\min}$ (see Fig. 2.7b). If the verified control output $u(t_k)$ yields a safe distance, then $u(t_k)$ is applied to the system.

Nevertheless, if no feasible control output $u(t_k)$ is found such that the safe distance is satisfied $\delta(E, V, t_{k+1}) \geq d_{\text{safe}}(E, V, t_{k+1})$ (as illustrated in Fig. 2.7c), a gradual emergency maneuver $a_{\text{safe}}(t_k)$ is applied, which guarantees safety, for any deceleration profile of the leading vehicle. Until a new safe control output $u(t)$ is found, the pre-computed emergency maneuver is applied.

2.3.3.1 Safe distance and acceleration profile of the emergency maneuver.

In [158], a formal analysis for safe distance computation is presented, however, only the case where constant acceleration is applied is considered. Here, four different deceleration profiles are analyzed. We select the solution that guarantees safety at all times and additionally assures comfort by minimizing jerk. Based on these criteria, we propose a mixed deceleration profile $a_{\text{safe}}(t)$ and we compute the safe braking distance d_{ego} such that a collision can be avoided. Moreover, the jerk values are kept within given limits such that uncomfortable driving is avoided, if the dangerous situation resolves.

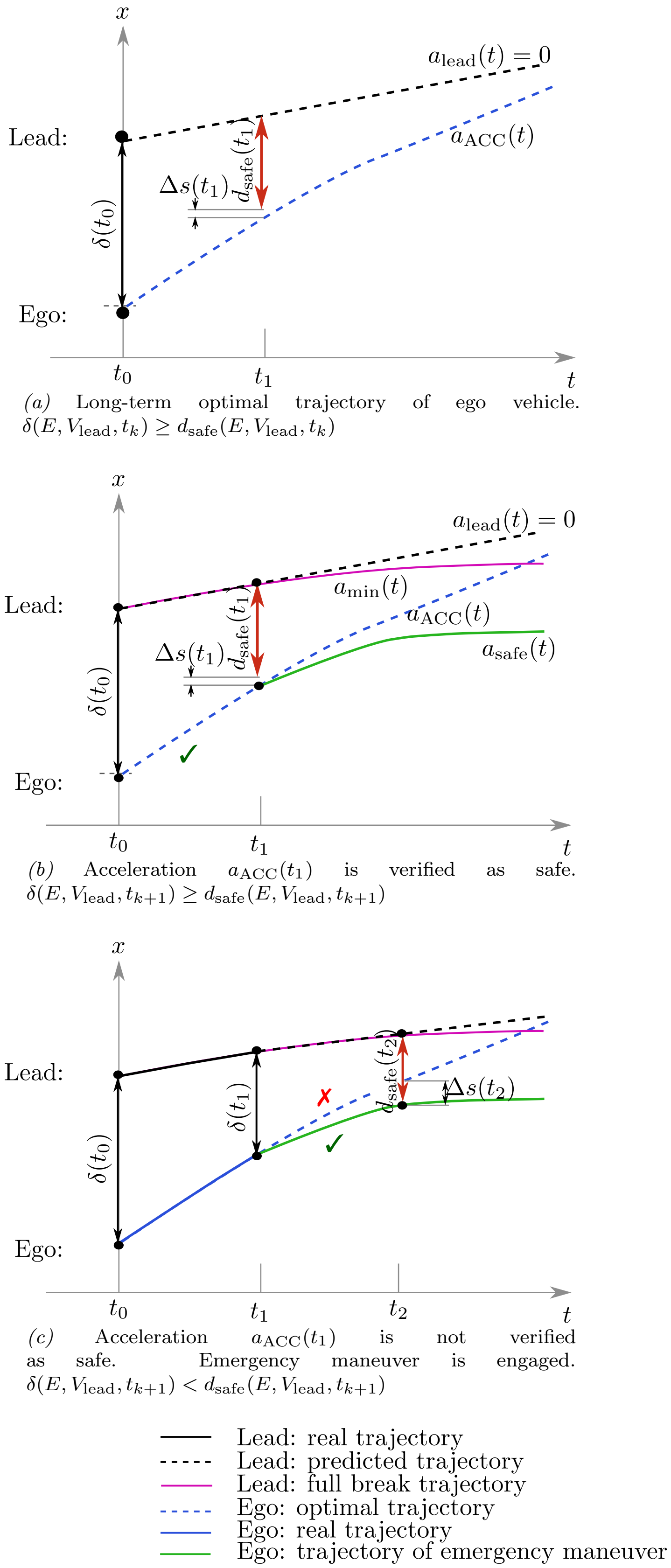
Constant full deceleration. One of the most straightforward approaches is to apply constant full deceleration: $a_{\text{safe}}(t) = a_{min},\ t \geq 0$. This profile provides the smallest safe distance possible. However, applying full deceleration leads to uncomfortable driving. Moreover, due to the jerky behavior, traffic flow might be hindered.

Linear deceleration. Another possible profile is linear deceleration: $a_{\text{safe}}(t) = \dfrac{a_{\min} t}{c},\ t \geq 0,\ c > 0,$ $a_{\min} \leq a_{\text{safe}}(t) < 0$. The jerk values introduced by this profile are lower since the acceleration is linearly decreasing. However, the braking distance corresponding to this profile is larger than the one corresponding to the constant full deceleration profile described previously.

Exponential deceleration. The exponential deceleration profile is defined as follows:

$$a_{\text{safe}}(t) = 1 - c^t,\ t > 0,\ c > 1,\ a_{\min} \leq a_{\text{safe}}(t) < 0. \tag{2.9}$$

When applying the exponential deceleration, the jerk value is even less compared to the one produced by the linear deceleration. Therefore, if the leading vehicle fully brakes for only one time step, the ego vehicle will smoothly brake, making this deceleration profile suitable for systems whose measurements are affected by noise. In the following, the computation of the braking distance is derived: Given $a = \dot{v}$ and $v = \dot{s}$, where v and s are velocity and position; s is the solution of the differential equation $\ddot{s} = a_{\text{safe}}(t)$. Let us first define the braking time of the ego vehicle $t_{\text{stop,H}}$ as the time when the velocity reaches 0, where the initial velocity is $\dot{s}(0) = v_0$, and t_{H} represents the time when the maximum deceleration is reached. The braking distance d_{ego} is the exact solution of $\ddot{s} = a_{\text{safe}}(t)$, computed for the braking time



(a) Long-term optimal trajectory of ego vehicle. $\delta(E, V_{\text{lead}}, t_k) \geq d_{\text{safe}}(E, V_{\text{lead}}, t_k)$

(b) Acceleration $a_{\text{ACC}}(t_1)$ is verified as safe. $\delta(E, V_{\text{lead}}, t_{k+1}) \geq d_{\text{safe}}(E, V_{\text{lead}}, t_{k+1})$

(c) Acceleration $a_{\text{ACC}}(t_1)$ is not verified as safe. Emergency maneuver is engaged. $\delta(E, V_{\text{lead}}, t_{k+1}) < d_{\text{safe}}(E, V_{\text{lead}}, t_{k+1})$

Figure 2.7: Adaptive cruise control. Switching between nominal control and emergency maneuver.

$t_{\text{stop,H}}$, by double integrating the acceleration:

$$d_{\text{ego}} = \frac{1}{\ln^2 c} + \frac{t_H^2}{2} - \frac{c^{t_H}}{\ln^2 c} + \frac{t_{\text{ stop,H}}}{\ln c} + v_0 t_{\text{stop,H}}. \tag{2.10}$$

Mixed deceleration. The previous deceleration profiles introduce several benefits (e.g., providing the shortest safe distance) but also a series of drawbacks: (i) if constant maximum deceleration is applied, the jerk is a Dirac function, so the value goes to infinity, and (ii) by applying linear or exponential deceleration, while the value of jerk is comfortable, the braking distance is too long. This could result in frequent cut-in maneuvers performed by vehicles driving in adjacent lanes. To overcome these disadvantages, a mixed deceleration profile is proposed:

$$\begin{gathered} a_{\text{safe}}(t) = \begin{cases} 1 - c^t & \text{if } t \leq t_{\text{H}}, \\ a_{\text{min}} & \text{if } t_{\text{H}} < t \leq t_{\text{stop,H}}, \end{cases} \\ t_{\text{H}} = \log_c (1 - a_{\text{min}}), \\ t > 0,\ c > 1,\ a_{\text{min}} \leq a_{\text{safe}}(t) < 0. \end{gathered} \tag{2.11}$$

The parameter t_{H} represents the time when the maximum deceleration is reached during exponential deceleration so that we continue with full braking. Hereafter, the mixed deceleration profile is utilized to generate the acceleration profile that is employed during the emergency maneuver.

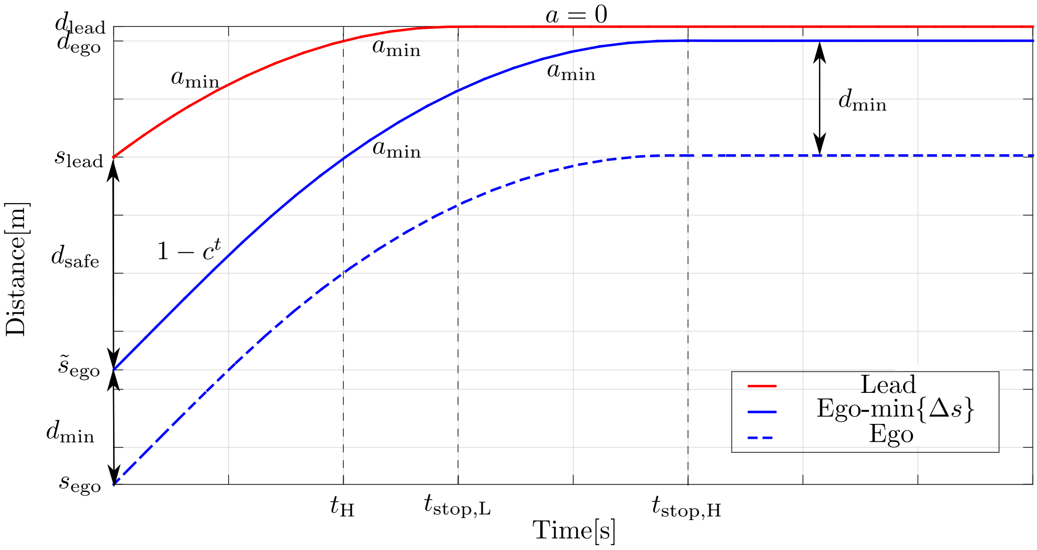


Figure 2.8: Safe distance computation.

Let $s_{\text{L}}(t) = s_{\text{lead}} + v_{\text{lead}}t - \frac{1}{2}|a_{\text{min}}|t^2$ be the function that describes the leading vehicle position and $v_L(t) = v_{\text{lead}} - |a_{\text{min}}|t$ be the leading vehicle velocity, considering the worst-case scenario where full brake is applied. Let $s_{\text{H}}(t)$ be the function that describes the ego vehicle's position. Then, the required inter-vehicle distance d_{safe} that must be maintained in order to guarantee safety is computed as:

$$d_{\mathrm{safe}} = s_{\mathrm{lead}} - s_{\mathrm{E}} - d_{\mathrm{min}},$$

where the distance between $s_{\mathrm{L}}(t)$ and $s_{\mathrm{H}}(t)$ over a time interval $\Delta\tau$ is:

$$d_{\mathrm{min}} = \min_{t\in\Delta\tau} \left(s_{\mathrm{L}}(t) - s_{\mathrm{H}}(t)\right).$$

To compute d_{safe}, we analyze the monotonicity of $s_{\mathrm{L}}(t) - s_{\mathrm{H}}(t)$. Both $s_{\mathrm{L}}(t)$ and $s_{\mathrm{H}}(t)$ are monotonically increasing over time intervals $\Delta\tau = [t_{\mathrm{min}}, t_{\mathrm{max}}]$, where $t_{\mathrm{min}}, t_{\mathrm{max}} \in \{t_{\mathrm{stop,L}},\ t_{\mathrm{H}},\ t_{\mathrm{stop,H}}\}$ and

$$t_{\mathrm{stop,L}} = \frac{v_{\mathrm{lead}}}{|a_{\mathrm{min}}|},$$

$$t_{\mathrm{H}} = \frac{\ln(1-a_{\mathrm{min}})}{\ln(c)},$$

$$t_{\mathrm{stop,H}} = \frac{1}{|a_{\mathrm{min}}|}\left(t_{\mathrm{H}} + \frac{1-c^{t_{\mathrm{H}}}}{\ln(c)} + v_{\mathrm{ego}}\right),$$

where $t_{\mathrm{stop,L}}$ is the braking time of the leading vehicle, and $|\cdot|$ represents the absolute value of a real number.

Next, to analyze the monotonicity of the inter-vehicle distance, we first compute all possible permutations between t_{H}, $t_{\mathrm{stop,H}}$, and $t_{\mathrm{stop,L}}$, since the acceleration mode changes at this points in time. In the following, it is considered that zero acceleration determines a standstill state. There are six possible situations, based on the applied accelerations ($a_{\mathrm{lead}} \in \{0, a_{\mathrm{min}}\}$ and $a_{\mathrm{ego}} \in \{0, a_{\mathrm{min}}, 1-c^t\}$), described in Table 2.1.

Table 2.1: Changes in the monotonicity of the inter-vehicle distance based on the time when the acceleration mode changes.

Situation #	**Condition**
# 1	$t_{\mathrm{H}} \le t_{\mathrm{stop,L}} \le t_{\mathrm{stop,H}}$
# 2	$t_{\mathrm{stop,L}} \le t_{\mathrm{stop,H}} \le t_{\mathrm{H}}$
# 3	$t_{\mathrm{stop,L}} \le t_{\mathrm{H}} \le t_{\mathrm{stop,H}}$
# 4	$t_{\mathrm{stop,H}} \le t_{\mathrm{H}} \le t_{\mathrm{stop,L}}$
# 5	$t_{\mathrm{stop,H}} \le t_{\mathrm{stop,L}} \le t_{\mathrm{H}}$
# 6	$t_{\mathrm{H}} \le t_{\mathrm{stop,H}} \le t_{\mathrm{stop,L}}.$

In the following, we compute $d_{\mathrm{min},\Delta\tau}$ considering all possible combinations of the applied deceleration of the ego and leading vehicle for each time interval. For example, in Fig. 2.8 three different cases can be distinguished: $t \in [0, t_{\mathrm{H}}]:\ a_{\mathrm{lead}} = a_{\mathrm{min}},\ a_{\mathrm{ego}} = 1 - c^t$; $t \in [t_{\mathrm{H}}, t_{\mathrm{stop,L}}]:\ a_{\mathrm{lead}} = a_{\mathrm{min}},\ a_{\mathrm{ego}} = a_{\mathrm{min}}$; $t \in [t_{\mathrm{stop,L}}, t_{\mathrm{stop,H}}]:\ a_{\mathrm{lead}} = 0,\ a_{\mathrm{ego}} = a_{\mathrm{min}}$. The other cases are similar depending on the situation. The cases when both vehicles are at standingstill or when the ego vehicle is at standingstill and the leading vehicle is braking are not considered because they represent obvious safe situations. Therefore, only the remaining four combinations of acceleration are analyzed, as shown in Table 2.2.

Table 2.2: Possible combinations of applied deceleration.

	case (a)	**case (b)**	**case (c)**	**case (d)**
Lead	$\mathrm{a}_{\min}$	standstill	standstill	$\mathrm{a}_{\min}$
Ego	$\mathrm{a}_{\min}$	$\mathrm{a}_{\min}$	$\mathrm{1\text{-}c^t}$	$\mathrm{1\text{-}c^t}$

Subsequently, each case is analyzed and $d_{\min}$ is computed as the minimum over all $d_{\min,\Delta\tau}$ where $\Delta\tau = [t_{\min}, t_{\max}]$, $t_{\min}, t_{\max} \in \{t_{\mathrm{H}}, t_{\mathrm{stop,H}}, t_{\mathrm{stop,L}}\}$, depends on the scenario.

- **Case (a)**: $\Delta v = v_{\mathrm{lead}}(t) - v_{\mathrm{Ego}}(t)$, $t \in \Delta\tau$;

if $\Delta v > 0$ then $\Delta s(t)$ is increasing on $\Delta\tau \Rightarrow d_{\min,\Delta\tau} = s_{\mathrm{L}}(t_{\min}) - s_{\mathrm{H}}(t_{\min})$;

if $\Delta v < 0$ then $\Delta s(t)$ is decreasing on $\Delta\tau \Rightarrow d_{\min,\Delta\tau} = s_{\mathrm{L}}(t_{\max}) - s_{\mathrm{H}}(t_{\max})$.

- **Case (b)**: $\Delta v = 0 - v_H(t) < 0$ then $\Delta s(t)$ is decreasing on $\Delta\tau \Rightarrow d_{\min,\Delta\tau} = s_{\mathrm{L}}(t_{\max}) - s_{\mathrm{H}}(t_{\max})$.
- **Case (c)**: $\Delta v = 0 - v_H(t) < 0$ then $\Delta s(t)$ is decreasing on $\Delta\tau \Rightarrow d_{\min,\Delta\tau} = s_{\mathrm{L}}(t_{\max}) - s_{\mathrm{H}}(t_{\max})$.
- **Case (d)**: The variation of the velocity is described by $\Delta v = a_{\min}t + v_{\mathrm{lead}} - t - v_{\mathrm{ego}} - \dfrac{1-c^t}{\ln(c)}$; to find if Δs is increasing or decreasing, $\Delta v = 0$ is computed. Therefore, the critical point of Δs is:

$$t_0 = -\frac{q\ \ln(c) + p\ \mathit{LambertW}\left(0, \dfrac{\ln(c)}{p\sqrt[p]{c^q}}\right)}{p\ \ln(c)},$$

where $p = (a_{\min} - 1)\ln(c)$, $q = 1 + (v_{\mathrm{lead}} - v_{\mathrm{ego}})\ln(c)$, and the $\mathit{LambertW}$ function is the inverse function of $f(W) = We^W$. To check if $\Delta s(t)$ has a minimum or a maximum value at time t_0, we compute the second derivative: $\Delta\ddot{s}(t) = \Delta a(t) = a_{\min} - (1 - c^t)$. Since $\Delta a(t) < 0 \Rightarrow \Delta s(t_0)$ has a maximum at t_0. Therefore, the minimum of $\Delta s(t)$ can be at either $t_{\min}$ or $t_{\max}$. Three further cases can be distinguished:

(d.1): $t_0 < t_{\mathbf{min}} \Rightarrow \Delta s(t)$ is decreasing on $\Delta\tau \Rightarrow d_{\min,\Delta\tau} = s_{\mathrm{L}}(t_{\max}) - s_{\mathrm{H}}(t_{\max})$;

(d.2): $t_0 > t_{\mathbf{max}} \Rightarrow \Delta s(t)$ is increasing on $\Delta\tau \Rightarrow d_{\min,\Delta\tau} = s_{\mathrm{L}}(t_{\min}) - s_{\mathrm{H}}(t_{\min})$;

(d.3): $t_0 \in [t_{\mathbf{min}}, t_{\mathbf{max}}] \Rightarrow d_{\min,\Delta T} = \begin{cases} \Delta s(t_{\min}), & \text{if } \Delta s(t_{\min}) < \Delta s(t_{\max}) \\ \Delta s(t_{\max}), & \text{if } \Delta s(t_{\min}) \geq \Delta s(t_{\max}) \end{cases}.$

To summarize, first the time intervals $[t_{\min}, t_{\max}]$, $t_{\min}, t_{\max} \in \{t_{\mathrm{H}}, t_{\mathrm{stop,H}}, t_{\mathrm{stop,L}}\}$ are selected depending on the scenario, as shown in Table 2.1. Then, $d_{\min}$ is computed accordingly, based on the applied deceleration profiles. Finally, the safe distance d_{safe} is computed by applying the proposed deceleration profile $a_{\mathrm{safe}}(t)$ such that any collision can be avoided.

2.3.4 Reaction to Cut-in Maneuvers

Apart from the vehicles driving in the ego's lane, the selected vehicles driving in the adjacent lanes can influence the behavior of the ego vehicle. Recall that among the surrounding vehicles driving within the stopping distance of the ego vehicle, only those who are initiating a lane change towards the ego's lane (referred to as cutting-in vehicles) are selected.

To decide how the ego vehicle should react to the cutting-in vehicle, let us first introduce the concept of Inevitable Collision State (ICS) [49]:

Definition 2.4 (Inevitable collision state)**.** A vehicle is in an inevitable collision state if a collision with an obstacle (static or dynamic) is inevitable, no matter the future trajectory of the vehicle. □

A cut-in maneuver performed by a surrounding vehicle can result in three possible traffic scenarios:

1. The ego vehicle is in an inevitable collision state (not only the safe distance is violated, but the collision cannot be avoided, even by applying full brake);
2. The safe distance is violated, but the ego vehicle is not in an inevitable collision state and the safe distance can be restored by engaging a *clearance acceleration*;
3. The safe distance is not violated; therefore, the behavior of the ego vehicle is not affected by the cut-in maneuver.

In order to determine which traffic scenario the ego vehicle will end up in due to the cutting-in maneuver, the future occupancy of the cutting-in vehicle is required. Once a lane change maneuver towards the ego lane is predicted, the trajectory associated with it is predicted as well. However, instead of using a prediction algorithm, we are directly projecting the current occupancy of the cutting-in vehicles $\Gamma(V, t_0)$, $V \in \{V_{\text{left,i}}, V_{\text{right,i}}\}$ onto the ego's lane, by considering that a lane change is performed instantaneously. This approach will enhance safety due to the overapproximative manner of computing the future trajectory of the cutting-in vehicle, since by considering an instantaneous lane change, the most aggressive lane change a vehicle can perform, is already incorporated. Moreover, this approach is also computationally inexpensive.

Definition 2.5 (Projected occupancy of a surrounding vehicle)**.** The projected occupancy $\Gamma_p(V, t)$ of a surrounding vehicle $V \in \{V_{\text{left,i}}, V_{\text{right,i}}\}$ onto the ego vehicle's lane is represented by the oriented bounding box corresponding to the occupancy $\Gamma(V, t)$ rotated in the driving direction and translated to the center line of the ego's lane. □

After the predicted occupancy is generated, if no inevitable collision state is detected, a clearance acceleration is computed for the ego vehicle, such that the violated safe distance is restored. In the following, we present a method for determining whether the ego vehicle is in an inevitable collision state with respect to the projected occupancy $\Gamma_p(V, t)$, by considering that the cutting-in vehicle V can instantaneously reach the ego vehicle's lane, once a lane change is predicted.

2.3.4.1 Inevitable Collision State

To check if a cut-in maneuver could lead to an ICS, two different behaviors are assumed for the cutting-in vehicle once the ego vehicle's lane is reached. **Case (i)**: the cutting-in vehicle fully accelerates; and, **Case (ii)**: the cutting-in vehicle drives with constant velocity. For both cases, it is assumed that the ego vehicle engages full deceleration $a_{\min}$ starting when the cutting-in vehicle reaches the ego vehicle's lane until standstill is reached at time

$$t_{\text{E, stop}} = \frac{v_E(t_0)}{a_{\min}}. \tag{2.12}$$

Case (i): To verify if a collision is inevitable, we first assume that the cutting-in vehicle applies full acceleration $a_{\max}$ once the target lane — on which the ego vehicle drives — is reached. Then, we check whether a collision would occur between the current time t_0 and the stopping time of the ego vehicle, $t_{\text{E, stop}}$. To find the possible collision time, we solve $s_E(t) = s_V(t)$ where

$$s_E(t) = s_E(t_0) + v_E(t_0)t + \frac{a_{\min}t^2}{2}, \tag{2.13}$$

$$s_V(t) = s_V(t_0) + v_V(t_0)t + \frac{a_{\max}t^2}{2}. \tag{2.14}$$

This results in a quadratic equation in t, with two possible solutions:

$$t_1 = \frac{-(v_C(t_0) - v_E(t_0)) + \sqrt{\Delta_1}}{a_{\max} - a_{\min}}, \tag{2.15}$$

$$t_2 = \frac{-(v_V(t_0) - v_E(t_0)) - \sqrt{\Delta_1}}{a_{\max} - a_{\min}}, \tag{2.16}$$

where $\Delta_1 = (v_V(t_0) - v_E(t_0))^2 - 2(a_{\max} - a_{\min})(s_V(t_0) - s_E(t_0))$. Finally, a collision is inevitable if any of the solutions t_1 or t_2 satisfies the following condition:

$$(t_0 \leq t_1 \leq t_{\text{E, stop}}) \vee (t_0 \leq t_2 \leq t_{\text{E, stop}}) \rightarrow \text{ICS}. \tag{2.17}$$

Case (ii): If no ICS is detected in Case (i), we further assume the average case scenario, where the cutting-in vehicle continues driving with constant velocity, once the target lane is reached. To check whether there is an ICS in this case, we compute the intersection time between the position of the two vehicles $s_V(t)$ and $s_E(t)$, where

$$s_V(t) = s_V(t_0) + v_V(t) \cdot t, \tag{2.18}$$

by solving $s_V(t) = s_E(t)$. This quadratic equation has two solutions t_3 and t_4,

$$t_3 = \frac{(v_V(t_0) - v_E(t_0)) + \sqrt{\Delta_2}}{a_{\min}}, \tag{2.19}$$

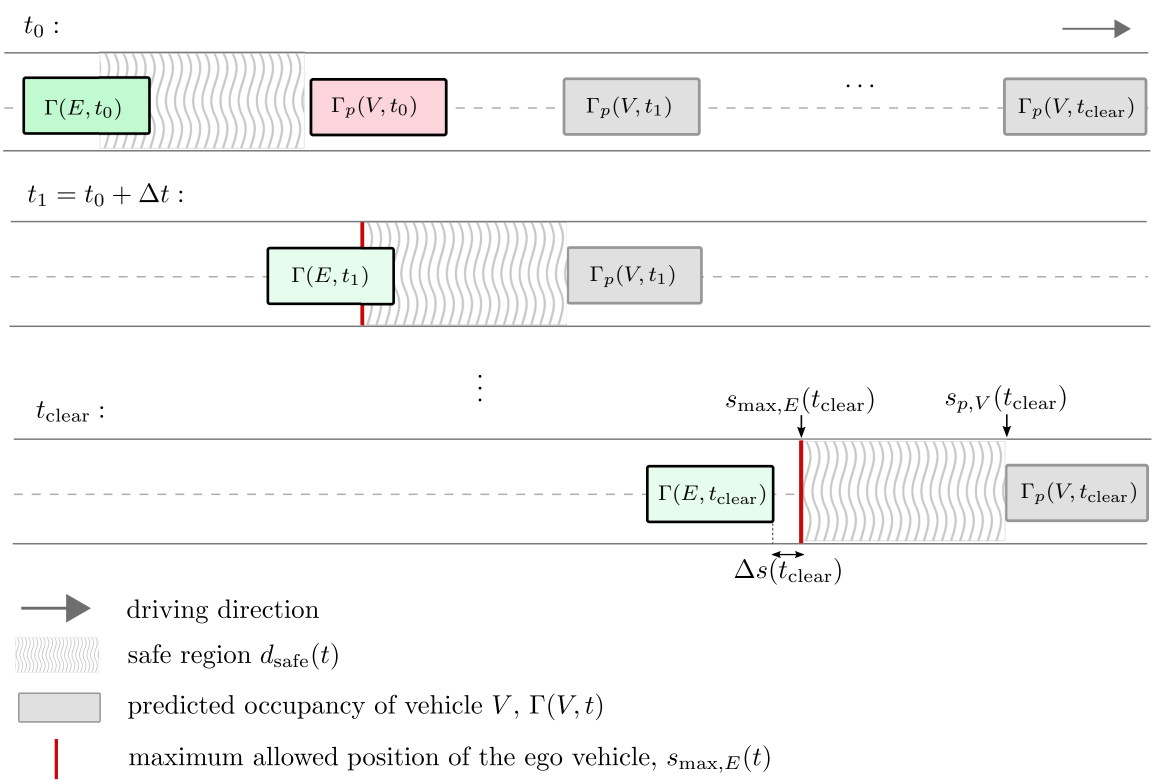

t_0 :
$\Gamma(E, t_0)$
$\Gamma_p(V, t_0)$
$\Gamma_p(V, t_1)$
…
$\Gamma_p(V, t_{\text{clear}})$
$t_1 = t_0 + \Delta t$:
$\Gamma(E, t_1)$
$\Gamma_p(V, t_1)$
⋮
t_{clear} :
$s_{\max,E}(t_{\text{clear}})$
$s_{p,V}(t_{\text{clear}})$
$\Gamma(E, t_{\text{clear}})$
$\Gamma_p(V, t_{\text{clear}})$
$\Delta s(t_{\text{clear}})$
driving direction
safe region $d_{\text{safe}}(t)$
predicted occupancy of vehicle V, $\Gamma(V, t)$
maximum allowed position of the ego vehicle, $s_{\max,E}(t)$

Figure 2.9: After a cut-in maneuver is predicted, the maximum admissible position of the ego vehicle is calculated, such that the inter-vehicle distance between the ego and the cutting-in vehicle is the safe distance. A clearance acceleration should be generated such that the violated safe region is restored within a given time horizon t_{clear}.

$$t_4 = \frac{(v_V(t_0) - v_E(t_0)) - \sqrt{\Delta_2}}{a_{\min}}, \tag{2.20}$$

where $\Delta_2 = (v_E(t_0) - v_V(t_0))^2 - 2a_{\min}(s_E(t_0) - s_V(t_0))$. There exists an ICS if the roots t_3 or t_4 satisfy the following condition:

$$(t_0 \leq t_3 \leq t_{\text{E, stop}}) \vee (t_0 \leq t_4 \leq t_{\text{E, stop}}) \rightarrow \text{ICS}. \tag{2.21}$$

If ICS is detected (via eqs. (2.17) or (2.21)), then the ego vehicle must apply full deceleration $a_{\min}$ in order to mitigate the collision, as shown in Alg. 1. Otherwise, if no ICS is detected under the previous assumptions, a clearance acceleration $a_{\text{clear}}(t)$ is computed for the ego vehicle such that the safe distance is restored within a given time horizon t_{clear}, as subsequently described in Sec. 2.3.4.2.

For the worst-case scenario where the cutting-in vehicle performs a full brake, the ICS can be evaluated by using the approach in [158]. If there is an ICS detected, then the ego vehicle should engage full brake, in order to mitigate the collision. However, the cutting-in vehicle must be held responsible for the possible collision, since it violated the safe distance.

Algorithm 1 Behavior of the ego vehicle based on ICS detection.

```
Input: $\Delta t,\ x_E(t_0),\ x_V(t_0),\ V \in S^p_{\{\text{left, right}\}}$
Output: $a_{\text{clear}}(t), t = t_1, \cdots, t_{\text{clear}}$
Assume $a_{\max}$ for $V$ and $a_{\min}$ for $E$
if ICS == true                                      ▷ according to (2.17)
then
    $a_{\text{clear}}(t) \leftarrow a_{\min}$;      ▷ such that the collision can be mitigated
else
    Assume $a = 0$ for $V$ and $a_{\min}$ for $E$
    if ICS == true                                  ▷ according to (2.21)
    then
        $a_{\text{clear}}(t) \leftarrow a_{\min}$;  ▷ mitigate the collision
    else
        Calculate $a_{\text{clear}}$ as described in Sec 2.3.4.2.
```

2.3.4.2 Clearance acceleration a_{clear} computation

If the safe inter-vehicle distance is violated by a surrounding vehicle V that performs a cut-in maneuver, the ego vehicle should gradually brake, such that the safe distance is restored within a given time horizon t_{clear}.

In order to restore the safe distance within a given time horizon t_{clear}, a clearance acceleration profile $a_{\text{clear}}(t)$ is generated. However, to compute $a_{\text{clear}}(t)$, the future safe distance $d_{\text{safe}}(\Gamma_p(V,t))$, $t = t_1, \cdots, t_{\text{clear}}$ is required. As the future safe distance for the entire time horizon $d_{\text{safe}}(\Gamma_p(V,t))$ cannot be computed since it depends on the unknown $a_{\text{clear}}(t)$, we approximate $d_{\text{safe}}(\Gamma_p(V,t))$ with an upper-bound of the safe distance $\overline{d_{\text{safe}}}(\,\cdot\,)$.

We further assume that the cutting-in vehicle does not brake after the target lane is reached. Moreover, once the safe distance $d_{\text{safe}}(\Gamma_p(V,t))$ is violated, we limit the maximum velocity of the ego vehicle to the current value for the entire time horizon t_{clear}, $v_E(t) \leq v_E(t_0)$, $t = t_1, \cdots, t_{\text{clear}}$, i.e., the ego vehicle does not accelerate until the safe distance is restored.

Definition 2.6 (Upper bound of the safe distance $\overline{d_{\text{safe}}}$)**.** Assuming the minimum acceleration of the cutting-in vehicle V is $a_V(t) = 0$, and requiring the clearance acceleration of the ego vehicle $a_{\text{clear}}(t) \leq 0$, $t = t_1, \cdots, t_{\text{clear}}$, we define the upper bound of the safe distance calculated for the entire clearance time horizon as $\overline{d_{\text{safe}}}(t) = d_{\text{safe}}(t_0)$. □

Recall that the clearance acceleration $a_{\text{clear}}(t) \leq 0$, where $t_0 < t \leq t_{\text{clear}}$, is calculated only if an ICS is not detected when assuming that the cutting-in vehicle does not engage a braking maneuver $a_V(t) \geq 0$ (see Alg.1). If $d_{\text{safe}}(t)$ is greater than $d_{\text{safe}}(t_0)$, $t_0 < t \leq t_{\text{clear}}$, this would result in an ICS since at t_0, the ICS is already verified considering that the ego vehicle applies full emergency brake. On the other hand, if an ICS was detected, instead of calculating the clearance acceleration, the collision would attempt to be mitigated, as illustrated in Alg.1.

Next, a clearance acceleration $a_{\text{clear}}(t)$ is calculated such that the safe distance is restored within a time horizon t_{clear}. The overall approach for computing $a_{\text{clear}}(t)$ is presented in Alg. 2 and subsequently explained. First, the safe distance $d_{\text{safe}}(t_0)$ is computed. Then, for all time instances within the clearance time horizon $\forall t_i \in \{t_1, t_1 + \Delta t, \ldots, t_{\text{clear}}\}$, the required safe distance $d_{\text{safe}}(t_i)$ is overapproximated with $d_{\text{safe}}(t_0)$. Next, we predict the future position of the cutting-in vehicle $s_{p,V}(t_i)$ for each time instance t_i, by utilizing (2.18). Let us denote by $s_{\max,E}(t_i)$, $t_i = t_1, \cdots, t_{\text{clear}}$, the maximum admissible position of the ego vehicle (see Fig. 2.9), such that the safe distance is satisfied with respect to the predicted position $s_{p,V}(t_i)$ of the cutting-in vehicle V, then

$$s_{\max,E}(t_i) = s_{p,V}(t_i) - d_{\text{safe}}(t_i). \tag{2.22}$$

Once the maximum allowed position of the ego vehicle is computed, the clearance acceleration $a_{\text{clear}}(t_i)$ is generated such that $\Delta s(t_{\text{clear}}) \geq 0$ where $\Delta s(t_{\text{clear}}) = d(E, V, t_{\text{clear}}) - d_{\text{safe}}(t_{\text{clear}})$. To calculate $a_{\text{clear}}(t_i)$, the following objectives are considered: The position error $\Delta s(t_i)$, and the jerk $j_{\text{clear}}(t_i) = \dot{a}_{\text{clear}}(t_i)$ should be minimized, while constraints on position and acceleration are imposed, as will be subsequently described. Therefore, the following objective function is introduced:

$$J = \sum_{i=1}^{N_{\text{clr}}} \gamma_{s,i}(s_{\max,E}(t_i) - s_E(t_i))^2 + \gamma_u j_{\text{clear}}^2(t_i), \tag{2.23}$$

where $N_{\text{clr}} = \dfrac{t_{\text{clear}}}{\Delta t}$ represents the optimization horizon, γ_u is the weighting coefficient that penalizes the jerk, and $\gamma_{s,i}$ denotes the weighting coefficient penalizing the deviation of the ego vehicle's position s_E at time instance t_i from the corresponding reference position $s_{\max,E}(t_i)$.

To progressively achieve clearance of the violated area within t_{clear}, we propose increasing the values of the weighting parameters $\gamma_{s,i}$, such that the position $s_{\max,E}(t_i)$ is gradually reached $0 < \gamma_{s,i} < \gamma_{s,i+1}$,

$\forall i \in \{1, \cdots, N_{\mathrm{clr}} - 1\}$. Finally, the control problem is formulated as follows:

$$\begin{aligned}
\min_{a_{\mathrm{clear}}(t_i)} \quad & \sum_{i=1}^{N_{\mathrm{clr}}} \gamma_{s,i}(s_{\max,E}(t_i) - s_E(t_i))^2 + \gamma_u j^2_{\mathrm{clear}}(t_i), \\
\text{subject to} \quad & \forall i \in \{1, \ldots, N_{clr}\} : \\
& s_{\max,E}(t_{\mathrm{clear}}) - s_E(t_{\mathrm{clear}}) \geq 0, \\
& a_{\mathrm{clear}}(t_i) \leq 0, \\
& \text{eq. (1) - (3), (13).}
\end{aligned} \tag{2.24}$$

Algorithm 2 Computation of acceleration a_{clear}.

```
Input: $\Delta t,\ t_{\mathrm{clear}},\ s_{\mathrm{E}}(t_0), v_{\mathrm{E}}(t_0),\ s_{\mathrm{V}}(t_0),\ v_{\mathrm{V}}(t_0)$
Output: $a_{\mathrm{clear}}(t),\ t = t_1, t_1 + \Delta t, \ldots, t_{\mathrm{clear}}$
while $t_{clear} \geq 0$ do
    Calc. $d_{\mathrm{safe}}(t_0)$;
    forall $t_i \in \{t_1, t_1 + \Delta t, \ldots, t_{clear}\}$ do
        $d_{\mathrm{safe}}(t_i) \leftarrow d_{\mathrm{safe}}(t_0)$;
        Calc. predicted position of $V$, $s_{p,V}(t_i)$;
        Calc. max. admissible position of $E$, $s_{\max,E}(t_i)$;
        Calc. $a_{\mathrm{clear}}(t_i)$ subject to (2.24);
    $t_{\mathrm{clear}} \leftarrow t_{\mathrm{clear}} - \Delta t$;
return $a_{clear}(t)$
```

This control problem is solved at each time instance t_i, which results in a sequence of control inputs $a_{\mathrm{cleart}}(t)$; however, only the first control input is considered and the rest are discarded. Then, the clearance time t_{clear} is updated accordingly since the time required to clear the violated safe area is reduced, as the time passes. Next, the states are updated and the control problem is solved again, for the new optimization horizon, until the clearance time has passed.

2.3.5 Longitudinal Control Input Selection

After a control input is generated with respect to each selected vehicle (recall Fig. 2.4), only one control input is chosen to be applied to the ego vehicle. If no surrounding vehicle is selected, the cruise control acceleration command a_{CC} is applied such that the desired speed is maintained. Otherwise, if selected surrounding vehicles exist $V_{\mathrm{selected}} \neq \varnothing$, the control input that can achieve safety with respect to all surrounding vehicles must be chosen:

$$a_{\text{long}}(t) = \begin{cases} a_{\text{CC}}(t), & V_{\text{selected}} = \varnothing \\ \min\{a_{\text{ACC}}(t), a_{\text{cut-in}}(t)\} & V_{\text{selected}} \neq \varnothing. \end{cases} \tag{2.25}$$

The acceleration that produces the most defensive driving is then chosen and applied to the ego vehicle. Since all values of the generated acceleration profiles can achieve safety with respect to the corresponding vehicles, selecting the acceleration profile that achieves the most defensive driving ensures a safe behavior considering all selected vehicles.

2.4 Experimental Results

The presented approaches are evaluated with real traffic data for more than 300 vehicles. The data is collected on a segment of US highway 101 (Hollywood Freeway) located in Los Angeles, California, on June 15th, 2005, as part of the Next Generation SIMulation (NGSIM)[1] project.

In this simulation, the vehicles from the dataset are considered as leading vehicles in the ACC setup. For each vehicle, the following information is available at each sampling time: position, velocity, and acceleration. In the typical scenarios, the lead vehicle is driving with variable acceleration; however, in order to make the scenarios even more difficult, sudden brakes are added. The ego vehicle is positioned behind the leading vehicle, with a randomly generated initial velocity and acceleration. For all considered scenarios, the time step Δt was set to 0.1 seconds, and the time horizon N was set to 3 seconds. The allowed range of velocity, acceleration, jerk, and distance error are shown in Table 2.3.

Table 2.3: Allowed value range of different variables for the ACC setup.

Variable	v[m/s]	a[m/s^2]	j[m/s^3]	Δs[m]
Allowed value range	[0,60]	[-10,10]	[-2,2]	[0,10]

Safe MPC-based ACC. Two different deceleration profiles are used to evaluate the safe MPC-based ACC: full deceleration and mixed deceleration. For both cases, we evaluate the arithmetic mean $\overline{j}$, $\overline{\Delta s}$, $\overline{\delta}$, and the standard deviation σ_j, $\sigma_{\Delta s}$, σ_δ associated with the variables $j, \Delta s,$ and δ, respectively, for all considered vehicles. The simulation results are shown in Table 2.4.

Table 2.4: Simulation results of safe ACC.

Brake	$\overline{j}$[m/s^3]	σ_j[m/s^3]	$\overline{\Delta s}$[m]	$\sigma_{\Delta s}$[m]	$\overline{\delta}$[m]	σ_δ[m]
Full	-0.005	0.883	3.369	3.557	22.073	11.244
Mixed	-0.006	0.298	0.287	1.071	23.773	6.080

[1] https://www.fhwa.dot.gov/publications/research/operations/its/06135/

As can be seen in Table 2.4, although the mean jerk value generated by applying full deceleration is small, the standard deviation ($\sigma_j = 0.883$) shows that there is a broader range of jerk values compared to that generated from the mixed deceleration profile ($\sigma_j = 0.298$). Moreover, because of the frequent application of full braking, the safe distance tracking parameter $\overline{\Delta s}$ shows a worse performance, when compared with the case when mixed deceleration is applied. For mixed deceleration, it can be seen that the value of $\overline{j}$ is small, which indicates comfortable driving without jerky maneuvers. The standard deviation σ_j is also small, thus most of the jerk values are close to the mean value. The results show good tracking performance, as the mean value of Δs is small. The average distance between vehicles is around 23m, which is comparable to the distance provided by the two-seconds distance rule[1] [124], which depends on the velocity.

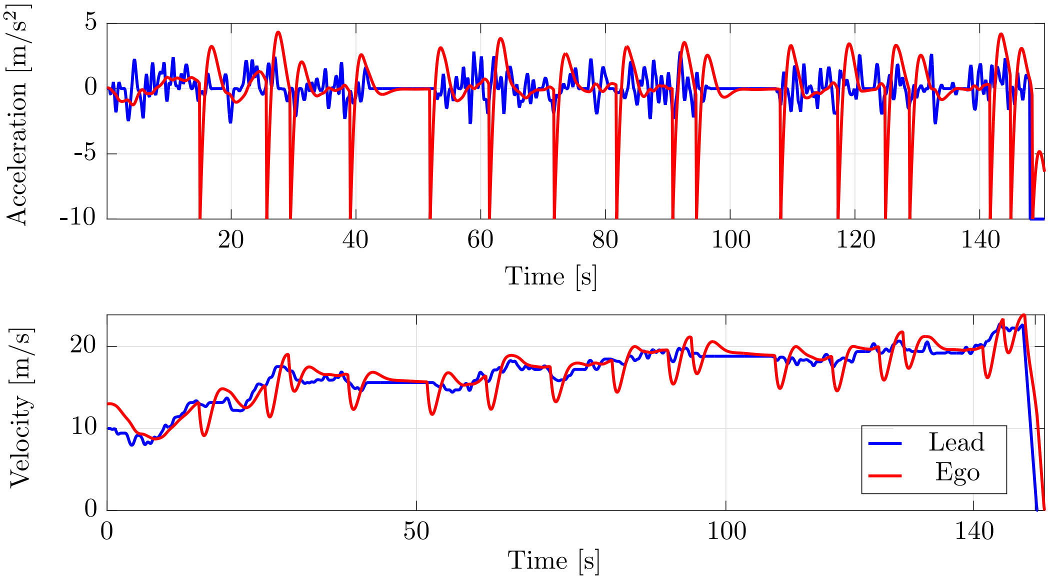


Figure 2.10: Safe MPC-based ACC: Full deceleration.

For illustration purposes, we only present the detailed simulation results for one considered scenario, where the time duration of recording took more than 2 minutes. The simulation results when applying the full deceleration profile are depicted in Fig. 2.10. It can be seen that large variations in the ego vehicle's acceleration lead to large variations in velocity (see Fig. 2.10). Moreover, the jerk caused by frequently applying full brake induces uncomfortable driving.

Platooning using safe MPC-based ACC. In the following, we propose a four-vehicle setup to evaluate the string stability. The trajectory of vehicle #1 is taken from the US101 dataset. The other vehicles are driving one behind each other, directly behind vehicle #1. The three vehicles are controlled

[1] http://www.rotr.ie/rules-for-driving/speed-limits/speed-limits_2-second-rule.html

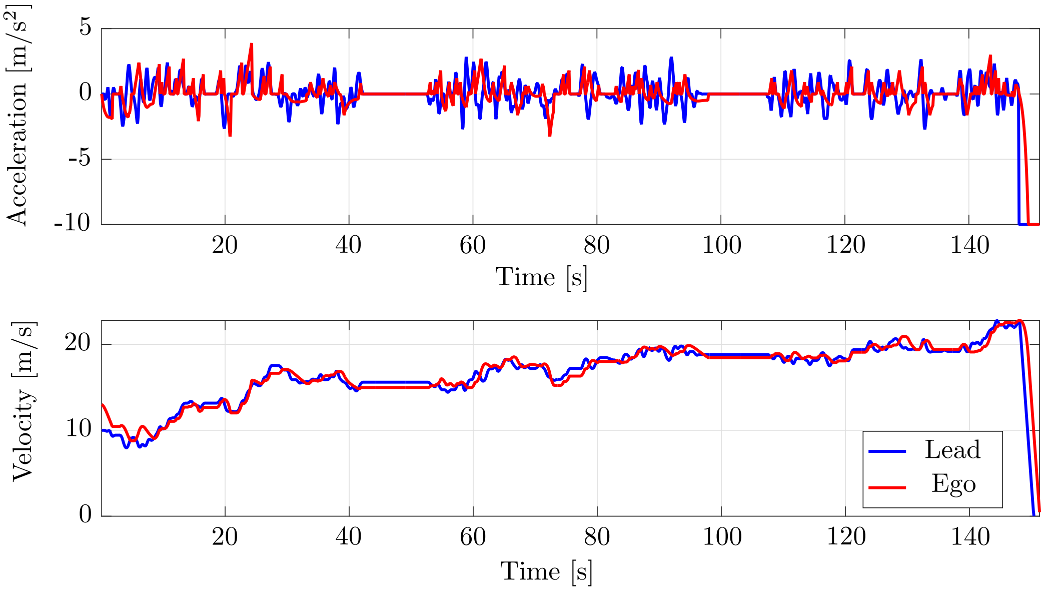


Figure 2.11: Safe MPC-based ACC: Mixed deceleration.

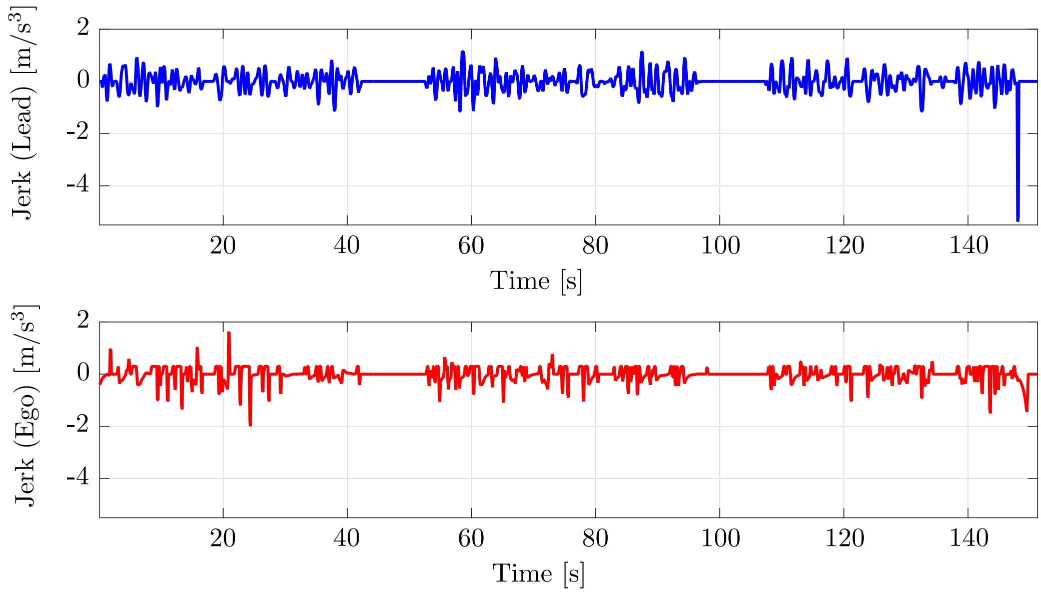


Figure 2.12: Safe MPC-based ACC: Jerk values for mixed deceleration.

by our proposed algorithm with the task that vehicle #2 safely follows vehicle #1, vehicle #3 follows vehicle #2, and vehicle #4 follows vehicle #3.

The velocities of the ACC-equipped vehicles (#2, #3, #4) smoothly follow the velocity of the leading vehicle (see Fig. 2.13) when the mixed deceleration profile was applied. In approximately 10% of the considered time, the emergency maneuver is engaged in order to not violate the safe distance. Still, jerk values are maintained in the specified comfortable value range [74]. Even while the lead vehicle suddenly performs full braking, the ACC-equipped vehicles smoothly decelerate. Additionally, the position error Δs introduced by the leading vehicle's braking is attenuated in the upstream direction, as illustrated in Fig. 2.13.

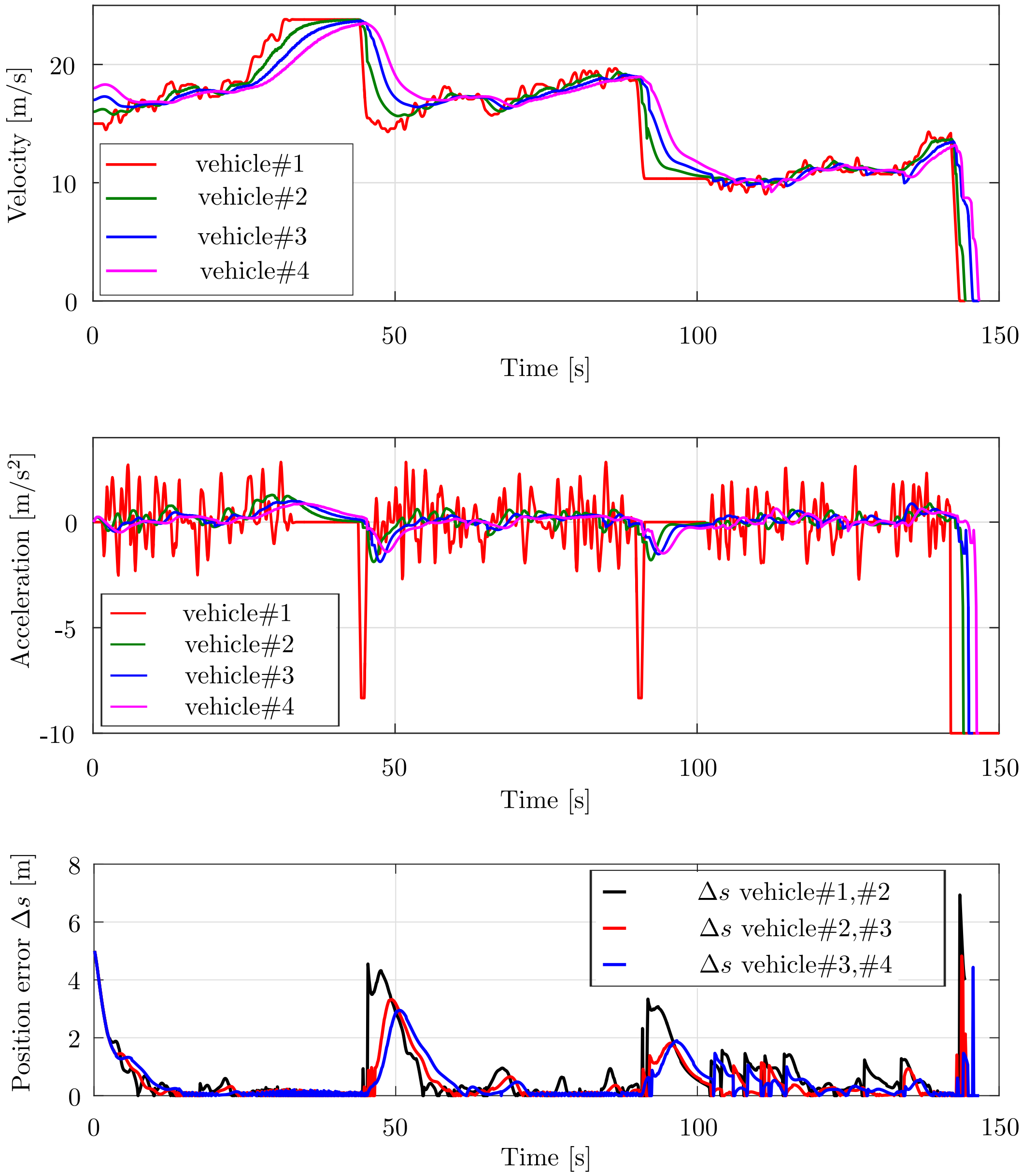


Figure 2.13: Safe MPC-based ACC: Platooning setup with mixed deceleration.

The mean jerk values $\bar{j}$ and the standard deviation σ_j are small (see Table 2.4), which implies comfortable driving without jerky maneuvers. Maintaining the inter-vehicle distance as close as possible to the safe distance d_{safe} by minimizing Δs shows good tracking performance. In this way, both safety and comfort are achieved by utilizing the proposed ACC concept.

All simulations were performed on a machine with 2.2 GHz, Intel i7 processor and 16 GB 1600 MHz DDR3 memory. The mean value of the computation time is 0.08s, showing that this approach is real-time capable.

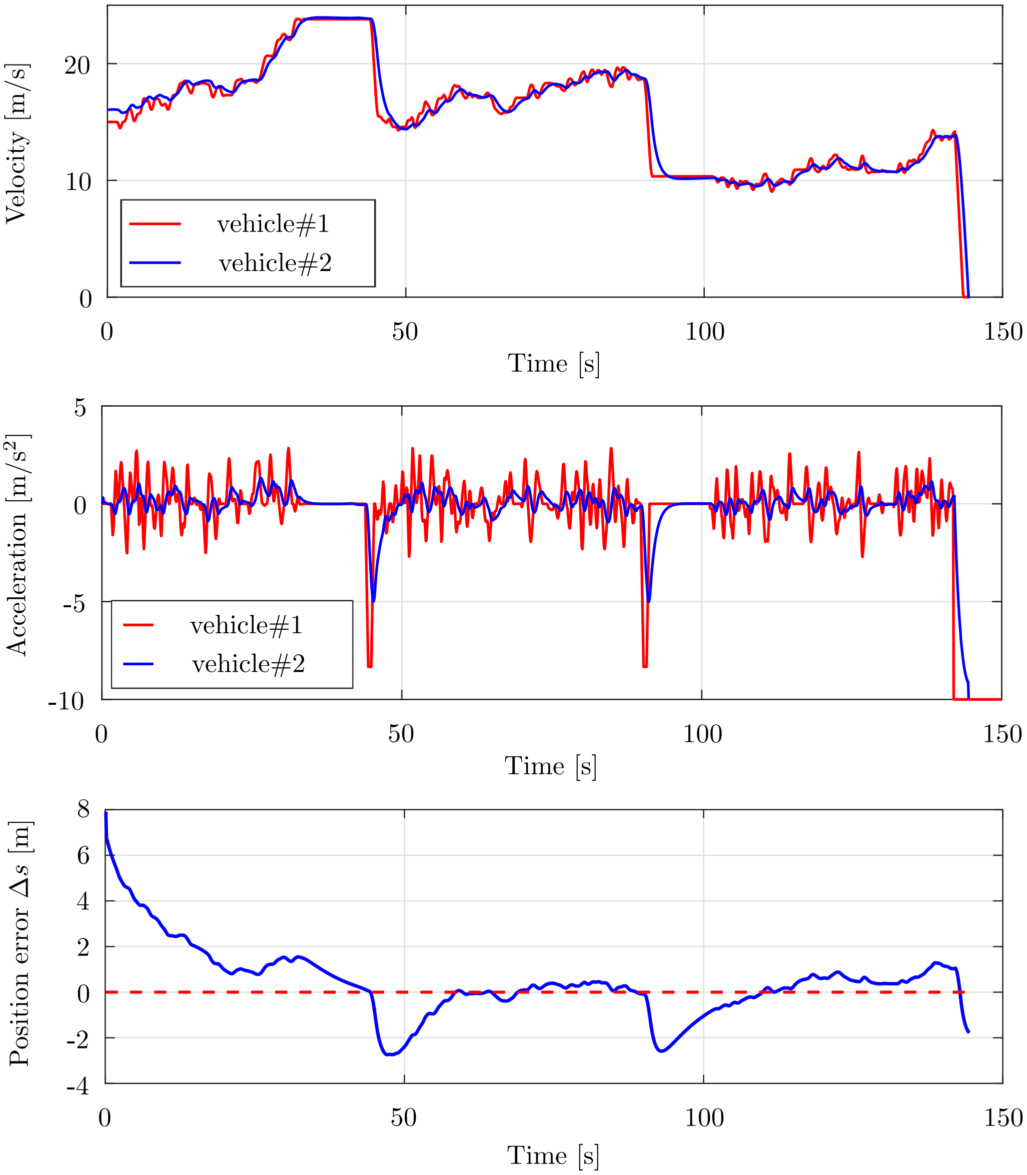


Figure 2.14: Simulation results: PI-based ACC.

PI-based ACC. Lastly, we evaluate our method against a state-of-the-art ACC approach applied in the automotive industry [39, 200], which utilizes proportional-integral control (PI). Here, we use an implementation based on [200] where the desired inter-vehicle distance is a function of constant spacing, constant time headway, and the velocity of the leading vehicle. Of course, other spacing policies can be used, as proposed in the paper [200] (e.g., variable time headway).

Even though the performance of the selected algorithm [200] shows to be effective with respect to position and velocity tracking, the PI controller itself cannot guarantee safety. Since the position errors Δs have negative values, as illustrated in Fig. 2.14, the controller fails to safely track the desired inter-vehicle distance.

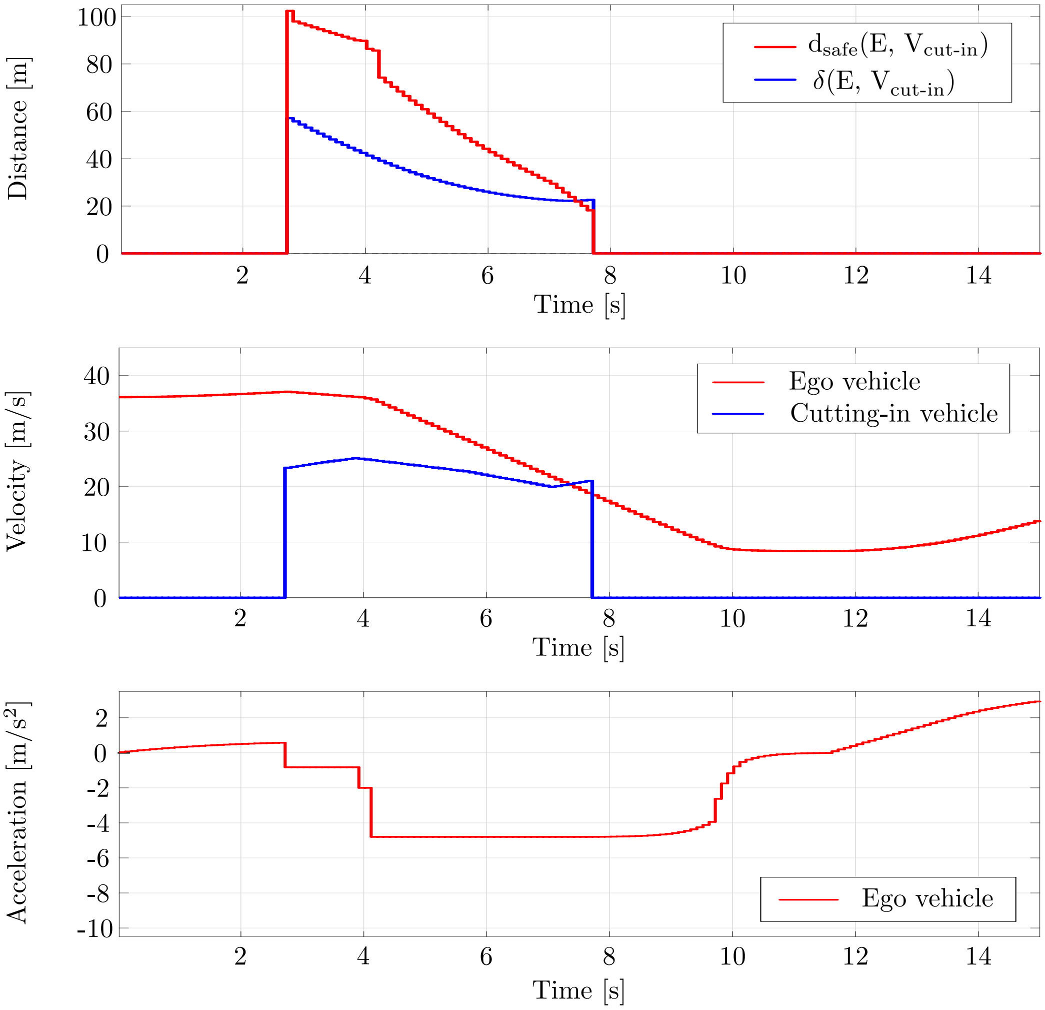


Figure 2.15: Simulation results: Reaction to a cutting-in vehicle.

Reaction to cut-in maneuvers. In this section, the reaction to a cut-in maneuver is evaluated. The computation was performed on a computer with a 2.40 GHz Intel Core i5-6200U processor.

A cut-in maneuver was recognized at approximately 2.3 seconds after the simulation started. The initial relative velocity between the ego vehicle and the vehicle for which a cut-in maneuver was predicted, was approximately 13 m/s and the inter-vehicle distance was about 60 m, which was significantly lower than the required safe distance of approximately 100 m. Throughout this simulation, the clearance time t_{clear} was set to 7 seconds. The violated safe distance was, however, cleared in less than 6 seconds due to the corresponding braking of the ego vehicle, as shown in Fig. 2.15.

Safe Longitudinal Control Evaluation. In [123], a slightly modified[1] version of the presented safe longitudinal control was evaluated by a user study performed by using a driver simulator provided by the automobile manufacturer BMW Group. The aim of this user study was to assess the comfort and safety of the entire system from a user experience point of view and to provide a comparison to a state-of-the-art ACC system. The considered traffic scenarios, the questionnaire used, and the evaluated hypotheses are listed in the Appendix.

The results of the evaluation are shown in Table 2.5. The grey-colored cells represent the scenarios and adjacent to that, are the evaluated hypothesis for which the results are significant. For the other three cases (Scenario 1 — both hypotheses, and Scenario 5 — the comfort-related hypothesis), no significant results were concluded since it was not possible to differentiate between the behavior of the safe longitudinal control and the state-of-the-art ACC. Thus, both hypotheses have been validated over the majority of the scenarios considered through the user study and this shows a comfortable and safe-feeling user experience together with a performance similar to that provided by the considered state-of-art ACC system.

Table 2.5: Evaluation of the user study results [123].

Scenario	Evaluated hypothesis
1) Emergency brake of the first leading vehicle	Safety
	Comfort
2) Standstill second leading vehicle	Safety
	Comfort
3) Aggressive cut-in maneuver	Safety
	Comfort
4) Long cut-in maneuver	Safety
	Comfort
5) Entering a traffic jam	Safety
	Comfort

2.5 Conclusions

In this chapter, a novel architecture for the longitudinal control of autonomous vehicles with safety guarantees has been proposed.

The distinctiveness of the presented longitudinal control framework is that safety can be formally guaranteed by having an emergency maneuver available at all times, that ensures the ego vehicle can be brought to a safe state. Thus, safety is ensured even in unexpected critical traffic scenarios, e.g., the

[1] The main modification is represented by introducing a jerk optimal transition acceleration profile between two consecutive maneuvers.

closest leading vehicle performs a full brake maneuver or it changes lanes due to a slower vehicle driving ahead that could bring the ego vehicle in a dangerous situation if no appropriate deceleration is applied. One unique feature introduced by the presented approach is that the required safe distance considers not only the most likely behavior, but full deceleration of the vehicle driving ahead, which makes possible to safely react in extreme traffic situations. Moreover, the precomputed deceleration profile that brings the ego vehicle to a safe state can be tuned such that either a more aggressive reaction or a milder one can be used when the emergency maneuver is engaged. This allows an adapted response to a dangerous situation based on user preferences, which is then reflected in the required inter-vehicle safe distance.

In most previous work, an emergency maneuver is usually generated only when a dangerous situation occurs. However, this may lead to a collision since it is possible that no feasible emergency maneuver exists anymore at the time when the dangerous situation is triggered. On the other hand, our approach keeps an emergency maneuver available at all times that accounts for the worst-case behavior of the other traffic participants. Thus, one of the key benefits introduced by our technique is that the pre-computed emergency maneuver can bring the ego vehicle to a safe state even if a critical traffic situation occurs.

Another advantage compared to previous work is that safety is guaranteed not only with respect to the closest leading vehicle but with all relevant vehicles driving ahead. By keeping a pre-computed emergency maneuver available that corresponds to each selected surrounding vehicle, we avoid a situation that can lead to an imminent collision state.

The proposed framework was tested via simulations that demonstrated its performance and that the approach is safe, and yet, it does not introduce vehicle behavior that is too conservative. One of the most significant assets of our framework — apart from guaranteeing safety — is represented by its modularity. Due to the flexibility of the proposed framework where each part can be exchanged with different methodologies (e.g.: various deceleration profiles that characterize the emergency maneuver can be used), this approach can be easily applied in practice.

Our approach, unfortunately, cannot handle the situation where an obstacle suddenly appears in front of the ego vehicle (e.g., a falling object from an above bridge). Although, once the dangerous situation is assessed and the ego vehicle would begin mitigating an imminent collision, safety can no loner be achieved. However, to the best of our knowledge, there is no approach in the literature that can guarantee absolute safety.

Chapter 3

Overapproximative Occupancy Set Computation of Traffic Participants

One of the main challenges facing safety guarantees for autonomous vehicles is how to cope with the unknown future behavior of surrounding vehicles. In the previous chapter, we addressed the safety problem for a specific scenario, which is, considering that the ego vehicle remains in the lane in which it is currently driving. However, to guarantee safety in more complex traffic scenarios, a prediction algorithm that considers all possible future behaviors of the other traffic participants is required. Most of the previous work tackled this problem by calculating the probability distribution over time of other traffic participants' future behavior or by generating multiple possible future trajectories. These approaches, however, cannot guarantee that the planned trajectory for the ego vehicle is collision-free. Instead of using non-formal prediction techniques such as single behaviors or probabilistic approaches to predict the future behavior of the other traffic participants, in this chapter, we calculate an over-approximation of all possible occupancies of surrounding traffic participants over time. This makes it possible to prove whether a planned trajectory of an autonomous vehicle can safely be followed or whether an emergency maneuver is required to avoid a possible collision.

The remainder of this chapter is organized as follows: In Sec. 3.1, a review of the related literature is presented and the contributions of this chapter are introduced. Sec. 3.2 presents the basic idea behind the integration of our occupancy prediction to a collision avoidance concept along with the constraints made on the behavior of other traffic participants. In Sec. 3.3, the representation of the road network and the mathematical modeling of other traffic participants are described, while Sec. 3.4 introduces the algorithm for calculating the occupancy prediction. Sec. 3.5 presents the experimental results, including the comparison with a high-fidelity vehicle model and real traffic data. In the final part of this chapter, Sec. 3.6, the conclusions are presented.

This chapter is based on [10].

3.1 Introduction

To guarantee safety for autonomous vehicles, provably correct decision algorithms using formal methods are required [6, 93, 132]. The capability to cope with different levels of uncertainty — in both measurements (e.g., road boundaries, static obstacles, or traffic participants) and future behavior of other traffic participants — represents one of the most challenging aspects facing the development of autonomous vehicles.

The approach we propose enables us to generate formally correct maneuvers, despite the aforementioned uncertainties, by computing the set of future occupancies of other traffic participants. We can therefore guarantee that the ego vehicle will not produce a collision if no intersections between the occupancy of the ego vehicle and the predicted occupancy set of the other traffic participants are detected for all points in time [6].

In order to plan a safe trajectory for the ego vehicle, several aspects must be considered such as: planning a maneuver [107] that avoids the predicted future occupancies of other traffic participants, maneuver recognition [40, 113, 199], and trajectory prediction of the surrounding vehicles. Predictions that are made for particular road sections, such as merging lanes, or different intersections, are mainly addressed by using machine learning techniques [77,174,186]. In this chapter, however, we will tackle only the computation of the predicted occupancy sets of other traffic participants, while the framework that shows how the occupancy sets can be integrated into maneuver planning problem will be subsequently presented in Chapter 4. In [111], a comprehensive survey on behavior prediction has been published where the considered literature was categorized into physics-based, maneuver-based, and interaction-aware prediction. In the following, we provide a literature review organized into four main categories [10]: approaches computing (*i*) a single future behavior, (*ii*) a countable set of future behaviors, (*iii*) probability distributions of future behaviors, and (*iv*) uncountable sets of future behaviors; this categorization is somewhat orthogonal compared to the one presented in [111].

Single future behavior. Most of the previous work that tackles the prediction of future behaviors of surrounding traffic participants generates only the most probable future behavior [24, 31, 44, 96, 192]. These approaches are widely used because they provide a straightforward way to consider the prediction of the other traffic participants in motion planning or in collision prediction problems. In [24], the future position of surrounding vehicles is estimated by assuming constant acceleration and yaw rate, which are tracked by an extended Kalman filter. To generate evasive trajectories for the ego vehicle [44], single behavior predictions are completed for each traffic participant. Since the maneuver planner in [192] targets comfortable driving under consideration of social behavior, only a single future behavior of surrounding traffic participants is considered. In [31], in order to determine whether a collision can be avoided through steering and/or braking, only one trajectory is predicted for each surrounding road user. In [96], the risk assessment of traffic situations is performed by considering multiple possible maneuvers of the ego vehicle. However, only one single future trajectory is calculated for the other traffic participants.

Various heuristic methods based on a single behavior — often referred to as *surrogate measures* — have been developed for driving assistance systems where fast computation is prioritized higher than the formal correctness of warnings. Time to collision (TTC) [188] is one of the most known surrogate measures that assumes the velocities and directions of vehicles are constant, and thus computes the time when a crash would occur under these assumptions. Various extensions to TTC have been proposed such as time-exposed TTC and time-integrated TTC [131], as well as combinations of several surrogate measures [179].

Countable set of future behaviors. Since considering single trajectories for the other traffic participants may be insufficient, and to mitigate the fact that infinitely many future trajectories can exist, many previous works instead consider a finite number of future behaviors. Usually, these methods are widely used for collision assessment. In [85], in order to compute the time to trigger an emergency brake, multiple physically possible trajectories of the ego vehicle and one of the other vehicles are generated offline. Multiple simulations are often weighted by probabilities, which are also known as Monte Carlo simulations. This technique is often used for online threat assessment of vehicles [32, 42, 45] or to create a threat database [98]. Motion clusters in combination with a particle filter are used in [69] to determine the most likely future motion. Nevertheless, these approaches generate a finite number of predictions that cannot guarantee safety since infinitely many traffic scenarios exist.

Probability distribution of future behaviors. Another cluster of approaches addresses the fact that infinitely many possible future behaviors exist. Instead of generating a countable set of predicted trajectories, the probability distribution of possible future behaviors is computed. The behavior of other traffic participants is predicted using Dynamic Bayesian Networks in [59]. Another probabilistic approach has been suggested in [106] where Gaussian distributions are used to represent the future occupancy with a special focus on the efficient computation of the collision probability. In [95], a lane-based probabilistic distribution is used to predict the future behavior of the surrounding traffic participants, which is then used for collision risk assessment. Nevertheless, safety cannot be formally guaranteed by any of these aforementioned approaches, since a collision probability of zero does not imply that a collision is not possible. This, however, is guaranteed in [13] by formally abstracting traffic participants to Markov chains using reachability analysis. In [11], a comparison of this approach with Monte Carlo simulations has been performed. Yet, since only a finite number of possible trajectories are generated with respect to the probability distribution among the lanes, safety still cannot be guaranteed.

Uncountable set of future behaviors. The necessity of generating the set-based prediction for other traffic participants is highlighted in the framework proposed in [185]. However, no algorithm that can formally compute occupancy prediction is proposed.

In [6], techniques from reachability analysis are utilized to rigorously compute an over-approximation of the occupancy of surrounding vehicles. Since the computation is done such that the resulting predicted sets are over-approximative, and therefore considers all possible behaviors, this approach is predestined for certification (e.g., [180]). Generating the set-based prediction of traffic participants is a challenging task due to their dynamics that are typically nonlinear and subject to constraints (e.g., one has to exclude behaviors, such as leaving the road boundary). Since those constrains cannot be formulated as bounds of inputs that are independent of the current state of the system, traditional techniques for reachability analysis cannot be applied. Previous work can be grouped by their applied set representation: polytopes [36], zonotopes [12], zonotope bundles [8], rectangular grids [41], ellipsoids [103], support functions [60], oriented rectangular hulls [173], and axis-aligned boxes [68]. However, standardized set representations can only be used when unrealistic behavior does not have to be excluded; otherwise, the prediction may result into non-convex or disjoint reachable sets [164, 171].

In [62], the authors tackled the prediction problem, which is required for collision risk assessments, by considering a combination of set-based and stochastic prediction methods. Nevertheless, the set-based computation presented in [62] is heuristic and it does not fit into a formal analysis. Reachability analysis has multiple field applications such as driving assistant systems [46, 93] or mobile robotics. Within the field of mobile robotics there are many applications of reachable sets, but they mostly use overly simplified models for road transportation applications. For example, in [184] intervals of possible velocities in all directions are considered whereas more complex models assume intervals of both velocity and acceleration [30]. More complex models are based on Dubin's car [196] or a tricycle model [35].

3.1.1 Contributions

The review of the existing work shows that a formal approach for computing the occupancy prediction for other traffic participants is required in order to guarantee safe maneuvers. The concept presented in this chapter is an extension of the work described in [6, 7] where it was shown that the occupancy prediction of other traffic participants can be computed in a formal manner via reachability analysis. The approach presented in this chapter significantly extends previous work in the following ways:

- We present a framework that can formally predict the occupancy area of other traffic participants, on an arbitrary road network. Compared to previous work [6, 7] where only single lanes without forks or joints were considered, the new presented framework makes it possible to predict the future behavior of surrounding vehicles on multi-lane roads, as well.

- While previous work [6, 7] assumed that the shortest path through a road network (as later defined in Problem 1) exists, in this work an efficient algorithm that provides an over-approximation for the fastest way though a lane is presented.

- In contrast to the previous work, here we consider arbitrary lane changes made by other traffic participants.

- We validate our approach against real traffic data for the first time and verify whether the predicted sets always enclose the corresponding occupancy of the recorded trajectory of the surrounding vehicles.

3.2 Motivation and Objective

The proposed concept is described as follows: a planned trajectory of an autonomous vehicle is defined as being safe if its corresponding occupancy does not overlap with the occupancy of any surrounding traffic participant at any moment. A limited timeframe, however, contains infinitely many points in time. Thus, one cannot guarantee that a collision will not occur by only verifying if there is an intersection between the occupancy of the ego vehicle and the other traffic participants for discrete points in time. To check if a given trajectory is free from collision, we verify if a possible intersection exists for successive time intervals as shown in Fig. 3.1. We can, therefore, guarantee safety for the reference trajectory if no intersection exists for consecutive time intervals between the occupancy of the ego vehicle and occupancy of the other vehicles by doing only a finite number of collision checks. If the time step for which the occupancy sets are calculated is too large, the results from the collision check may be too conservative. In other words, if the time step is too large, an intersection between the corresponding occupancy sets of the ego and surrounding vehicles could be detected regardless of the safe behavior of the ego vehicle. Nevertheless, this can be tackled by recursively separating time intervals $[t_k, t_{k+1}]$ into $[t_k, \tilde{t}]$ and $[\tilde{t}, t_{k+1}]$ with $\tilde{t} = 0.5(t_k + t_{k+1})$ for which occupancy sets are intersecting. For the initial time step $t_{k+1} - t_k$ we suggest a value of 0.5 s.

In Fig. 3.2 we illustrate the method for calculating safe reference trajectories for the ego vehicle. Next, we introduce some various types of trajectories, that will better guide the reader through the figure.

- *Long-term reference trajectory*: We assume that a trajectory planner provides a trajectory for the ego vehicle, that is calculated for a given time horizon (typically, this time horizon is a couple of seconds). Note that this long-term reference trajectory is generated based on a non-formal prediction of other traffic participants.

- *Intended trajectory*: First part of the long-term reference trajectory that is subject to formal verification.

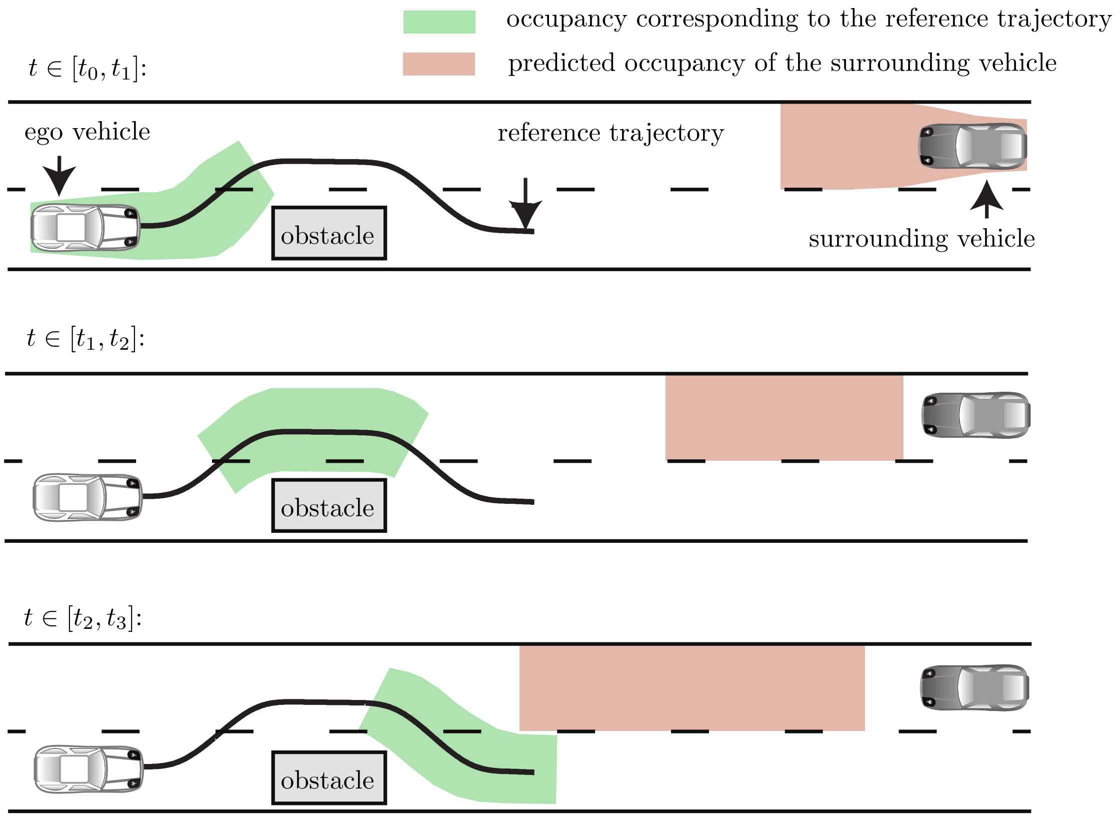


Figure 3.1: Occupancy of traffic participants for selected time intervals [10].

- *Fail-safe trajectory*: A trajectory that brings the vehicle to a safe state, such as a safe distance behind another vehicle or at standstill in a shoulder lane.
- *Potential trajectory*: Obtained as the concatenation of the intended trajectory and a fail-safe trajectory, which has not yet been verified as safe.
- *Safe trajectory*: The potential trajectory that has been verified as safe. The verification of the safe trajectory is done for all times. Note that a collision may be inevitable if the reference trajectory is replanned [149] and if the reference trajectory would not result in a safe state.

At each time instance, the procedure illustrated in Fig. 3.2 is performed, and continuously repeated such that all changes that occur in the considered traffic scenario are included. We begin with a verified safe trajectory of the ego vehicle from the previous time step t_{k-1} (Fig. 3.2, ①). Then, the predicted occupancy of other traffic participants is calculated, using the method that will be subsequently described in this chapter (Fig. 3.2, ②). Please note that in order to keep the illustration compact, we do not separately show the occupancy for consecutive time intervals, but instead only show the union of partial occupancy sets. New potential trajectories are planned based on the occupied regions over time (Fig. 3.2, ③). The further potential trajectories are then generated starting from the previous one at a point B

(see Fig. 3.2, ③), which is chosen based on the observation that the computation time for the verification t_{comp} is roughly linear in the prediction time horizon t_h: $t_{comp} \approx \lambda t_h$ ($\lambda \in \mathbb{R}, \lambda > 0$) where λ depends on the computational capabilities. In most cases, to obtain the verified result of the new potential trajectory upon arriving at B, we choose B to be the point at which the vehicle will arrive at time $\lambda t_h + \epsilon$ ($\epsilon \in \mathbb{R}, \epsilon > 0$); larger values of ϵ provide more conservative results but limit the risk of triggering the fail-safe trajectory in case the verification result is not obtained in time.

We define a potential trajectory (i.e., a concatenation of the intended and the fail-safe trajectories) as safe for the ego vehicle following it if its corresponding occupancy never overlaps the occupancy of another traffic participant. Note that multiple further potential trajectories may be verified as safe (see Fig. 3.2, ③). Out of those safe trajectories, the one that minimizes a user-defined cost function is selected (Fig. 3.2, ④ b). However, if no further safe trajectory is found, the previous trajectory is continued (Fig. 3.2, ④ a), which may result in steering along the fail safe trajectory. If there is enough time to replan, and new safe trajectories are found, the ego vehicle may never engage the fail-safe trajectory.

As illustrated in Fig. 3.2, ③, the larger occupancies of other vehicles obstruct space for the trajectory planning of the ego vehicle. Given that the reach of the resulting occupancy prediction sets is directly proportional to the prediction horizon, trajectory planning should be done for two time horizons in parallel. In addition to long-term trajectories that are generated based on non-formal occupancy prediction techniques, such as single behaviors or probabilistic methods, as shown in the upper illustration of Fig. 3.3, emergency trajectories that consider the occupancy sets of the other vehicles are computed. In the upper illustration Fig. 3.3, an example of an overtaking planned maneuver is illustrated, which considers only the single predicted behavior of the other traffic participant. Note that we do not apply our verification concept over the entire intended long-term trajectory. Since the uncertainty of behaviors of the surrounding traffic participants grows with time, the reach of the predicted occupancy sets would grow and cover most of the road, making it almost impossible to verify the entire planned trajectory. Instead, we apply our verification concept only to short potential trajectories (the first part of a long-term reference trajectory plus a fail-safe trajectory) as depicted in the bottom illustration of Fig. 3.3. Since we are using a short time horizon for verification, the proposed set-based techniques do not obstruct overly large regions for trajectory planning. If the maneuver is safe, the next part of the long-term plan is executed; otherwise, the fail-safe maneuver is engaged. As a result, the proposed set-based occupancy prediction guarantees safe maneuvers, while non-formal techniques provide long-term plans based on likely behaviors of other traffic participants.

This chapter proposes a framework that automatically generates the occupancy sets of the surrounding traffic participants on arbitrary road networks. Because this framework should adhere to real-time constraints, a fast and robust algorithm for computing the occupancy prediction is required. Given that it is provably impossible to compute exact occupancy sets [151], a method that allows one to calculate

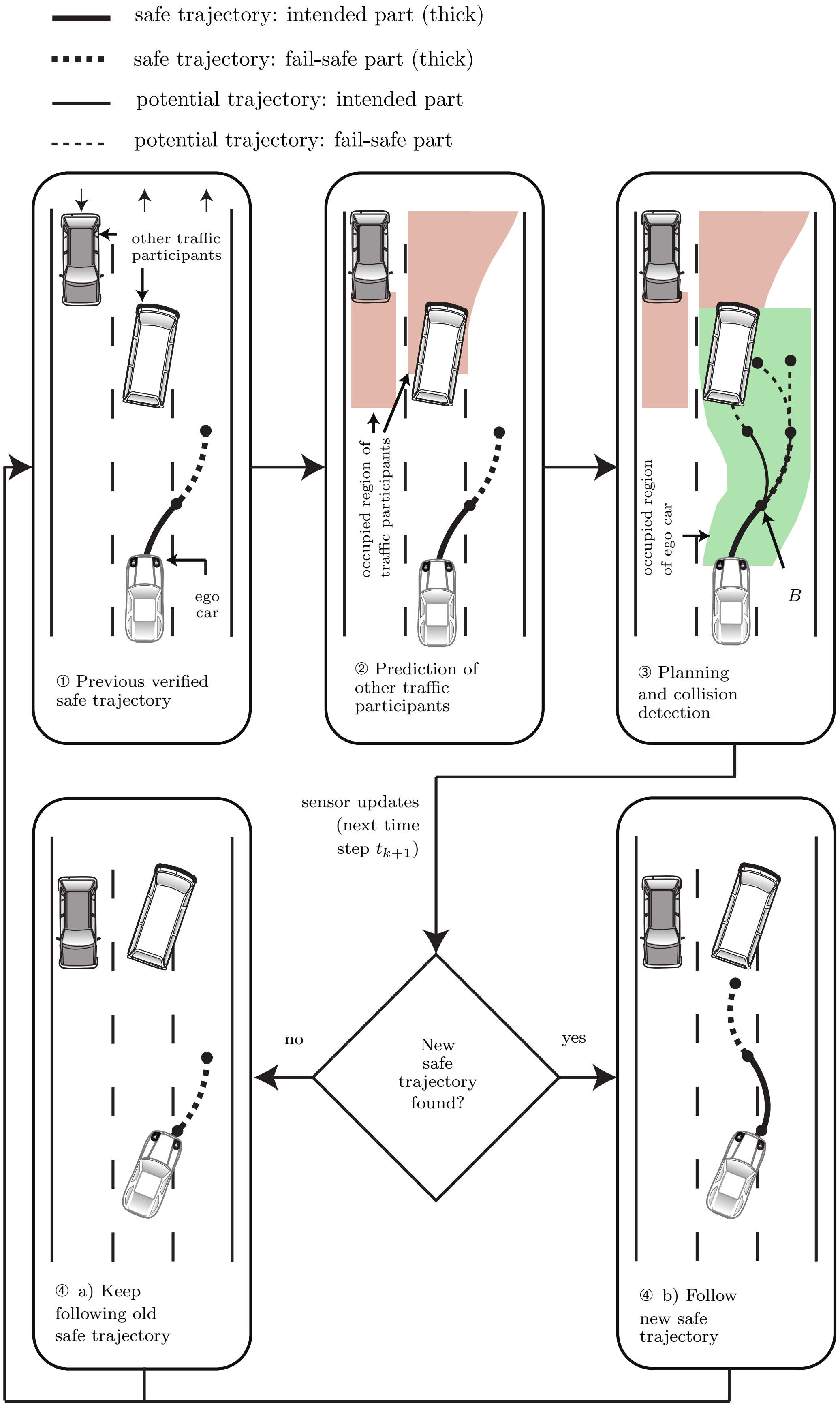

safe trajectory: intended part (thick)
safe trajectory: fail-safe part (thick)
potential trajectory: intended part
potential trajectory: fail-safe part
other traffic participants
ego car
① Previous verified safe trajectory
occupied region of traffic participants
② Prediction of other traffic participants
occupied region of ego car
B
③ Planning and collision detection
sensor updates (next time step t_{k+1})
New safe trajectory found?
no
yes
④ a) Keep following old safe trajectory
④ b) Follow new safe trajectory

Figure 3.2: Overview of online verification of automated vehicles [10].

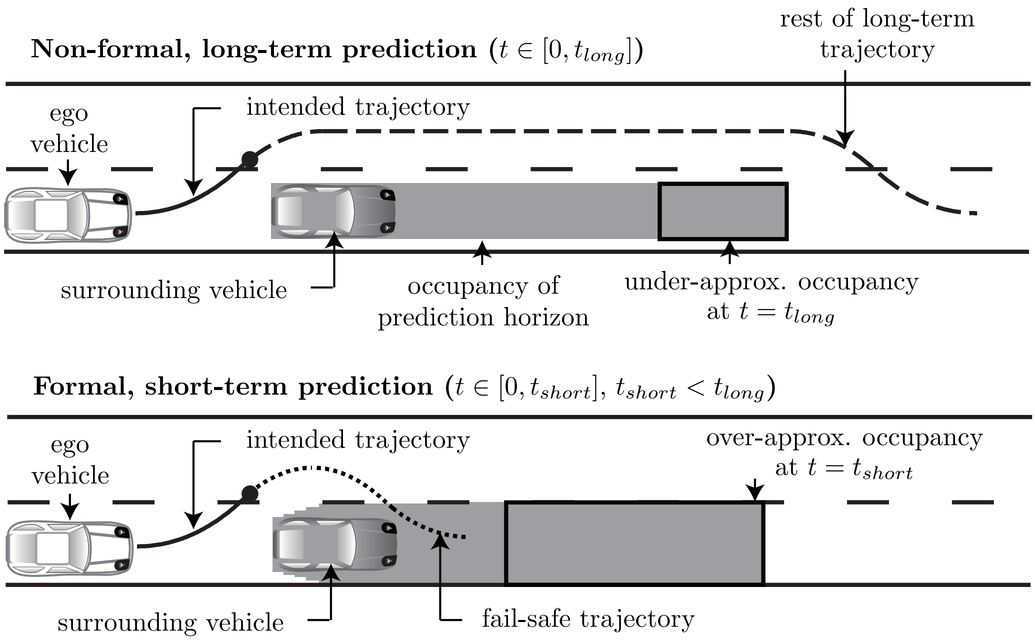

Non-formal, long-term prediction ($t \in [0, t_{long}]$)
rest of long-term trajectory
ego vehicle
intended trajectory
surrounding vehicle
occupancy of prediction horizon
under-approx. occupancy at $t = t_{long}$
Formal, short-term prediction ($t \in [0, t_{short}]$, $t_{short} < t_{long}$)
ego vehicle
intended trajectory
over-approx. occupancy at $t = t_{short}$
surrounding vehicle
fail-safe trajectory

Figure 3.3: Comparing non-formal, long-term prediction with formal, short-term prediction of other traffic participants [10].

an overapproximation prediction is required. The unique feature of our approach is that we specifically consider uncertainties in both the future behavior of traffic participants and their measurements of position, orientation, length, width, velocity, and acceleration. Next, we introduce the model of other traffic participants and a model of the road network that are used for the oveapproximative occupancy set prediction.

3.3 Mathematical Modeling

In the following, a formal representation of road networks and the vehicle model used for predicting occupancy sets are introduced. The road network is based on the concept of *lanelates* introduced in [27], and the model of the surrounding traffic participants is tailored such that their unknown future behavior is considered.

3.3.1 Road Network Representation

For predicting the occupancy sets of the other traffic participants, we first require a formal and robust representation of road networks. For this purpose, *lanelets* [27] are used, which are atomic, interconnected, and drivable road segments.

Definition 3.1 (Lanelet [27])**.** A lanelet is defined by its *left* and *right bound* where each bound is represented by an array of points (a polyline), as illustrated in Fig. 3.4. □

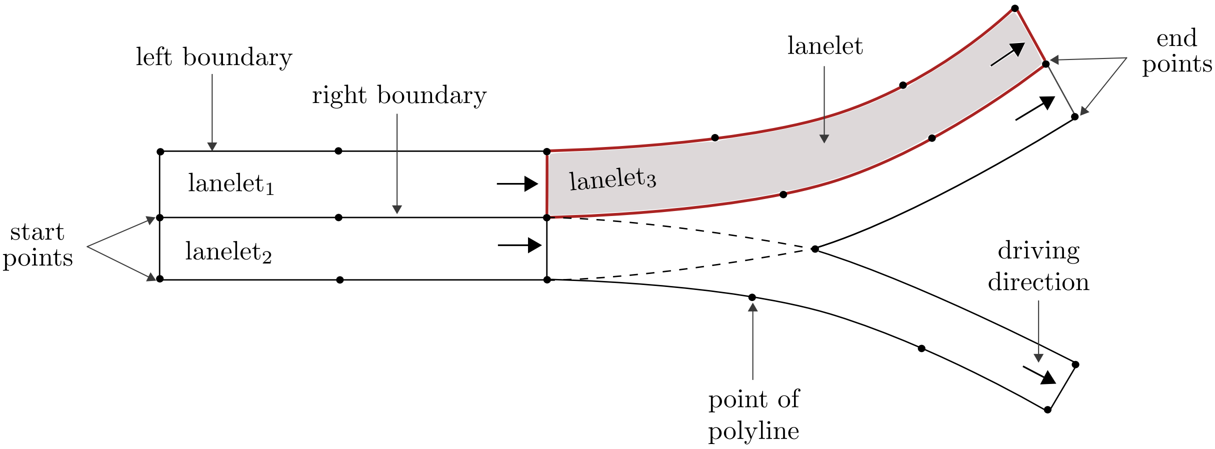


Figure 3.4: Lanelets description.

Additionally, we introduce the *start points* and *end points* of a lanelet as the first and final points of the left and right border in driving direction, respectively, as illustrated in Fig. 3.4.

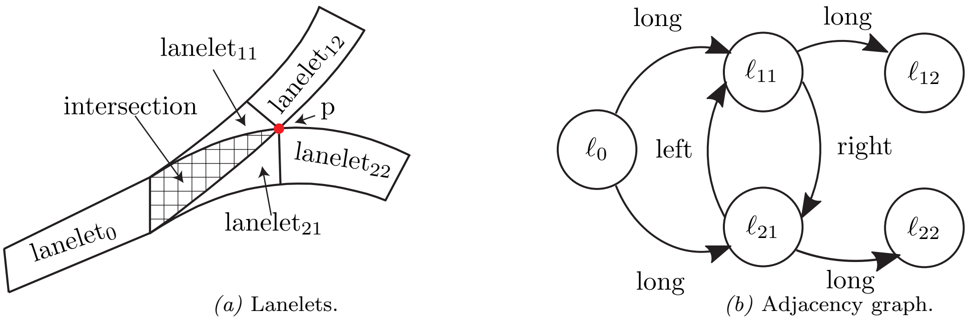
Among the other environment inputs required by autonomous vehicles, road network information (in particular, lane information, e.g. [205]) is expected to be available. The left and right bounds of lanelets could be directly determined from an open-source map like OpenSteetMap[1], by e.g. importing a raw map into the free *JavaOpenStreetMap* (JOSM)[2]. The raw data consisting of a *left border* and *right border* is annotated by the speed limit (maximum allowed speed on that particular lanelet) and a unique *ID*.

Since all possible paths through the current road network must be considered such that an overapproximative prediction can be generated, an *adjacent lanelet matrix* is computed firstly. Let us define the road network as a *directed graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the set of vertices $\mathcal{V}$ corresponds to the lanelets, while the directed edges set $\mathcal{E}$ contains the possible transitions between two adjacent lanelets. Moreover, each vertex has four different types of outgoing edges: longitudinal, left, right, empty. $A_\mathcal{G}$ represents the adjacent lanelet matrix and is defined as follows: $A_\mathcal{G} : \mathcal{V} \times \mathcal{V} \to \{\text{long}, \text{right}, \text{left}, \varnothing\}$, where × denotes the Cartesian product. Without loss of generality, we assume that all laterally adjacent lanes have the same length. This is illustrated for the road network shown in Fig. 3.4, where lanelet$_1$ and lanelet$_2$ have the same length. In this way, the number of lateral adjacencies for multi-lane roads can be reduced.

We define two lanelets as *longitudinally adjacent*, i.e., $A_\mathcal{G}(\text{lanelet}_1, \text{lanelet}_3) = \text{long}$, if the left and right start points of a lanelet coincide with the corresponding final points of the other lanelet. Lanelet lanelet$_2$ is right-adjacent to lanelet$_1$, i.e., $A_\mathcal{G}(\text{lanelet}_1, \text{lanelet}_2) = \text{right}$, if the points of the right border of lanelet$_1$ are identical to the ones of the left border of lanelet$_2$. Similar definition applies by analogy for the left-adjacent lanes. Lanelet lanelet$_1$ is left-adjacent to lanelet$_2$, i.e., $A_\mathcal{G}(\text{lanelet}_2, \text{lanelet}_1) = \text{left}$, if the points of the left border of lanelet$_2$ are identical to the ones of the right border of lanelet$_1$. In

[1] www.openstreetmap.org

[2] https://josm.openstreetmap.de



(a) Lanelets. *(b)* Adjacency graph.

Figure 3.5: Road fork description.

practice, however, the connection points of the lanelets may not be identical (e.g., due to measurement uncertainties). Therefore, the implementation should accept small deviations in the connection points of lanelets when constructing the adjacency matrix.

If the considered road network contains only lateral and longitudinal adjacent lanes, the construction of the adjacent road matrix is straightforward. However, most road networks also contain road forks, see Fig. 3.4. Therefore, the road adjacency matrix must be constructed in order to ensure that the computed occupancies are over-approximative. We consider that a lane change is possible as long as there exists an intersection of lanes as shown in Fig. 3.5a. According to the definition of the adjacent lanelet matrix, one can only model whether or not lane crossings are possible along the entire length of a lanelet. Hence, to construct the adjacency matrix, we partition the lanelets accordingly. We first introduce the point p which represents the intersection of the corresponding lane boundaries of the road fork, as shown in Fig. 3.5a. If the final points of the outer bounds of lanelet$_{11}$ and lanelet$_{21}$ correspond to the point p, and lanelet$_{21}$ and lanelet$_{22}$ continue along the corresponding lanes as shown in Fig. 3.5a, all lanelets fulfill the constraint that they are either adjacent along their full length or not at all. The resulting adjacency matrix is presented in Fig. 3.5b. The adjacency makes it possible to define a lane:

Definition 3.2 (Lane)**.** A lane is defined as the union of lanelets that are longitudinally adjacent. □

In the following section, a model of the other traffic participants used to calculate the occupancy prediction is described.

3.3.2 Model of Other Traffic Participants

The mathematical model of other traffic participants, inspired by [6], is derived assuming the following constraints:

*C*1: when a parameterized speed $v_{\max}$ is reached ($v_{\max}$ could be set to a certain percentage above the official speed limit[1]), positive longitudinal acceleration is stopped;

[1] The speed limitation values may be different for specific types of vehicles, e.g., passenger car vs. truck.

$C2$: assuming that one can estimate the maximum engine power, that is characterized by a parameterized speed v_S, then the positive longitudinal acceleration is inversely proportional to speed above v_S;

$C3$: since the primary traffic scenario considered here is represented by driving on highways, we assume that driving backwards is not allowed;

$C4$: maximum absolute acceleration is bounded by $a_{\max}$;

$C5$: actions that cause a vehicle to cross the road/lane/sidewalk/crosswalk boundary are forbidden; however, unless various traffic regulations or lane markings forbid it, crossing lane boundaries is allowed.

While the constraints $C1$, $C3$, and $C5$ are derived from the traffic rules described in the Vienna Convention on Road Traffic [176], the others — $C2$ and $C4$ — represent physical constraints. If no speed limit exists, as on the German Autobahn, $C1$ can also be considered a physical constraint, provided by the maximum achievable speed of a vehicle. Formalization of traffic rules is a new research area [157,159], but out of the scope of this thesis. A different set of constraints may be considered when surrounding vehicles are automated and able to communicate their future plans to each other. In that situation, one only has to consider the uncertainty in following planned trajectories due to sensor noise, as demonstrated in [4], or uncertainty introduced by actuators. Based on the broadcasted high-level plan, we can adjust the adjacent lanelet matrix $A_{\mathcal{G}}$, accordingly. For example, if the considered traffic participant is expected to keep the current lane, then the adjacency edges corresponding to a lane change maneuver (both left and right) can be removed. Moreover, different sets of traffic regulations can be considered for each traffic participant — either by adding or removing traffic rules — for instance, depending on the vehicle type (e.g., if required, an emergency vehicle can violate the speed limit prescribed by the corresponding traffic sign). Nevertheless, removing some of the constraints does not affect the soundness of the verification procedure, but it does increase the uncertainty in the behavior of other traffic participants, that only leads to more conservative behavior of the ego vehicle. In the following, we model the dynamics of other traffic participants by a point mass (assuming that a rectangle encloses the body of the vehicle):

$$\ddot{s}_x(t) = a_x(t), \quad \ddot{s}_y(t) = a_y(t), \tag{3.1}$$

where $s_x(t),\ s_y(t)$ denote the position and $a_x(t),\ a_y(t)$ denote the acceleration in x and y coordinates, respectively. In order to restrict $a_x(t)$ and $a_y(t)$ according to the constraints $C1$-$C4$, unit vectors that point towards the longitudinal and lateral direction of the vehicle are introduced: $\Phi_{\text{long}}(t) = \frac{1}{v}[v_x(t),\ v_y(t)]^T$, $\Phi_{\text{lat}}(t) = \frac{1}{v}[-v_y(t),\ v_x(t)]^T$, where $v = \|[v_x, v_y]^T\|_2$ and $\|.\|_2$ denotes the Euclidean norm. Thus, we can define a_x and a_y as a function of the longitudinal acceleration $a_{\text{long}}(t)$ and the lateral acceleration $a_{\text{lat}}(t)$

as follows:

$$\begin{bmatrix} a_x \\ a_y \end{bmatrix} = \Phi_{\text{long}} a_{\text{long}} + \Phi_{\text{lat}} a_{\text{lat}}. \tag{3.2}$$

Considering the maximum tire friction potential, let us define a normalized steering input u_1, where $u_1 = 1$ refers to full steering to the left and $u_1 = -1$ refers to full steering to the right, from which results the lateral acceleration

$$a_{\text{lat}} = a_{\max} u_1. \tag{3.3}$$

According to Kamm's circle, we can limit the remaining acceleration potential in longitudinal direction, as follows:

$$a_{\text{c1,long}} = \sqrt{(a_{\max})^2 - (a_{\text{lat}})^2}, \tag{3.4}$$

which is a good approximation since the peak forces on the tires are almost identical for the longitudinal and the lateral directions, see e.g. [166, Fig. 14-16]. Thus, the maximum possible acceleration (constraint $C4$) is considered. Assuming that the vehicle has the engine power P and the vehicle mass m, one can derive the maximum longitudinal acceleration from $\frac{P}{m\,v} = a_{\max}\frac{v_S}{v}$, where $v_S = \frac{P}{a_{\max} m}$ is the speed above which the acceleration is limited by the engine power and no longer by the tire friction. However, since it is not trivial to estimate the v_S parameter, as a fallback solution one may set $v_S = \infty$, which provides an over-approximation for the occupancy set computation. Similar to the lateral acceleration, a normalized control input u_2 for the longitudinal acceleration is introduced, where $u_2 = \pm 1$ represents full braking and full acceleration within the acceleration potential. We therefore consider the restrictions to forward driving only, limited engine power, and the maximum speed (constraints $C1$-$C3$) by limiting the longitudinal acceleration as follows:

$$a_{\text{c2,long}} = \begin{cases} a_{\max}\frac{v_S}{v}, & v_S < v < v_{\max} \wedge u_2 > 0, \\ a_{\max}, & (0 < v \leq v_S \vee (v > v_S \wedge u_2 \leq 0)), \\ 0, & v \leq 0 \vee (v \geq v_{\max} \wedge u_2 \geq 0). \end{cases} \tag{3.5}$$

The longitudinal acceleration that satisfies the proposed constraints $C1$-$C4$ results by combining $a_{\text{c1,long}}$ and $a_{\text{c2,long}}$ (the constraint $C5$ for not leaving the road is considered later) is defined as:

$$a_{\text{long}} = \begin{cases} a_{\text{c2,long}}\, u_2, & a_{\text{c2,long}}\, |u_2| \leq a_{\text{c1,long}}, \\ a_{\text{c1,long}}\, \text{sgn}(u_2), & a_{\text{c2,long}}\, |u_2| > a_{\text{c1,long}}. \end{cases} \tag{3.6}$$

Likewise, to model *static obstacles*, e.g. stationary vehicles or just other objects, one can simply consider them as other traffic participants with zero velocity.

3.4 Occupancy Prediction

In this section, we present the occupancy prediction algorithm for the other traffic participants based on models of the road network and of other traffic participants, just described in Sec. 3.3. Apart from the main requirement of the prediction algorithm to generate provable correct results, another crucial specification is real-time capability. Efficient computation is required in order to realize frequent updates and to be able to quickly react to changing traffic situations — since the occupancy prediction is iteratively computed as described in Sec. 3.2.

Let us now introduce what we refer to as a model in this chapter.

Definition 3.3 (Model)**.** Given is a dynamical system $\dot{x} = f(x(t), u(t))$, where x is the state, u is the input, and t is the time. The possible initial states and the inputs are bounded by sets: $x(0) \in \mathcal{X}_0$, $\forall t : u(t) \in \mathcal{U}$. The model of this system is defined as the tuple (ordered set) $M = (f, \mathcal{X}_0, \mathcal{U})$. □

Given a model M, we define a reachable set as follows.

Definition 3.4 (Reachable set)**.** The reachable set of a model M (see Def. 3.3) at time $t = r$ is

$$\mathcal{R}(M, r) = \left\{ x(0) + \int_0^r f(x(t), u(t)) dt \middle| x(0) \in \mathcal{X}_0, \forall t : u(t) \in \mathcal{U} \right\}$$

and the reachable set of a time interval $t \in [0, r]$ is

$$\mathcal{R}(M, [0, r]) = \bigcup_{t \in [0,r]} \mathcal{R}(M, t).$$

□

Nevertheless, if we would consider the constraints $C1$-$C5$ of the vehicle dynamics as described in Sec. 3.3.2, we would have to model the dynamics of the vehicle using a combination of discrete and continuous dynamics, which is also referred to as a hybrid system [14]. For instance, if the vehicle reaches the maximum velocity $v_{\max}$, the dynamics must change to constant velocity. However, the computation of the reachable sets of hybrid systems is too time consuming for our application. Therefore, we use abstractions of the original model M instead.

Definition 3.5 (Abstraction)**.** Given is a model M of a dynamical system. The model M_i is an abstraction of M if the reachable set of the abstraction contains the reachable set of the model, i.e., $\forall t > 0 : \mathcal{R}(M, t) \subseteq \mathcal{R}(M_i, t)$. □

When choosing an abstract model, two of the most important requirements to be satisfied are the following: preserving the essential behavior of the original model and permitting computationally more efficient methods for reachability analysis. Since we are ultimately only interested in the occupancy, which is determined by the position and orientation of other traffic participants, we aim to find abstractions that are tight with respect to these variables. Given a state vector $x \in \mathbb{R}^n$, where the first two elements x_1, x_2 are the x-position and the y-position, and the third element x_3 is the orientation, we denote $\text{proj}(x) = [x_1, x_2, x_3]^T$ as the operator that projects the state vector onto position and orientation. Further, $\text{proj}\big(\mathcal{R}(M, t)\big) := \{\text{proj}(x) | x \in \mathcal{R}(M, t)\}$ returns the set of possible positions and orientations.

We exploit the fact that in the end, we are only interested in the position and orientation using the following proposition taken from [6, Prop. 5.1]:

Proposition 3.1 (Over-approximative occupancy)**.** Given are models M_i, $i = 1, \dots, m$ which are abstractions of model M_0, i.e., $\forall t > 0 : \mathcal{R}(M_0, t) \subseteq \mathcal{R}(M_i, t)$. The occupancy of the model M_0 can be over-approximated by

$$\forall t > 0 : \mathrm{proj}\big(\mathcal{R}(M_0, t)\big) \subseteq \bigcap_{i=1}^{m} \mathrm{proj}\big(\mathcal{R}(M_i, t)\big).$$

□

Using the above model abstraction for the computation of reachable sets facilitates considerably faster over-approximative results, while still providing accuracy. There are two main reasons why the abstractions are more rapidly computed: first, they do not evaluate the complete set of states, and since the computational complexity of reachability analysis is superlinear in the number of state variables, this is a major advantage. Secondly, the constraints $C1$-$C5$ are considered by performing an intersection of the results of abstractions that are purely continuous, instead of considering hybrid dynamics. We can obtain a more precise occupancy prediction by having a larger number of computed abstract models which comes along with the disadvantage of increased computation time. We can, therefore, tune a trade-off between accuracy and computational time depending on the requirements. For example, for the traffic participants driving in close proximity to the ego vehicle, a more accurate prediction may be required, such that the ego vehicle can plan its trajectory. For the remainder of the surrounding traffic participants that are driving a farther distance away, however, such a precise prediction may be unnecessary due to the uncertainty. Note that there is a potential to improve the computation time by using multi-core hardware architectures, where each abstract model can run in parallel in its own processing thread.

The occupancy sets are calculated for consecutive time intervals $\tau_k := [t_k, t_{k+1}]$ for a user-specified step size $r = t_{k+1} - t_k$ and user-specified time horizon t_h. In order to ensure that collision detection is not missed, we calculate the occupancy prediction for consecutive time intervals instead of consecutive points in time. In this thesis, two abstract models for computing the occupancy of each traffic participant are considered. The first abstraction is denoted by M_1, and it considers the constraints $C3$ and $C4$ (see Sec. 3.3.2). The predicted occupancy using abstraction M_1 is denoted by $\mathcal{O}_1(t) \supseteq \mathrm{proj}\big(\mathcal{R}(M_1, t)\big)$ and is so-called *acceleration-based occupancy*. The second abstraction M_2 considers the constraints $C1, C2$, and $C4$ in longitudinal direction. Its corresponding occupancy is denoted by $\mathcal{O}_2(t)$, and we refer to it as *lane-following occupancy*. The last constraint $C5$ is considered by intersecting the drivable area, delimited by the road boundaries $\mathcal{O}_{\mathrm{road}}$, with the occupancy sets resulting from considering the other two abstraction M_1 and M_2. As a result, to consider all proposed constraints, the *overall occupancy* $\mathcal{O}(t)$ is computed as follows:

$$\mathcal{O}(t) = \mathcal{O}_1(t) \cap \mathcal{O}_2(t) \cap \mathcal{O}_{\mathrm{road}}. \tag{3.7}$$

The next step is to decide on a set representation for the overall occupancy $\mathcal{O}(t)$. The recurrent operations that must be performed at each time step are represented by collision checks on the occupied regions of the other traffic participants with the corresponding occupancy of the ego vehicle. To efficiently perform those collision checks and to exactly compute the intersections in (3.7), we require a set representation that has the following properties: (*i*) is closed under intersection, (*ii*) is able to represent non-convex sets, and (*iii*) provides efficient algorithms for collision detection. Therefore, we choose polygons, as illustrated in Fig. 3.6, since other common set representations (ellipsoids, boxes, zonotopes, polyhedra, etc.) are all convex.

Definition 3.6 (Polygon)**.** Given is a tuple $(v_1, \dots, v_p)$ of p vertices $v_i \in \mathbb{R}^2$. A polygon $P(v_1, \dots, v_p)$ is a set in the plane bounded by a border that consists of straight lines between neighboring vertices v_i and v_{i+1}, except for the last vertex v_p which is connected to the first one v_1, see e.g. Fig. 3.8b. □

In order to satisfy real-time computation constraints, a fast algorithm is required for identifying the intersections of polygons as specified in (3.7). Many algorithms already exist that compute the intersection of two polygons [1]. However, considering the strict time requirements, we choose the *Greiner Hormann Polygon Clipping Algorithm* since it can efficiently deal with non-convex polygons [63].

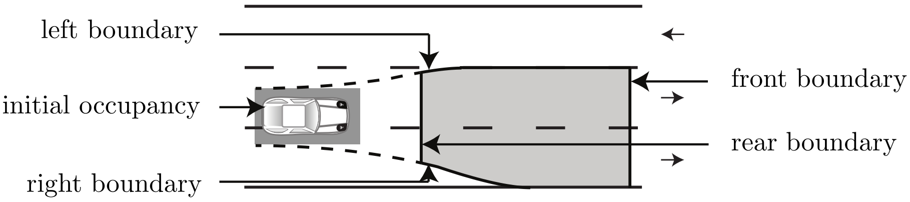


Figure 3.6: Initial occupancy and boundaries of the predicted occupancy set.

In order to compute the predicted occupancy $\mathcal{O}_1(t)$ and $\mathcal{O}_2(t)$ in (3.7), we first define a set of parameters that characterize each vehicle. The considered parameters are listed in Table 3.1. Due to the fact that the prediction of the occupancy sets is performed in an over-approximative manner, all possible lanelets that a vehicle can follow have to be considered. To identify the reachable lanelets, we search the adjacency graph of the road network (see e.g. road fork in Fig. 3.5b). In order to improve computational performance, the algorithm can be parallelized for every vehicle. Next, the occupancy prediction for each considered abstract model is described, where acceleration-based occupancy (abstraction M_1) is presented in Sec. 3.4.1 and lane-following occupancy (abstraction M_2) is presented in Sec. 3.4.2.

3.4.1 Acceleration-Based Occupancy (Abstraction M_1)

The first abstraction of vehicle dynamics considers that absolute acceleration is limited ($C4$) and that driving backwards is not allowed ($C3$). The constraint $C4$ implies that the over-approximated occupancy at time t can be described by a circle with center $c(t)$ and radius $r(t)$ (see [164]) as shown in Fig. 3.7,

Table 3.1: Vehicle parameters used for computation of occupancy sets.

parameter	**variable [unit]**
max. acceleration	a_{max} [m/s^2]
max. velocity	v_{max} [m/s]
switching velocity	v_S [m/s]
vehicle width	$\tilde{w}$ [m]
vehicle length	$\tilde{l}$ [m]

when constraint $C3$ (driving backwards) is not yet considered:

$$c(t) = \begin{bmatrix} s_x(0) \\ s_y(0) \end{bmatrix} + \begin{bmatrix} v_x(0) \\ v_y(0) \end{bmatrix} t, \quad r(t) = \frac{1}{2} a_{max} t^2. \tag{3.8}$$

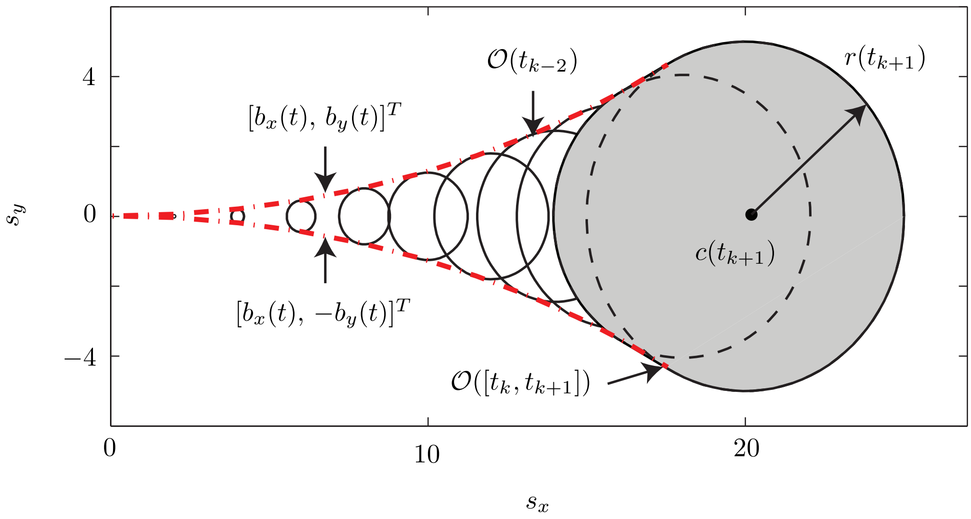


Figure 3.7: Acceleration-based occupancy sets.

Without loss of generality, assuming that $s_x = 0, s_y = 0, v_x = v$ and $v_y = 0$, the boundary of the acceleration-based occupancy is described by a two-dimensional function $[b_x(t), b_y(t)]^T$, where

$$b_x(t) = v_0 t - \frac{a_{max}^2 t^3}{2v_0}, b_y(t) = \sqrt{\frac{1}{4} a_{max}^2 t^4 - \left(\frac{a_{max}^2 t^3}{2v_0}\right)^2}, \tag{3.9}$$

as shown in [6]. To prevent driving backwards, the solution in x-direction is to set the maximum value of $b_{x_{max}} := b_x(t_{max})$ for $t \geq t_{max}$. The value $t_{max} = \sqrt{2/3}\frac{v_0}{a_{max}}$ is found by solving for $\dot{b}_x(t) = 0$. The outer bound of the occupancy $\mathcal{O}_1(\tau_k)$ for a given time interval $\tau_k = [t_k, t_{k+1}]$ is obtained by using the following lemma:

Lemma 1 ($\mathcal{O}_1(\tau_k)$ without vehicle dimensions)**.** The occupancy $\mathcal{O}_1(\tau_k)$ without considering vehicle dimensions is over-approximated by a polygon $P(q_1, \dots, q_6)$, where

$$\begin{aligned}
q_1 &= [c_x(t_k) - r(t_k),\, c_y(t_k) + r(t_k)]^T,\\
q_2 &= [b_x(t_k),\, c_y(t_{k+1}) + r(t_{k+1})]^T,\\
q_3 &= [c_x(t_{k+1}) + r(t_{k+1}),\, c_y(t_{k+1}) + r(t_{k+1})]^T,\\
q_4 &= [c_x(t_{k+1}) + r(t_{k+1}),\, c_y(t_{k+1}) - r(t_{k+1})]^T,\\
q_5 &= [b_x(t_k),\, c_y(t_{k+1}) - r(t_{k+1})]^T,\\
q_6 &= [c_x(t_k) - r(t_k),\, c_y(t_k) - r(t_k)]^T.
\end{aligned}$$

□

Proof. The resulting occupancy is over-approximate by the convex hull of the occupancy $\mathcal{O}(t_k)$ and $\mathcal{O}(t_{k+1})$, as illustrated in Fig. 3.8a. This is an over-approximation since the boundary $[b_x(t), b_y(t)]^T$ form (3.9) is concave. The points where $[b_x(t), b_y(t)]^T$ and $[b_x(t), -b_y(t)]^T$ intersect with $\mathcal{O}_1(t_k)$ and $\mathcal{O}_1(t_{k+1})$ are denoted by $\hat{q}_1$ - $\hat{q}_4$ (see Fig. 3.8a).
The exact boundary $\mathcal{O}_1(\tau_k)$ is over-approximated by the vertices q_1-q_6 and $\hat{q}_1$ - $\hat{q}_4$ (see Fig. 3.8b). Finally, we over-approximate the result in Fig. 3.8b by its convex hull, such that $\hat{q}_1$ - $\hat{q}_4$ are removed (see Fig. 3.8c) resulting in the vertices $q_1, \dots, q_6$ of the occupancy polygon. ■

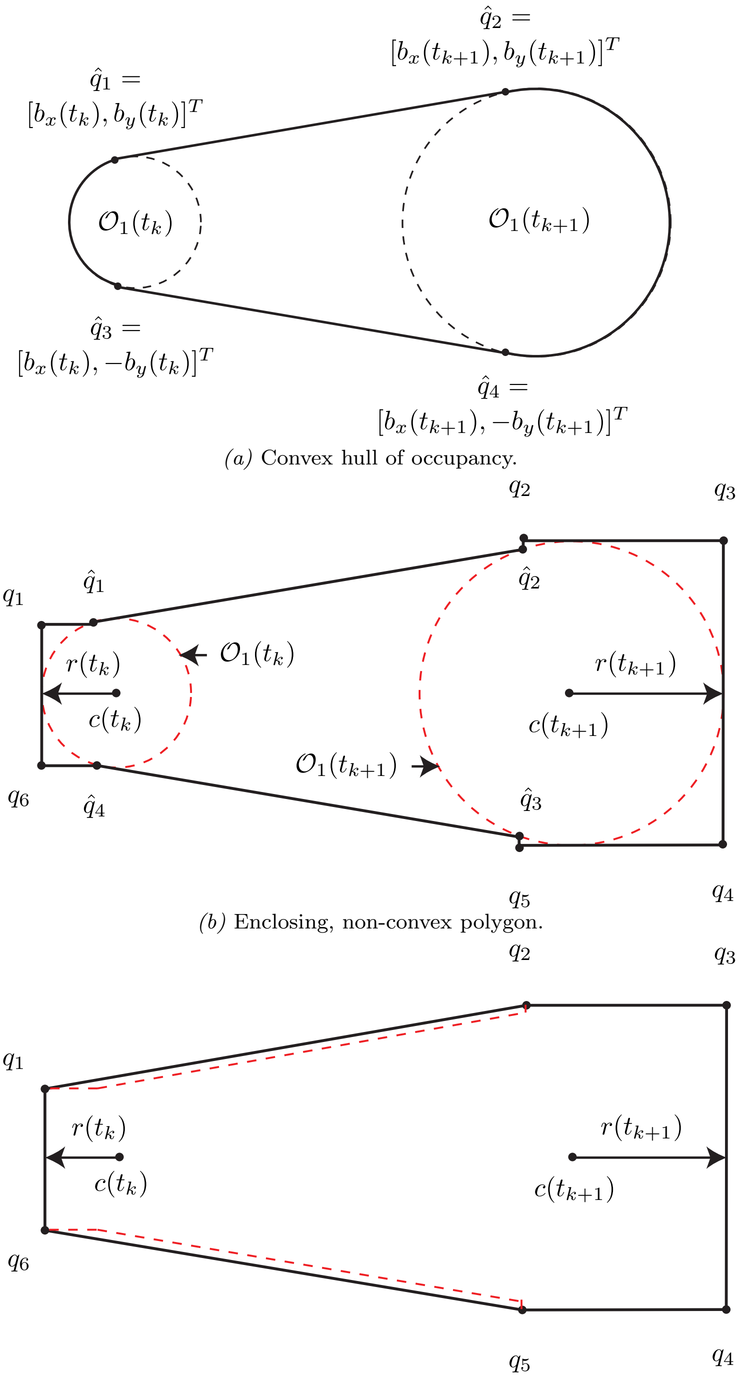
In order to reduce memory consumption and computation time for collision checks, we remove $\hat{q}_1$ - $\hat{q}_4$ such that we obtain a polygon with fewer vertices at the cost of a negligible over-approximation, as illustrated in Fig. 3.8c. Until now, we have assumed that each vehicle is a point mass without considering its dimensions. To include this in the occupancy prediction, we enclose the body of a vehicle including uncertainty in its initial position, in a rectangle with length $\tilde{l}$ and width $\tilde{w}$ and whose reference point is the centroid. After adding measurement uncertainties, we enclose the occupancy of the vehicle in a larger rectangle of length l and width w as depicted in Fig. 3.9. Assuming that each point of the vehicle has the acceleration bound a_{max} and after introducing the notation $M(\mathcal{X}^*)$ for a model whose set of initial states is $\mathcal{X}_0 = \mathcal{X}^*$, we can formulate that

$$\mathcal{R}(M(\mathcal{X}_0), t) = \bigcup_{x_0 \in \mathcal{X}_0} \mathcal{R}(M(x_0), t), \tag{3.10}$$

which directly follows from the definition of reachable sets in Def. 3.4. Using (3.10), we can derive the over-approximative occupancy of a vehicle, based on an initial occupancy set, as follows:

Theorem 3.1 ($\mathcal{O}_1(\tau_k)$ with vehicle dimensions)**.** The exact occupancy $\mathcal{O}_1(\tau_k)$, when $\mathcal{X}_0$ is a rectangle of length l and width w, is over-approximated by a polygon $P(p_1, \dots, p_6)$ using the vertices q_1-q_6 from Lemma 1, where

$$\begin{aligned}
p_1 &= q_1 + [-0.5l,\, 0.5w]^T, & p_4 &= q_4 + [0.5l,\, -0.5w]^T,\\
p_2 &= q_2 + [-0.5l,\, 0.5w]^T, & p_5 &= q_5 + [-0.5l,\, -0.5w]^T,\\
p_3 &= q_3 + [0.5l,\, 0.5w]^T, & p_6 &= q_6 + [-0.5l,\, -0.5w]^T.
\end{aligned}$$



(a) Convex hull of occupancy.

(b) Enclosing, non-convex polygon.

(c) Enclosing, convex polygon.

Figure 3.8: Computation steps to obtain convex, over-approximative occupancy.

□

Proof. The proof follows directly from Lemma 1 and (3.10). Fig. 3.9 illustrates the resulting vertices from the theorem. ■

Due to position and orientation invariance, we have assumed without loss of generality, a specific relative

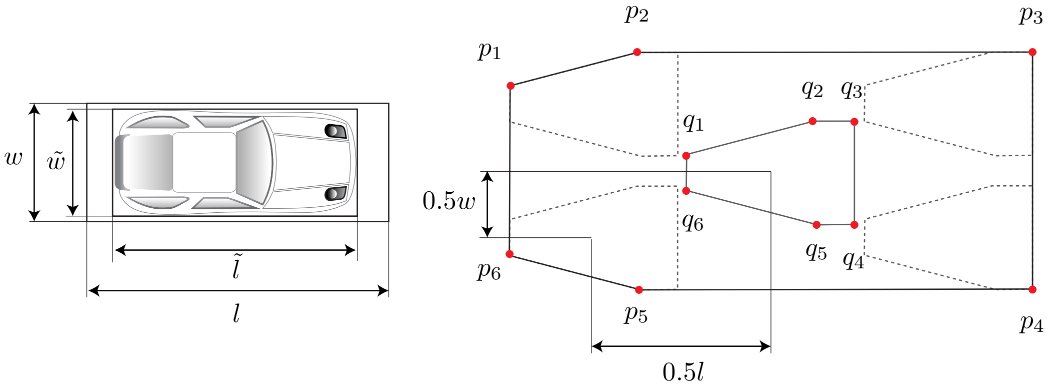


Figure 3.9: Occupancy polygon where the vehicle dimensions are considered. The points q_1-q_6 are taken from Fig. 3.8.

position and orientation of the initial set. However, to obtain the occupancy for an arbitrary situation, the calculated occupancy must be translated and rotated according to the initial position and orientation of the vehicle.

3.4.2 Lane-Following Occupancy (Abstraction M_2)

So far, only a limited absolute acceleration of other traffic participants has been considered. Nevertheless, as described by the constraints $C1$ and $C2$, a vehicle cannot always accelerate with a_{max} in the longitudinal direction (i.e., driving direction) to consider maximum velocity ($C1$) and maximum engine power ($C2$). Constraints $C1 - C2$ are evaluated on partial paths along inner lane boundaries, which increases the efficiency. After finding the maximum position along these partial paths, we assume that the vehicle location can be anywhere in lateral direction (i.e., perpendicular to the path) within the lane boundaries. Let us introduce another constraint $C4^*$ — a variation of $C4$ — such that the limited maximum acceleration can be enforced not only in longitudinal, but also in lateral direction in abstraction M_2. Therefore, the traveled distance along a path is independent of its shape when considering the constraints $C1$, $C2$, and $C4^*$. This leads to:

Proposition 3.2 (Traveled distance along path for M_2)**.** Since we do not consider lateral acceleration in constraints $C1$, $C2$, and $C4^*$, the maximum velocity is not influenced by the shape of the followed path.

□

Nevertheless, the lateral acceleration is still considered, but only in abstraction M_1, according to the Kamm's circle. The calculation of the front position along a route, denoted by $\xi_f(t)$ (f: front), is applied to model M_2 as described in [6]. We first introduce the interval of initial path coordinates $[\underline{\xi}_0, \overline{\xi}_0]$ and velocities $[\underline{v}_0, \overline{v}_0]$. Then, we compute the front position along a path $\xi_f(t)$ by simulating the vehicle model described in Sec. 3.3.2 with maximum acceleration $u_2 = 1$ from $\xi_f(0) = \overline{\xi}_0$ and $v(0) = \overline{v}_0$. Due to the monotonicity of the longitudinal dynamics, only a single simulation is required. However, since infinitely many paths exist, even if a single lanelet is followed, the question of how to select the shortest path is raised. This matter was not considered in the work described in [6], where it was assumed that the shortest path is represented by the center path along a lane, which does not result in an over-approximation since one can cut corners. In the following, we present an approach that is able to abstract the behavior of the other traffic participants in a way that satisfies constraints $C1$, $C2$, and $C4^*$ regardless of the selected path through the lanelet's sequence. To this end, we propose an algorithm that efficiently calculates a lower bound of the shortest path through a lanelets sequence, derived from a given road network.

Next, let us formalize the problem of determining the shortest path along a lane. We can assume, without loss of generality, that a vehicle can be represented by a point if the distances between lane boundaries are reduced such that the dimensions of the vehicle are considered. Note that, if this step would not be performed (to increase calculation speed), the soundness of the proposed approach would not be affected, i.e., the resulting shortest path would be even shorter compared to the exact solution. The resulting solution is still overapproximative, but at a cost of being more conservative.

Problem 1 (Shortest path through a lane)**.** Given is an arbitrary lane as depicted in Fig. 3.11. Let us denote the start border by $b_{start}(s_x, s_y)$ and the end border by $b_{end}(s_x, s_y)$. A point $[s_x, s_y]^T$ is on the start line for $b_{start}(s_x, s_y) = 0$ and $b_{start}(s_x, s_y) < 0$ when a point is inside the lane segment. This is analogous for $b_{end}(s_x, s_y)$. The lane boundaries are formulated as inequality constraints, where one is on the right hand side from the left boundary in driving direction if $b_{left}(s_x, s_y) < 0$ and analogously for $b_{right}(s_x, s_y) < 0$. We can then formulate the area of the lane segment as the set

$$\begin{aligned}\mathcal{O}_{segment} =& \{[s_x, s_y]^T | b_{start}(s_x, s_y) < 0, b_{end}(s_x, s_y) < 0, \\ & b_{left}(s_x, s_y) < 0, b_{right}(s_x, s_y) < 0\}.\end{aligned} \tag{3.11}$$

Next, we introduce a path variable ξ such that the x-position and y-position are functions of it: $s_x(\xi)$, $s_y(\xi)$. The path variable ξ can result from $s_x(\xi)$, $s_y(\xi)$ as

$$\begin{aligned}\xi &= \int_0^{\xi} \sqrt{s_x'^2(\hat{\xi}) + s_y'^2(\hat{\xi})}\, \mathrm{d}\,\hat{\xi}, \\ s_x'(\hat{\xi}) &= \frac{\mathrm{d}\, s_x(\xi)}{\mathrm{d}\,\xi}\Big|_{\xi=\hat{\xi}}, \quad s_y'(\xi) = \frac{\mathrm{d}\, s_y(\xi)}{\mathrm{d}\,\xi}\Big|_{\xi=\hat{\xi}}.\end{aligned} \tag{3.12}$$

Finally, the problem of finding the shortest path from b_{start} to b_{end} while remaining within the road boundaries can be formulated as:

$$\min_{s_x(\xi), s_y(\xi)} J(s_x(\xi), s_y(\xi)) = \int_0^{\xi_f} \sqrt{s_x'^2(\hat{\xi}) + s_y'^2(\hat{\xi})} \mathrm{d}\,\hat{\xi}$$

under the constraints

$$\begin{aligned} b_{start}(s_x(0), s_y(0)) &= 0, \\ b_{end}(s_x(\xi_f), s_y(\xi_f)) &= 0, \\ b_{left}(s_x(\xi), s_y(\xi)) &< 0, \\ b_{right}(s_x(\xi), s_y(\xi)) &< 0. \end{aligned} \tag{3.13}$$

□

Taking into consideration that our application should run in a real-time environment, we have to find a solution to Problem 1 that adheres to the required time performance. However, solving Problem 1 is time consuming, even when considering only polygon obstacles [57, 70, 78]. Therefore, instead of providing the exact solution to Problem 1, we compute an under-approximation of the shortest path. To this end, we introduce the notion of corresponding paths:

Definition 3.7 (Corresponding path)**.** We define the corresponding path $h(\xi)$ with path variable ξ of the border b as

$$b(s_x(\xi), s_y(\xi)) = 0 \quad \Rightarrow \quad h(\xi) = [s_x, s_y]^T.$$

The other direction of the implication $h(\xi) = [s_x, s_y]^T \Rightarrow b(s_x(\xi), s_y(\xi)) = 0$ is not unique, and we only require that a possible solution is provided such that the set $\{[s_x, s_y]^T | b(s_x, s_y) < 0,\ s_x, s_y \in \mathbb{R}\}$ refers to the inner part behind the border. □

The path $h(\xi)$ either refers to the left or right border of a lane, depending on context. When using $h_{right}(\xi)$, we explicitly refer to the right border and analogously for $h_{left}(\xi)$. We denote the well-known signed curvature by

$$\kappa(\xi) = f_\kappa(h(\xi)) = \frac{s'_x(\xi)s''_y(\xi) - s''_x(\xi)s'_y(\xi)}{(s'_x(\xi)^2 + s'_y(\xi)^2)^{3/2}}.$$

In the following, two cases are distinguished when we under-approximate the length of the shortest path, depending on whether the inner border of the lane has an inflection point or not. Let us start with the case where there is no inflection point in the inner boundary of the lane.

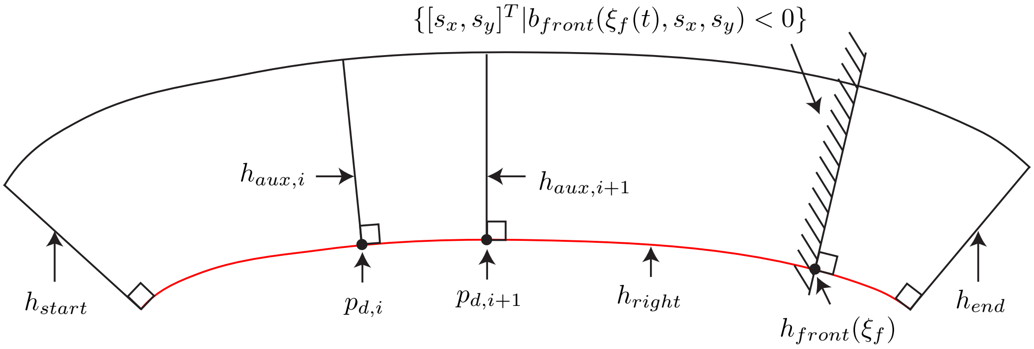


Figure 3.10: Shortest path through a lane without inner inflection point.

Lemma 2 (Shortest path without inner inflection point)**.** Given is a lane segment as described in Problem 1, where the inner lane bound does not change the sign of the curvature, i.e., $\nexists \xi : \kappa_{right}(\xi) := f_\kappa(h_{right}(\xi)) = 0$. Without loss of generality, we consider a right curve so that the inner bound is h_{right}. We further require that h_{start}, h_{end} are straight lines perpendicular to h_{right}. This can be formalized by introducing the normal vectors $n_{start}, n_{end} \in \mathbb{R}^2$ and the distance values $d_{start}, d_{end} \in \mathbb{R}$:

$$\begin{aligned}
&\forall \xi \in [0, \xi_f] :\\
&\kappa_{right}(\xi) < 0 \text{ (right turn)},\\
&\text{constraints in (3.13) (lane borders)},\\
&b_{start}(s_x, s_y) = n_{start}^T [s_x, s_y]^T - d_{start} \text{ (start line)},\\
&b_{end}(s_x, s_y) = n_{end}^T [s_x, s_y]^T - d_{end} \text{ (finish line)},\\
&\frac{|n_{start}^T h'_{right}(0)|}{\|n_{start}\|_2 \|h'_{right}(0)\|_2} = 1 \text{ (b_{start} perpendicular to h_{right})},\\
&\frac{|n_{end}^T h'_{right}(0)|}{\|n_{end}\|_2 \|h'_{right}(0)\|_2} = 1 \text{ (b_{end} perpendicular to h_{right})}.
\end{aligned} \tag{3.14}$$

The shortest path corresponding to Problem 1 is $[s_x^*(\xi), s_y^*(\xi)]^T = h_{right}(\xi)$ for $\xi \in [0, \xi_f]$. □

Proof. We divide the problem of the lemma into subproblems by introducing auxiliary line segments $h_{aux,i}(\alpha) = p_i + \alpha l_i$, $\alpha \in [0, 1]$, $p_i, l_i \in \mathbb{R}^2$, with the corresponding border that is denoted by $b_{aux,i}$ (see Def. 3.7). In this way, we obtain the same sub-problems, except that b_{start} is replaced by $b_{aux,i}$ and b_{end} is replaced by $b_{aux,i+1}$. Next, we calculate the shortest distance between two auxiliary line segments. Let $p_{d,i}$ and $p_{d,i+1}$ be the start and end points of the shortest line connecting both line segments $h_{aux,i}$ and $h_{aux,i+1}$ (see Fig. 3.10).

There exist four cases for the shortest distance between two line segments [117]:

(A) $p_{d,i}$ and $p_{d,i+1}$ are within $h_{aux,i}$ and $h_{aux,i+1}$,

(B) $p_{d,i}$ is within $h_{aux,i}$ and $p_{d,i+1}$ is one of the end points of $h_{aux,i+1}$,

(C) opposite of case (B),

(D) $p_{d,i}$ and $p_{d,i+1}$ are end points of $h_{aux,i}$ and $h_{aux,i+1}$.

In a two-dimensional setting, case (A) only occurs when the line segments intersect, which can be ruled out (see Fig. 3.10). Cases (B) and (C) result in longer solutions compared to case (D); this can be easily seen, since either the movement of $p_{d,i}$ along $h_{aux,i}$ or the movement of $p_{d,i+1}$ along $h_{aux,i+1}$ increases the distance.

When the number N of auxiliary straight lines $h_{aux,i}$ approach infinity, the union $\bigcup_i^N \{p_{d,i} + \gamma(p_{d,i+1} - p_{d,i}) | \gamma \in [0, 1]\}$ of the shortest line segments approaches the set of points on $h_{right}(\xi)$. ■

Until now, we have shown how to calculate the shortest path along a lane segment without an inner inflection point. Next, we use this result to compute an over-approximative occupancy through such a lane segment.

Lemma 3 (Occupancy without inner inflection point)**.** Given is a lane segment as formalized in (3.14), where the inner lane bound does not have an inflection point. We denote by $\xi_f(t)$ the front position along the inner path under constraints $C1$, $C2$, and $C4^*$, and by $b_{front}(\xi_f(t), s_x, s_y) = n_f^T(\xi_f(t))\big[[s_x, s_y]^T - h_{right}(\xi_f(t))\big]$ the front border (see Fig. 3.10), which is perpendicular to the inner bound (here: h_{right}) so that n_f is aligned with the inner bound: $|n_f^T(\xi) h'_{right}(\xi)| / (\|n_f(\xi)\|_2 \|h_{right}(\xi)\|_2) = 1$. Then, by using

$\mathcal{O}_{segment}$ from (3.11), the over-approximative occupancy under constraints $C1$, $C2$, and $C4^*$ is

$$\mathcal{O}_2(t) = \{[s_x, s_y]^T | b_{front}(\xi_f(t), s_x, s_y) < 0\} \cap \mathcal{O}_{segment}$$

assuming that the segment describes a turn of less than 90°. If this is not the case, one can split the segment into smaller segments without loss of generality. □

Proof. The fastest path for moving from one boundary of a lane to the other boundary is determined by the shortest path. This is true due to the invariance of the velocity with respect to the shape of a followed path, as shown in Prop. 3.2. Lemma 2 has shown that the shortest path within a lane without an inflection point is represented by the inner border itself. Therefore, it suffices to compute the front bound along the path of the inner border and perpendicular to it. The proof is concluded by applying Lemma 2, which indicates there is no other shorter path between b_{start} and b_{front}. ■

So far, the occupancy of a vehicle along a lane without an inflection point has been considered. To over-approximate the occupancy along an arbitrary lane, we introduce our inflection-point segmentation.

Definition 3.8 (Inflection-point segmentation)**.** We algorithmically define the inflection-point segmentation of a lane, with its result illustrated in Fig. 3.11:

(i) If the curvature of the inner left boundary is positive $\kappa_{left} > 0$, start at $[\tilde{s}_x, \tilde{s}_y]^T$, where $b_{start}(\tilde{s}_x, \tilde{s}_y) = 0$, $b_{left}(\tilde{s}_x, \tilde{s}_y) = 0$, and follow h_{left}. Otherwise (for the inner right boundary), start at $[\hat{s}_x, \hat{s}_y]^T$, where $b_{start}(\hat{s}_x, \hat{s}_y) = 0$, $b_{right}(\hat{s}_x, \hat{s}_y) = 0$, and follow h_{right}. If initially the left and right boundaries of the lane are both concave, according to this algorithm, the left side is the default one.

(ii) We denote an inflection point by γ_i, which is defined by a sign change of κ_{left} or κ_{right}, depending on the border that is being currently followed (see Fig. 3.11). We then follow the current lane boundary until an inflection point γ_i is reached. Next, we construct a line segment $h_{cross,i}(\alpha) = \gamma_i + \alpha g_i$, $\gamma_i, g_i \in \mathbb{R}^2$, $\alpha \in [0, 1]$, from one bound to the other, that is perpendicular to the bound belonging to γ_i. For the change from the left to the right bound, we have $g_i^T h'_{left} = 0$, $h_{cross,i}(0) = h_{left}(\gamma_{i,x}, \gamma_{i,y})$, $h_{cross,i}(1) = h_{right}(\mu_{i,x}, \mu_{i,y})$, where μ_i is the point from which one continues on the other bound (see Fig. 3.11). We analogously perform this procedure for the change from the right to the left bound.

(iii) Additional line segments $\tilde{h}_{cross,i}(\alpha) = \mu_i + \alpha \tilde{g}_i$, $\alpha \in [0, 1]$ that are drawn to intersect the previously followed border (see Fig. 3.11) are required in order to obtain a proper segmentation. In contrast to $h_{cross,i}(\alpha)$ that is perpendicular to the current lane boundary, $\tilde{h}_{cross,i}(\alpha)$ is perpendicular to the opposite lane bound ($\tilde{g}_i^T h'_{right} = 0$).

(iv) Next, we define two types of regions R_i and IR_i, as illustrated in Fig. 3.11: Given the corresponding bounds $b_{cross,i}$ and $\tilde{b}_{cross,i}$ of the line segments $h_{cross,i}$ and $\tilde{h}_{cross,i}$, the regions $R_i = \{[s_x, s_y]^T | b_{cross,i}(s_x, s_y) < 0, \tilde{b}_{cross,i-1}(s_x, s_y) < 0\} \cap \mathcal{O}_{segment}$ without the inner inflection point and intermediate regions $IR_i = \{[s_x, s_y]^T | \tilde{b}_{cross,i}(s_x, s_y) < 0, b_{cross,i}(s_x, s_y) < 0\} \cap \mathcal{O}_{segment}$, where one border only consists of the single point μ_i.

(v) The bound to which μ_i belongs is followed until another inflection point is reached and the same procedure is repeated until the final border is reached ($b_{end}(s_x, s_y) = 0$).

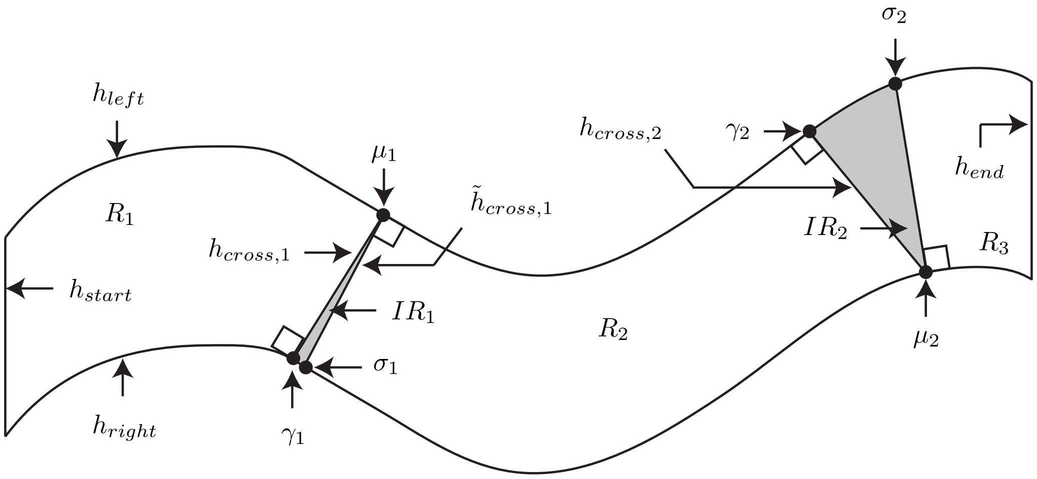


Figure 3.11: Inflection point segmentation of a lane.

□

Let us introduce the j^{th} inner paths $h_{in,j}(\xi)$ of an inflection-point segmentation, where $\xi \in [0, \xi_{max,j}[$, $h_{in,j}(0) = \mu_{i-1}$, and $\xi_{max,j}$ is defined as the value that ensures $h_{in,j}(\xi_{max,j}) = \gamma_i$. The inner path $h_{in,j}(\xi)$ is perpendicular to $\tilde{h}_{cross,j-1}$ at $\xi = 0$ and perpendicular to $h_{cross,j}$ at $\xi = \xi_{max,j}$. However, this is not necessarily ensured for the first and last partial inner path $h_{in,1}$ and $h_{in,e}$, touching h_{start} and h_{end}, respectively. Therefore, we construct the auxiliary first and last inner bounds $h_{in,init}(\xi)$ and $h_{in,final}(\xi)$ as shown in Fig. 3.12, which are perpendicular to h_{start}, h_{end} and touch $h_{in,1}(\xi)$, $h_{in,e}(\xi)$, such that orthogonality is ensured in those cases, as well.

The partially-connected aggregated path accumulated from the partial inner paths $h_{in,j}(\xi)$ is then defined as follows:

$$h(\xi) = \begin{cases} h_{in,init}(\xi), & \xi \in [0, \tilde{\xi}_0[, \\ h_{in,1}(\xi), & \xi \in [\tilde{\xi}_0, \tilde{\xi}_1[, \\ h_{in,2}(\xi - \tilde{\xi}_1), & \xi \in [\tilde{\xi}_1, \tilde{\xi}_2[, \\ \vdots & \\ h_{in,final}(\xi - \tilde{\xi}_{e-1}), & \xi \in [\tilde{\xi}_{e-1}, \tilde{\xi}_e]. \end{cases} \tag{3.15}$$

Next, given the front path variable ξ_f, one must evaluate the aggregate path (3.15) to obtain $[s_{x,f}, s_{y,f}]^T = h(\xi_f)$. The front bound is constructed as described in Lemma 3.

Theorem 3.2 (Occupancy along lane). Let us denote by $j_\xi(t)$ the index of the region corresponding to $\xi(t)$, i.e., $\xi(t) \in [\tilde{\xi}_{j-1}, \tilde{\xi}_j[$ in (3.15). Next, we introduce the occupancy within the j_ξ^{th} inflection-point region obtained from Lemma 3 as $\mathcal{O}_{2,j_\xi}(t)$. Thus,

$$\mathcal{O}_2(t) = (\bigcup_{j=0}^{j_\xi(t)-1} R_j) \cup (\bigcup_{j=0}^{j_\xi(t)-1} IR_j) \cup \mathcal{O}_{2,j_\xi}(t).$$

□

Proof. The result follows directly from Fig. 3.11, Fig. 3.12, and Lemma 3. ■

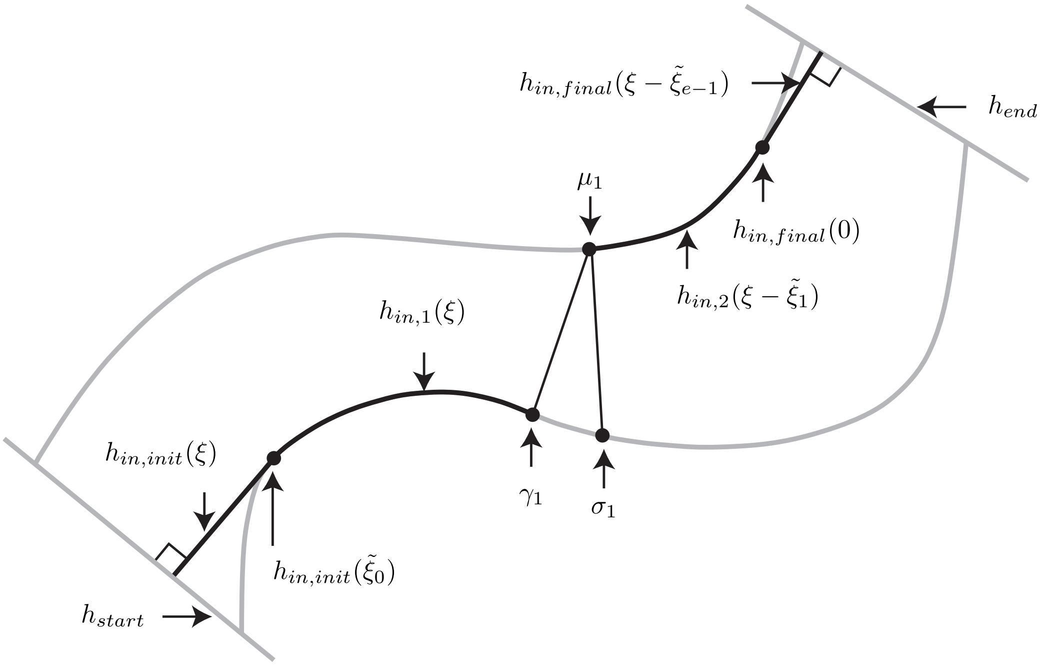

$h_{in,final}(\xi - \tilde{\xi}_{e-1})$
h_{end}
μ_1
$h_{in,final}(0)$
$h_{in,2}(\xi - \tilde{\xi}_1)$
$h_{in,1}(\xi)$
$h_{in,init}(\xi)$
γ_1
σ_1
$h_{in,init}(\tilde{\xi}_0)$
h_{start}

Figure 3.12: Inner paths of a lane segment.

Recall that in order to consider all future possible behaviors of a vehicle, we not only compute the occupancy for the current lane but also for surrounding lanes if a lane change is possible according to the road adjacency graph.

3.5 Numerical Experiments

In this section, we demonstrate our proposed approach for different multi-lane road networks considering road forks, as well. Moreover, the predicted occupancy sets computed using our proposed methodology are compared to results obtained from a high-fidelity model. The behaviors of the high-fidelity model are obtained by using rapidly-exploring random trees (RRTs) [108]. Real-world measurements from US highway 101, i.e., a dataset that is part of the Federal Highway Administration's (FHWA) Next Generation Simulation (NGSIM)[1] project, are used to validate our approach. For this purpose, we check whether all the recorded data is enclosed by our set-based occupancy prediction for each traffic participant, for each corresponding time instance.

The results were obtained on a machine with 2.2 GHz Intel Core i7 processor and 16 GB 1600 MHz DDR3 memory. The required vehicle parameters are listed in Table 3.1. For simplicity, we use the same parameters for all predictions: $a_{max} = 10$ m/s^2 (obtained from friction coefficient $\mu = 1.02$ and

[1] http://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm

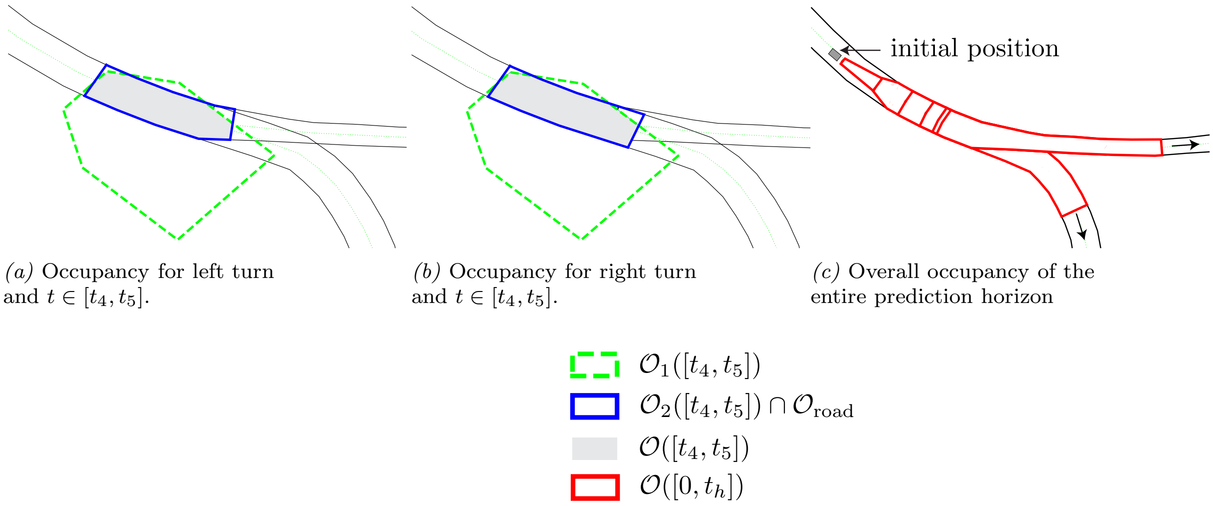
$g = 9.81$ m/s^2; see [189, Fig. 3.3]), $v_{max} = 30$ m/s (considering US highway speed limit of 65 mph plus overspeeding), $v_S = 10$ m/s, $w = 1.8$ m, and $l = 4.2$ m.

We first present the prediction for a single vehicle to demonstrate how our approach works for multiple lanes and road forks. Then, we present the validation results using the data from US highway 101 and the comparison with a high-fidelity vehicle model.

3.5.1 Multi-Lane Road Networks Involving Road Forks

In this section, two different scenarios are considered: a road fork and a multi-lane road network. Both traffic scenarios are created based on real roads modeled in *OpenStreetMap*, which we have processed with *JavaOpenStreetMap*(JOSM)[1]. The time step size is set to $\Delta t = t_{i+1} - t_i = 0.5$ s, the time horizon $t_h = 3$ s, and the initial velocity $v(0) = 25$ m/s for both traffic scenarios. Recall that the occupancy sets are calculated for consecutive time intervals, as was explained in Sec. 3.4. Therefore, no possible future occupancy is missed during the prediction.

Fig. 3.13 illustrates the first scenario, in which the vehicle can drive in two potential lanes. To illustrate the underlying computation of the occupancy for each abstraction model, we plot the predicted sets in individual plots for the left and right turn, respectively, within time interval $[t_4, t_5]$ (Fig. 3.13a and Fig. 3.13b). The predicted occupancy of both lane options for the complete time horizon $\mathcal{O}([t_0, t_h])$ is shown in Fig. 3.13c.


initial position
$\mathcal{O}_1([t_4, t_5])$
$\mathcal{O}_2([t_4, t_5]) \cap \mathcal{O}_{\text{road}}$
$\mathcal{O}([t_4, t_5])$
$\mathcal{O}([0, t_h])$

(a) Occupancy for left turn and $t \in [t_4, t_5]$.

(b) Occupancy for right turn and $t \in [t_4, t_5]$.

(c) Overall occupancy of the entire prediction horizon

Figure 3.13: Occupancy prediction for scenario I.

The second traffic scenario that is considered is illustrated in Fig. 3.14. The road consists of one lane in one direction, and three lanes in the opposite direction. Initially, the vehicle for which the occupancy prediction is done is driving in the center lane, and therefore, in the absence of any other constraint, it

[1] https://josm.openstreetmap.de

can change lanes both towards the left and the right adjacent lane. The predicted occupancy sets are illustrated in Fig. 3.14 for the entire time horizon of $t_h = 3$ s.

Table 3.2 summarizes the computation times for the scenarios described above. It can be seen that the computation time represents only a small fraction of the prediction horizon, i.e., less than 10 %. Thus, our proposed prediction algorithm is suitable for online computation. Moreover, if the prediction of each vehicle is performed in parallel — due to the fact that the prediction of each vehicle can be done independently — one can compute the future occupancy of many surrounding vehicles in a similar time.

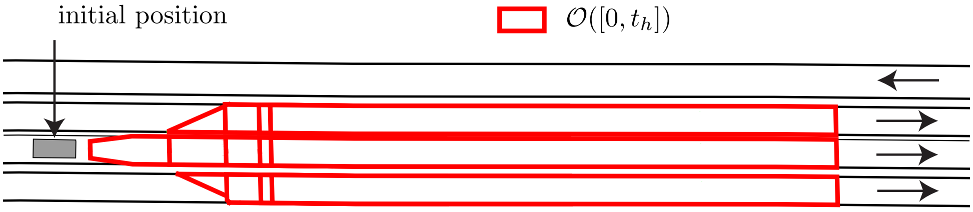


Figure 3.14: Occupancy prediction of scenario II.

Table 3.2: Computation Time for Scenario I and II.

	Computation time	**Prediction horizon**	**Fraction of prediction horizon**
Scenario I	0.19 s	3 s	6.3 %
Scenario II	0.17 s	3 s	5.7 %

3.5.2 Comparison with a High-Fidelity Vehicle Model

Due to the considered constraints $C1 - C5$ in Sec. 3.3.2, as well as in Prop. 3.1, Lemmas 1, 2, and 3, and Theorem 3.1 and 3.2, we have ensured that the predicted occupancy sets are over-approximative. On the one hand, this allows us to consider all possible behaviors of the other traffic participants, and therefore we are able to react to unexpected maneuvers. On the other hand, one could argue that our results — albeit formally correct — may be too conservative. Therefore, what remains is answering the question of how conservative our results are. To investigate this issue, a high-fidelity vehicle model is introduced, in order to compute possible behaviors using rapidly exploring random trees (RRTs) [108]. When computing subsets of reachable sets [28], RRTs have proven useful and, as demonstrated in e.g. [71], they are also more efficient compared to Monte Carlo simulations when diversity of the solutions is important. Please note, however, that RRTs are not a formal method so the comparison with our set-based prediction does not provide a proof for the over-approximative computation. Instead, it only shows how tight our results are when comparing against a high-fidelity model.

The evaluated high-order model considers the vertical load of all 4 wheels due to roll, pitch, yaw, their individual spin and slip, and nonlinear tire dynamics. Three masses determine the multi-body

Table 3.3: Initial values of the high-order model (see [5]).

sprung mass		unsprung mass		other	
	init.		init.		init.
name	**val.**	**name**	**val.**	**name**	**val.**
yaw angle	Ψ_0	roll angle (f)	0	wheel speed (lf)	ω_0
yaw rate	$\dot{\Psi}_0$	roll rate (f)	0	wheel speed (rf)	ω_0
roll angle	0	roll angle (r)	0	wheel speed (lr)	ω_0
roll rate	0	roll rate (r)	0	wheel speed (rr)	ω_0
pitch angle	0	y-velocity (f)	$v_{yf,0}$	pin joint diff. (f)	0
pitch rate	0	y-velocity (r)	$v_{yr,0}$	pin joint diff. (r)	0
x-velocity	$v_{x,0}$	z-position (f)	$z_{f,0}$	x-position	$s_{x,0}$
y-velocity	$v_{y,0}$	z-velocity (f)	0	y-position	$s_{y,0}$
z-position	0	z-position (r)	$z_{r,0}$		
z-velocity	0	z-velocity (r)	0		

dynamics as follows: the unsprung mass and the sprung mass of the front and rear axles. The forces between these masses are described by the dynamics of the suspension and the tire model. Our model is taken from [3, Appendix A] and has been used in previous publications [5]. The vehicle parameters were provided by vehicle 14 in [3, Appendix E], which is a BMW 320i. For the tire dynamics we use the PAC2002 Magic-Formula tire model whose parameters are taken from the example of a PAC2002 tire property file in [134]. The high-order model is initialized as summarized in Table 3.3, where f/r indicates front/rear. Next, we introduce the initial heading Ψ_0, the initial yaw rate $\dot{\Psi}_0$, the initial rotational speed of the wheels $\omega_0 = v_0/R$ (v_0: initial velocity, R radius of the wheel), the initial velocity in x/y-direction of the vehicle coordinate system $v_{x,0}/v_{y,0}$, the initial height over ground z_0, and the initial x/y-position $s_{x,0}/s_{y,0}$.

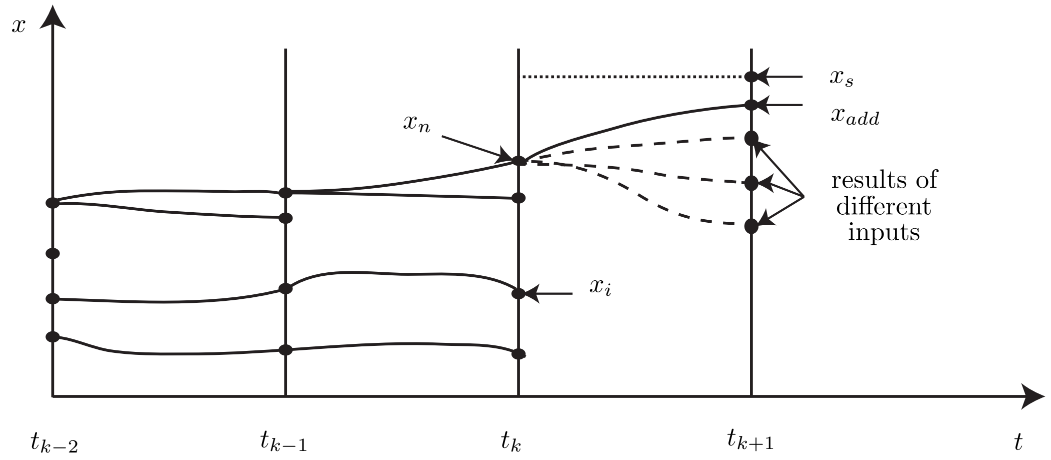


Figure 3.15: Sampling procedure of our RRT approach.

Since we are interested in the results for consecutive time intervals, we slightly modify the procedure from a standard RRT approach, e.g. [28], such that a constant sampling density for each time window is achieved, as illustrated in Fig. 3.15. In this thesis, we apply the procedure taken from [5]:

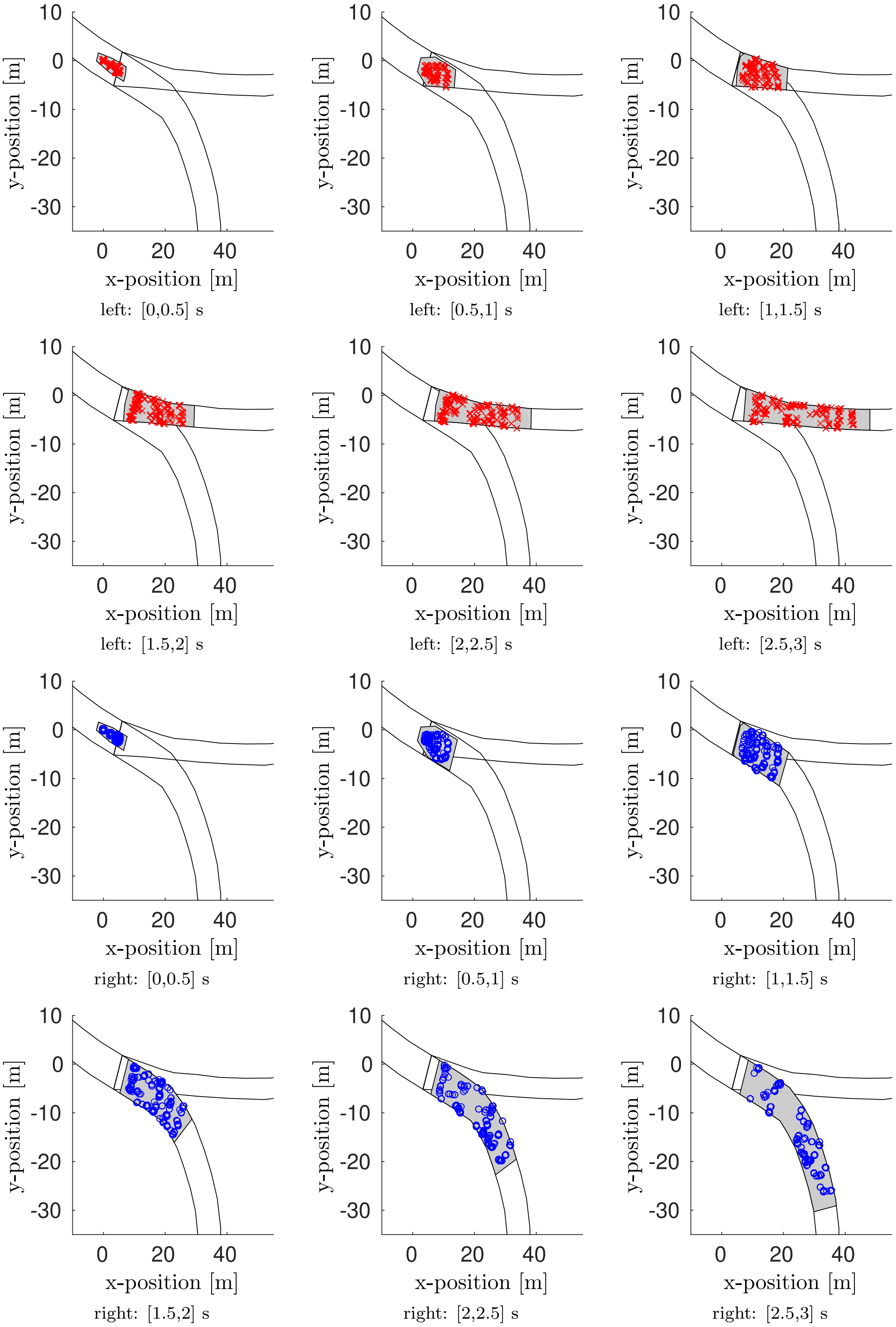
1. Initialize the discrete set of states for the next time interval as $\mathcal{X}(\tau_{k+1}) = \varnothing$.
2. Generate a sample x_s from the state space.
3. Find the nearest state x_n according to a distance measure ρ so that $x_n = \arg\min(\rho(x_s, x_i))$, where $x_i \in \mathcal{X}(\tau_k)$.
4. Obtain the input u that drives x_n to the new state x_{add} closest to x_s.
5. Add x_{add} to the set of states for the next time interval $\mathcal{X}(\tau_{k+1})$.
6. Repeat steps 2-5 for a predefined number of samples, then go to the next time interval and start with step 1.

Note that if we would initialize the discrete set of states for the next time interval as $\mathcal{X}(\tau_{k+1}) = \mathcal{X}(\tau_k)$, one would obtain the approach used in [28]. For sampling space $\mathcal{X}$ we use only the x-position and y-position and we choose the Euclidian distance as a distance measure. Then, the optimal input u determined by the steering angle and the brake/acceleration pedal angle, that steers the system from the state x_n to state x_s, by minimizing $\rho(x_{add}, x_s)$, is chosen by testing 12 combinations of steering and acceleration such that the resulting acceleration lies on Kamm's circle. New samples for the inputs are changed every 0.25 s, and each time interval $\mathcal{X}(\tau_{k+1})$ contains 100 samples. Different initial states compared to the previous subsection (Sec. 3.5.1) are used — the initial velocity is set to 10 m/s instead of 25 m/s, and the initial position is closer to the road bifurcation — such that we can demonstrate how the prediction is affected by changing the initial state.

In Fig. 3.16, the results corresponding to the high fidelity model are illustrated. One can observe that for each time interval, the vehicle is able to drive close to the borders of the predicted occupancy sets. The RRT sampling for turning left and right is illustrated separately. It is worth mentioning that computing the RRTs of high-fidelity models does not fulfill the real-time constraints. We have used the same machine for the calculation of set-based prediction, and sampling the RRTs; both calculations were performed using Matlab. In contrast to our approach that generated the results within one fraction of a second, the prediction using the high-fidelity model took approximately 10 hours.

3.5.3 Comparison with Real Traffic Data

In this section, we further assess our approach against real traffic data recorded from a highway, by showing that the tracked occupancy of each traffic participant lies within our predicted over-approximative occupancy sets. This evaluation is done for a time horizon of $t_h = 2$ s and use a time step size of $\Delta t = t_{i+1} - t_i = 0.4$ s. We introduce the $t_{rec,i}$ as the time of the i^{th} recorded set of occupancies. Starting



left: [0,0.5] s

left: [0.5,1] s

left: [1,1.5] s

left: [1.5,2] s

left: [2,2.5] s

left: [2.5,3] s

right: [0,0.5] s

right: [0.5,1] s

right: [1,1.5] s

right: [1.5,2] s

right: [2,2.5] s

right: [2.5,3] s

Figure 3.16: Comparison of set-based occupancy prediction with results from the RRT computation for each time interval. Crosses show possible positions of the high-fidelity model for the left turn and dots show possible positions for the right turn. Gray regions indicate the predicted occupancy.

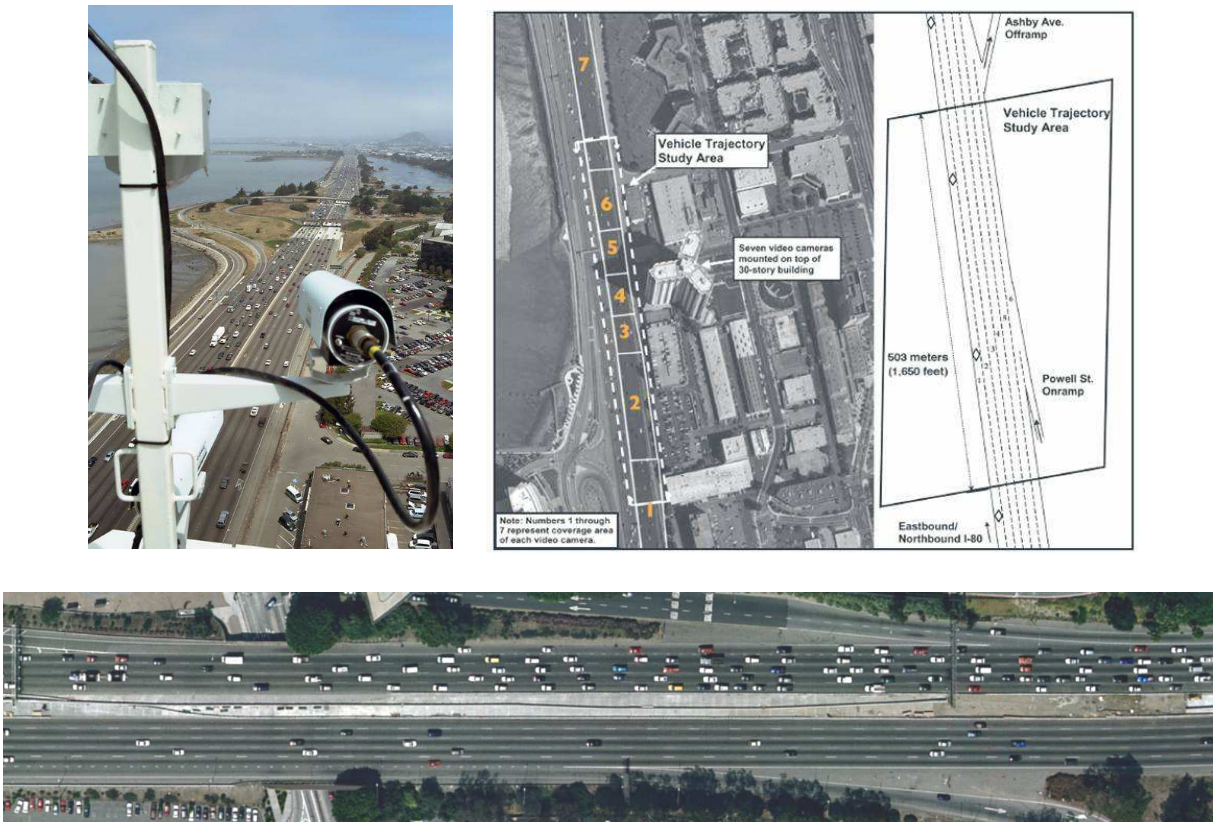

7
6
5
4
3
2
1
Vehicle Trajectory Study Area
Seven video cameras mounted on top of 30-story building
Note: Numbers 1 through 7 represent coverage area of each video camera.
Ashby Ave. Offramp
Vehicle Trajectory Study Area
503 meters (1,650 feet)
Powell St. Onramp
Eastbound/ Northbound I-80

Figure 3.17: *(a)* A video camera mounted on top of a building used to record trajectory data [65]. *(b)* Aerial photograph that shows the study area [65]. *(c)* Aerial photo of US highway 101 section.

from the occupancies at $t_{rec,i}$, we predict the occupied sets for the time interval $[t_{rec,i}, t_{rec,i} + t_h]$ and then check if the recorded occupancies are enclosed by the over-approximative predicted sets. If the real occupancies lie inside the predicted ones, we move forward to the next time interval $[t_{rec,i+1}, t_{rec,i+1} + t_h]$. This procedure is repeated until all times $t_{rec,i}$ within the given time horizon t_h have been verified.

The dataset was collected from the Next Generation Simulation (NGSIM) program[1]. The traffic data was obtained on a 0.6 km segment of the US Highway 101 (Hollywood Freeway) in Los Angeles, California, on June 15th, 2005, containing detailed trajectory data for each traffic participant, over the entire period from 07:50 am to 8:35 am. The area was monitored using eight video cameras placed on various buildings, as shown in Fig. 3.17.

For illustration purposes, we only present the results corresponding to a section of the recorded highway sector in Fig. 3.18. Otherwise, the prediction of the individual vehicles would not be visible due to the length of the recorded section of the highway. Nevertheless, we have calculated the predicted occupancy set for all recorded vehicles, for the given time horizon. The initial occupancy of the vehicles

[1] http://www.fhwa.dot.gov/

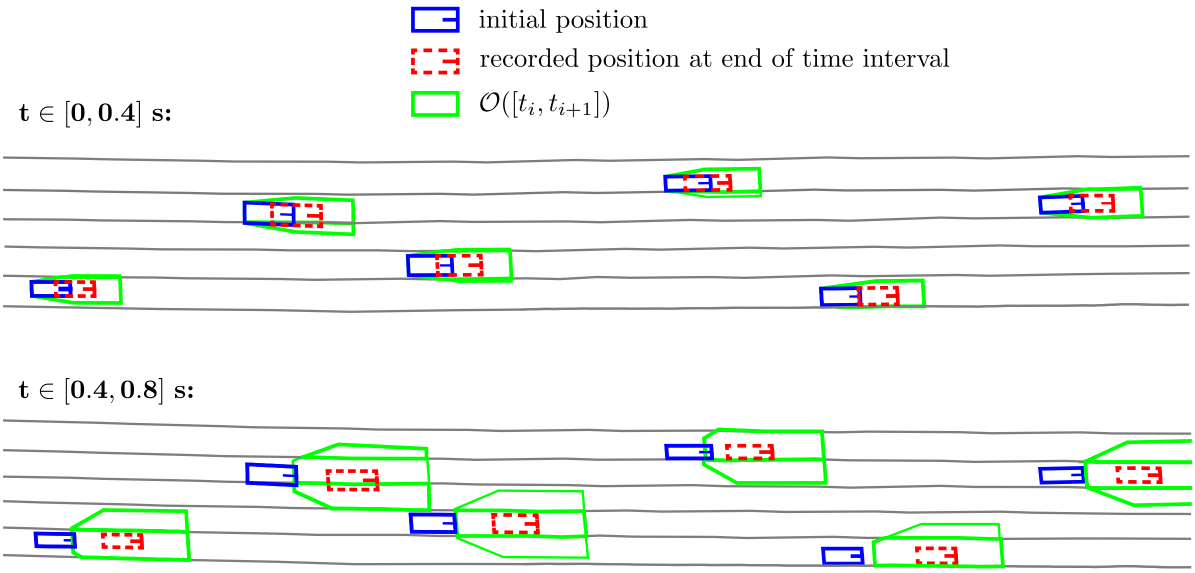

initial position
recorded position at end of time interval
$\mathcal{O}([t_i, t_{i+1}])$
$\mathbf{t} \in [\mathbf{0}, \mathbf{0.4}]$ s:
$\mathbf{t} \in [\mathbf{0.4}, \mathbf{0.8}]$ s:

Figure 3.18: Occupancy prediction using recorded vehicles from US highway 101.

is represented by solid rectangles where the inner line indicates the driving direction (see Fig. 3.18). The dashed rectangles show the recorded occupancy of each car at the end of the time interval ($t = t_{rec,i} + \Delta t$), while the polygons with solid border mark the predicted occupancy, as illustrated in Fig. 3.18.

The computation of the occupancy prediction took an average of 0.2 s per vehicle for 5 instead of 6 time intervals as in scenarios I and II, and thus is only slightly slower than the results of Table 3.2. The most significant result of this evaluation performed against real traffic data is that, out of a total number of 1074 considered vehicles, none of them violated the predicted occupancy.

3.6 Conclusions

In this chapter, we have introduced a novel framework that can formally calculate the predicted occupancy sets of surrounding traffic participants on arbitrary road networks. The highlight of our approach, compared to existing work in this field, is that it can formally guarantee the borders of occupied areas utilizing techniques from reachability analysis. The distinctiveness of our approach is that the correctness of the prediction is not jeopardized if a specific constraint is removed from the model of the other traffic participants due to a violation of that constraint by a vehicle. This action, however, comes with a cost that the predicted occupancy grows faster and therefore the ego vehicle is required to generate more conservative trajectories to avoid overlapping with the enlarged occupancy of that corresponding traffic participant.

Therefore, our approach proves to be inherently safe since the safety of the ego vehicle is not endangered if some constraints on the model of other traffic participants are omitted or removed due to current behavior, but it results into a more conservative behavior. Note that only the set-based techniques have this property as the predicted occupancy sets generated with the other techniques, e.g., multiple simulations, are not monotonically enlarged when removing constraints.

The evaluation of the proposed approach was done against real traffic data collected from a segment of US Highway 101. The validation results highlight the over-approximative characteristic of our approach, which is able to predict all the future behaviors of other traffic participants, for a given time horizon. At the same time, our results demonstrate that the predicted sets are tight when comparing the results obtained with a high-fidelity model. Moreover, our set-based prediction method is real-time capable since the prediction computations only required a fraction of the considered time horizon (less than 10 %). All these features confirm that our approach is indeed suitable for online motion planning and verification algorithms.

Chapter 4

Fail-Safe Motion Planning

This chapter proposes a method to generate comfortable and provably-safe maneuvers for autonomous vehicles. Similar to the longitudinal motion planning presented in Chapter 2, this approach consists of three steps: first, a long-term optimal trajectory is generated for the ego vehicle that accounts for the predicted, most likely future behavior of other traffic participants. However, since most of the time, the prediction of the other vehicles is not fully accurate due to various reasons (e.g., uncertainty in the measurements, or just due to unexpected maneuvers), safety cannot be guaranteed. Since a single prediction does not suffice to ensure safety, we compute the over-approximative occupancy sets of the other traffic participants, as described in Chapter 3, which consider all possible future maneuvers for short, consecutive time intervals. These sets are then incorporated as constraints when generating corresponding emergency maneuvers for the ego vehicle.

The intended part of the optimal long-term trajectory (recall Fig. 3.2) is followed by the ego vehicle only after it is verified as safe. If the given section of the trajectory (i.e., the intended part) is not verified as safe, a precomputed emergency maneuver that can bring the ego vehicle to a safe state is engaged. A trajectory is verified as safe if its corresponding occupancy does not intersect with the occupancy of the other traffic participants, and, starting at the end of this trajectory, there exists an emergency maneuver whose corresponding occupancy does not intersect with the corresponding over-occupancy sets of the surrounding vehicles.

Verifying at each time instance whether there exists a safe emergency maneuver is computational expensive, and often not required. Therefore, we calculate the maximum time horizon the ego vehicle can safely follow the long-term trajectory while guaranteeing that at the end of this time horizon, an emergency maneuver is guaranteed to exist.

The remainder of this chapter is organized as follows: In Sec. 4.1 we present a review of related literature. Then, we introduce our contributions in Sec. 4.2. Some preliminaries and assumptions are presented in Sec. 4.3, that guide the reader towards the problem formulation. The framework that provides the fail-safe motion planner is described in Sec. 4.4. The methodology used to calculate the latest point in time where an emergency maneuver is guaranteed to exist is presented in Sec. 4.5. Lastly, numerical experiments used to demonstrate the proposed approach are shown in Sec. 4.6.

The section regarding the generation of fail-safe motion planning is based on work published in [120]. The section concerning how to determine the maximum time for a vehicle to safely follow a given trajectory is published in [122].

4.1 Introduction and State of the Art

Most previous work on trajectory planning of vehicles considers only one predicted trajectory of each other traffic participant (e.g., the most probable trajectory). However, relying only on the most probable trajectory of other traffic participants is not safe, since unexpected maneuvers may result in inevitable collisions.

There also exist approaches that consider multiple future behaviors of the other traffic participants, as already described in Sec. 3.1. Yet, safety cannot be guaranteed, since this finite number of possible maneuvers of the surrounding vehicles does not consider the entire set of possible future maneuvers. Therefore, unexpected maneuvers that may occur, but are not considered, could result in a collision.

Generating safe trajectories for an autonomous vehicle in a dynamic environment is an exhaustive research area, yet no provable correct solution has been proposed, that considers all possible future behaviors of other traffic participants. In [168] a “model for safety assurance” of autonomous vehicles has been proposed, where it is shown that “improper behavior of others” can be considered as well. However, it is assumed that all traffic participants will try to avoid a possible dangerous situation. While generating trajectories for an autonomous vehicle in static environments is a problem for which optimal solutions are already available (e.g., [29], [107]), motion planning in dynamic environments is not a trivial task since the unknown future behavior of other traffic participants has to be incorporated into the planning problem.

Motion planning techniques have been comprehensively studied and different approaches were already proposed in the literature. These approaches can be categorized as following: 1) by planning in discrete space (e.g., grid-based approaches [107], planning using motion primitives [50], rapidly-exploring random trees [33, 102, 105], and road maps [26, 89, 90, 175, 177]), and 2) by planning in continuous space (e.g. optimal control, Model Predictive Control (MPC) [16, 147, 193, 194], and elastic bands [154]). A survey on existing algorithms for collision-free trajectory planning for mobile robots can be found in [76] and motion planning techniques for self-driving vehicles in [146].

Planning in discrete space

Sampling-based motion planning algorithms such as Probabilistic Road Maps (PRM) [26,89,90,175,177] or Rapidly-Exploring Random Trees (RRT) [33,102,105] demonstrate good performance in practice, in particular for path planning in high-dimensional non-convex state spaces. Nevertheless, the control inputs that are used to explore the configuration space have a great impact on the performance of sampling-based approaches.

To reduce the computational burden caused by sampling, predefined and parametrized trajectories, which are often referred to as a motion primitives (e.g., turn left, right turn, go straight, etc) are introduced in [50]. The generation of these maneuvers is done such that they can be easily inter-connected, to form a maneuver automaton [51]. Due to the off-line computation of the motion primitives, the planning algorithms that use a maneuver automaton are suitable for real-time applications.

The construction of formally verified maneuver automata using reachability analysis has been investigated in [72]. In [137], a heuristic graph search is used to find a feasible path. Another approach is presented in [61], where the motion primitives are modeled as a hybrid system, the discrete states are the predefined trajectories, and the control input that steers the system from one state to another is defined by maneuvers. Therefore, optimal path generation becomes a classic hybrid optimal control problem [167].

Planning in continuous space

Elastic bands have been introduced in [154], with the aim to fill the gap between path planning and control to make it possible to plan a trajectory directly in continuous space. The key feature of the elastic bands is that their corresponding paths can be deformed in real-time in order to react to changes in the environment. The utility of elastic bands has been demonstrated for various purposes, such as emergency maneuver generation [73], trajectory planning [64], or adaptive cruise control [56], where only one path is computed. To overcome the issue that a single elastic band may fail to describe a desired path, several elastic bands are generated in [163]. Then, a single solution that minimizes a given cost function is selected.

Optimal control or MPC [16,147,193,194] are techniques widely used for generating optimal trajectories where different constraints can be directly embedded into the planning problem. In [16], MPC is utilized for trajectory planning to prevent lane departure. Collision-free trajectories that take static obstacles into considerations have been investigated in [147]. In [194], collision avoidance is achieved through steering and braking under the assumption that the obstacles move with constant velocity. The authors of [97] proposed a provable safe system, however, the inter-vehicle and vehicle to infrastructure communication are vital to the proposed approach, since it is assumed that the information regarding lane changes intentions of all interacting vehicles can be exchanged.

Due to the dynamic characteristic of traffic scenarios, one must consider a replanning mechanism to be able to avoid possible collisions. Although most previous work considers this mandatory mechanism, there is no approach for determining the moment at which replanning should be performed in order to guarantee safety. Instead, replanning is done on the fly when a dangerous situation may already be inevitable. Therefore, the authors of [66] propose an algorithm that computes an adaptive time horizon that dictates when replanning should be performed. The so-called "time to potential failure" determines how long the current trajectory is safe under some given assumptions. However, if no further safe trajectory is found, a collision may be imminent. Therefore, there exists no guarantee that by the end of this time horizon, a safe maneuver can be found. To cope with the dynamic environment, the authors of [52] propose an adaptive planning horizon computation based on the changing rate of the environment's configuration. However, this approach cannot guarantee that another feasible maneuver exists after the computed time horizon.

The above methods cannot ensure safety in every traffic scenario since there exist unconsidered possible future maneuvers of other traffic participants, when generating a trajectory for the ego vehicle, which can result in a collision. Some emerging attempts toward safe motion planning exist. For example, in [25], an emergency maneuver is generated, considering the initial state as the final state of the generated long-term trajectory. However, not all possible future trajectories of the surrounding vehicles are considered. Instead, viable inter-vehicle communication and future trajectories are assumed to be known [25].

In our previous work, [120], a fail-safe motion planner is proposed. The proposed planner consists of two parts: i) an optimal trajectory is generated for the ego vehicle considering the most likely trajectories of other traffic participants, for a long time horizon and ii) at each time step, the overapproximative occupancy sets that enclose all the possible trajectories of other vehicles for a given time interval are computed. Then, it is verified whether there exists an emergency maneuver that can bring the ego vehicle to a safe state without intersecting any other occupancy sets of the surrounding vehicles. If such an emergency maneuver exists, the ego vehicle can continue following the optimal trajectory. Otherwise, the emergency maneuver generated at the previous time step is engaged until a new safe trajectory is found.

The authors of [150] divide planning tasks into three levels: route planning (provides specific tasks to be accomplished in order to reach the ultimate goal; e.g., follow a specific highway route), behavioral planning (facilitate decision-making in order to properly interact with the other traffic participants; e.g., changing a lane) and motion planning (generate an admissible set of actions such that the local objectives are fulfilled; e.g., generate consecutive values of acceleration such that a safe inter-vehicle distance is kept, with respect to the other traffic participants).

This work assumes that a route and a behavioral planner exist that provides a set of relevant driving decisions in order to reach the desired goal (e.g. lane change for switching between two roads) or to increase customer satisfaction (e.g., changing the lane to prevent driving behind a slower vehicle for a long time). There already exists comprehensive work that tackles questions like: "is a lane change mandatory?" or "is a lane change desirable?" by using different approaches such as rule-based methods [23, 48], utility-based frameworks [54, 139, 140, 191], probabilistic-based approaches [19, 165], model predictive control [141], game-theory-based methods [130], or Petri nets-based solution [53].

4.2 Contributions

Although there exists a lot of work on motion planning for autonomous vehicles, how to guarantee safety in a dynamic environment is still a major issue, due to the uncertainties introduced by the infinite number of possible maneuvers of other traffic participants. In addition, achieving comfortable and efficient driving while guaranteeing safety are, most of the time contradictory requirements (e.g. conservative maneuvers may be safe, however, they could lead to decreased traffic throughput that diminishes efficiency). A fail-safe motion planning framework is therefore required in order to be able to safely react in any traffic scenario.

With the aforementioned issues in mind and considering the gaps that exist in the current state-of-the-art, the main contributions of this chapter are:

- We propose an architecture to guarantee safety in motion planning by accounting for every possible future maneuver of other traffic participants, in both longitudinal and lateral directions.

- Unlike previous work, we generate not only one, but two trajectories for the ego vehicle. First, we compute an optimal trajectory that considers the most likely behavior of the other traffic participants. Another trajectory, corresponding to an emergency maneuver is generated by considering the overapproximative occupancy sets of the surrounding vehicles at each time step. The long-term, optimal trajectory is followed only if it is verified as safe, i.e., there exists an emergency maneuver that concatenated to the optimal trajectory can bring the ego vehicle to a safe state.

- In order to generate safe guaranteed emergency maneuvers for the ego vehicle, overapproximative occupancy sets (see Chapter 3 for details) are generated and included into the calculation of the evasive maneuvers.

- Moreover, since verification of the optimal trajectory at each time instance is computationally expensive, and often not required, instead of generating an emergency maneuver at each time step, we compute the maximum amount of time a vehicle can safely follow a given trajectory. Then, by the end of this time horizon a safe emergency maneuver is guaranteed to exist.

The section regarding the generation of fail-safe motion planning is based on work published in [120]. The section concerning how to determine the maximum time for a vehicle to safely follow a given trajectory is based on work published in [122].

4.3 Preliminaries and Problem Formulation

In this chapter, we extend the approach presented in Chapter 2, by considering that the ego vehicle can autonomously perform a lateral maneuver in addition to a longitudinal one. This implies that more thorough attention to the existing road network is required since this may influence the way the ego vehicle responds to changes in the traffic scenario.

In the following, we will consider a road network determined by multiple adjacent lanes with arbitrary curvature. As already mentioned, an efficient representation of road networks can be found in [27]. For a more concise road network representation, the reader is referred to Section 3.3.1.

Thereafter, we will intentionally consider only one surrounding vehicle V_{lead}, in order to focus on the novel aspects introduced in this chapter. However, this approach is applicable to arbitrarily many surrounding vehicles. Of course, the physical capability of the on-board sensors must be considered. Therefore, only the vehicles driving within the field of view shall be taken into account when planning a trajectory for the ego vehicle. Note that, unlike the field of view considered in Chapter 2, where we are only looking ahead of the ego vehicle, in this chapter it is a requirement to consider the rear field of view of the ego vehicle. This is mandatory in order to guarantee safety when the ego vehicle performs a lane change.

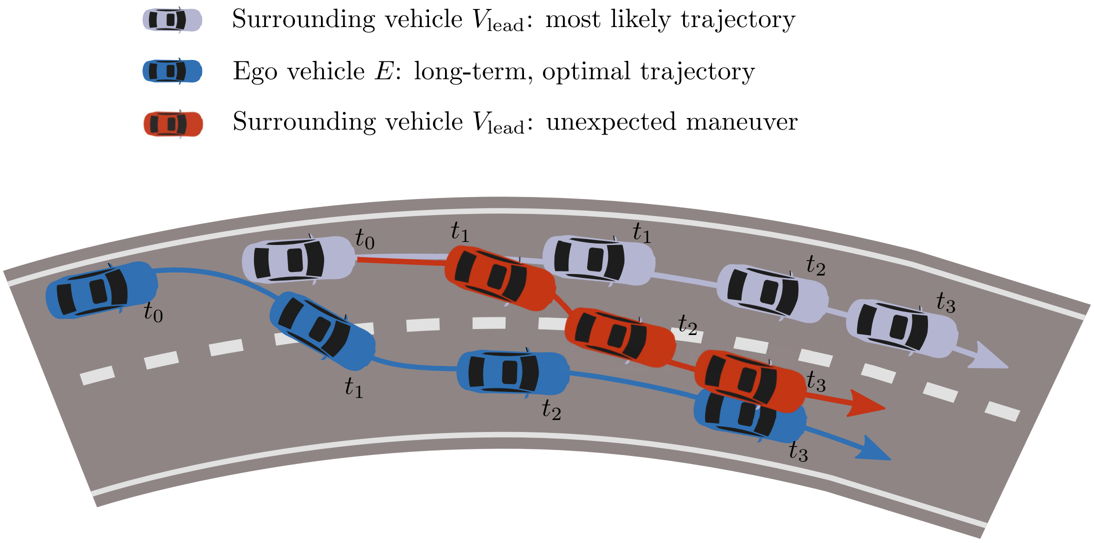

Surrounding vehicle V_{lead}: most likely trajectory
Ego vehicle E: long-term, optimal trajectory
Surrounding vehicle V_{lead}: unexpected maneuver
t_0
t_1
t_1
t_2
t_3
t_0
t_2
t_1
t_3
t_2
t_3

Figure 4.1: The long-term trajectory of the ego vehicle is generated by considering only the most likely behavior of the other traffic participants. However, if the leading vehicle performs an unexpected maneuver, the ego vehicle may not be able to replan or find any feasible emergency maneuver in order to avoid a possible collision.

Let us consider the traffic scenario illustrated in Fig. 4.1. Here, a long-term trajectory is generated for the ego vehicle (represented by blue) by considering only the most likely behavior of the surrounding vehicle (depicted with gray). In this scenario, the predicted maneuver of the surrounding vehicle is to keep its current lane. However, if the surrounding vehicle performs an unexpected maneuver, for example by changing lanes, the ego vehicle may not be able to find a further feasible trajectory (not even an emergency maneuver) such that a collision can be avoided.

The main objective of this chapter is to design a motion planner that guarantees an emergency maneuver for each time instance that can bring the ego vehicle to a safe state (e.g., standstill), without colliding with other traffic participants. Note that in order to guarantee safety, all possible future maneuvers of other traffic participants must be considered when calculating an emergency maneuver of the ego vehicle. However, although an emergency maneuver is kept available at each time instance, this does not imply that it must be engaged.

Moreover, depending on the current traffic situation, generating an emergency maneuver at each time instance may not be required (e.g., if the ego vehicle is driving at a large distance with respect to the other traffic participants). In order to optimize computing time, instead of generating an emergency maneuver at each time instance, we calculate the maximum time the ego vehicle can follow a given trajectory, while guaranteeing a safe emergency maneuver exists by the end of this time.

Let us provide some preliminaries and notation used throughout this chapter. In the following, the area defined by the left and right boundary of a road is referred to as *lane*. We do not differentiate between the vehicles driving in different lanes for the problem we want to tackle in this chapter[1]. Therefore, in this chapter, we denote the number of the considered surrounding vehicles by N.

To generate a safe emergency maneuver, the uncertainties introduced by the unknown behavior of other traffic participants and by the measurements of the environment (e.g., static obstacles or moving surrounding vehicles) must be considered. For this purpose, we calculate the overapproximation of the future occupancies of the other traffic participants, as described in [10] and implemented by the tool SPOT [101]. For more details on the generation of overapproximative occupancy sets of other vehicles, the reader is referred to Chapter 3. We denote by $Occ(V_k, \tau_i),\ k \in \{1, 2, \ldots, N\}$ the overapproximative predicted occupancy of a surrounding vehicle V_k. This is computed for a given time interval $\tau_i = [t_i, t_i + \Delta t]$, which encloses all possible occupancies complying with the considered set of traffic rules[2]. We define $\Gamma(V_k, \tau_i) \coloneqq \bigcup_{t \in \tau_i} \Gamma(V_k, t)$ as the union of the occupancy sets of the vehicle V_k corresponding to the time interval τ_i. Then, we can define the relation between the occupancy $\Gamma(V_k, \tau_i)$ and overapproximative

[1] We differentiate between vehicles driving on different lanes only in the lane-following mode (see Chapter 2), where different control strategies are generated depending whether there is a vehicle following driving situation, or a cut-in maneuver is performed by a surrounding vehicle.

[2] For more information regarding the considered traffic rules, the reader is referred to Chapter 3.

occupancy set $Occ(V_k, \tau_i)$ as the following:

$$\Gamma(\tau_i) \subset Occ(\tau_i), \forall \tau_i = [t_i, t_i + \Delta t], \tag{4.1}$$

as illustrated in Fig. 4.2.

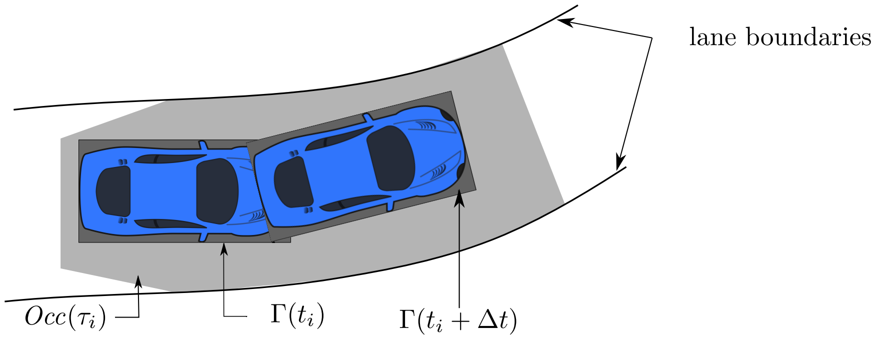


Figure 4.2: Occupancy and overapproximative occupancy of a vehicle.

Let us denote the optimal trajectory of the ego vehicle as $\mathcal{X}_{\text{opt}}(t) = [x_{\text{opt}}(t), x_{\text{opt}}(t + \Delta t), \cdots, x_{\text{opt}}(T_{\text{opt}})]$, where Δt is the time step, $T_{\text{opt}} = n_{\text{opt}} \cdot \Delta t$ is the time horizon for generating the optimal trajectory, $T_{\text{opt}} \in \mathbb{R}_+$, and $n_{\text{opt}} \in \mathbb{N}$ is a given parameter. Similarly, we denote by $\mathcal{X}_{\text{emg}}(t) = [x_{\text{emg}}(t), x_{\text{emg}}(t + \Delta t), \cdots,$ $x_{\text{emg}}(T_{\text{emg}})]$ the emergency maneuver initiated at time t, where $T_{\text{emg}} = n_{\text{emg}} \cdot \Delta t$ represents the time horizon for generating the emergency maneuver, with $n_{\text{emg}} \in \mathbb{N}$.

We denote by t^* the maximum time horizon for which the ego vehicle can follow a given optimal trajectory while guaranteeing safety. To guarantee safety, the following two constraints must be satisfied:

1. The occupancy associated with states within the optimal trajectory $\Gamma(x_{\text{opt}}(t_i)), \forall t_i \leq t^*, 0 \leq i \leq n_{\text{opt}}$ must not intersect with the corresponding overapproximative occupancy set of other traffic participants $Occ(V_k, \tau_i)$, $k \in \{1, \cdots, N\}$.

2. There exists an emergency maneuver initiated at t^* whose corresponding occupancy $\Gamma(x_{\text{emg}}(t_j))$, $\forall j \in \{i, \ldots, n_{\text{emg}}\}$ must not intersect with the corresponding overapproximative occupancy set of other traffic participants $Occ(V_k, \tau_j)$, $t^* \leq t_j \leq T_{\text{emg}}$, $k \in \{1, \cdots, N\}$.

The objective of this chapter is twofold: first, we propose a motion planning framework that guarantees the safety of the ego vehicle. That is, an emergency maneuver that considers all maneuvers of the surrounding vehicles is guaranteed to exist at each time instance. Second, since generating an emergency maneuver at each time instance is computationally expensive and often not required, we want to determine the maximum time the ego vehicle can safely follow the long-term trajectory. That means that there exists an emergency maneuver to be initiated at the end of this time horizon that can steer the ego vehicle towards a safe state.

We denote by $\lceil \cdot \rceil$ the *ceiling operator* which provides the least integer greater than or equal to the given parameter. Using the above notations, the addressed problem can be formulated as follows:

$$\begin{aligned} t^* \;=\; & \max_{0\leq i\leq n_{\text{opt}}} \quad t_i, \\ \text{subject to} \quad & \forall k \in \{1,\cdots,N\}\ \forall r \in \{0,\cdots,i-1\}, \\ & \forall j \in \{i,\cdots,n_{\text{emg}}\}, \exists \mathcal{X}_{\text{emg}}(t^*) : \\ & \Gamma(x_{\text{opt}}(\tau_r)) \cap Occ(V_k,\tau_r) = \varnothing \quad \wedge \\ & \Gamma(x_{\text{emg}}(\tau_j)) \cap Occ(V_k,\tau_j) = \varnothing, \\ & n_{\text{opt}} = \left\lceil \frac{T_{\text{opt}}}{\Delta t} \right\rceil,\ n_{\text{emg}} = \left\lceil \frac{T_{\text{emg}}}{\Delta t} \right\rceil. \end{aligned} \tag{4.2}$$

4.3.1 Vehicle System Dynamics

Since we consider that the ego vehicle can follow its own lane and change its lane as well, the kinematics of lateral vehicle motion must be taken into consideration. In contrast to Chapter 2, different motion model is used for the ego vehicle such that steering is considered as well. For this, we extend the longitudinal dynamics used in Chapter 2 to a kinematic bicycle model (see Fig. 4.3) as presented in [194]:

$$\dot{s}_x = v\cos\psi, \tag{4.3}$$

$$\dot{s}_y = v\sin\psi, \tag{4.4}$$

$$\dot{\psi} = \frac{v\ \delta}{l\left[1+\left[\dfrac{v}{v_{\text{ch}}}\right]^2\right]}, \tag{4.5}$$

$$\dot{\delta} = u_1, \tag{4.6}$$

$$\dot{v} = u_2, \tag{4.7}$$

where the x- and y-positions s_x, s_y, velocity v, yaw angle ψ, and steering angle of the wheels δ determine the state variables, as illustrated in Fig. 4.3. The control inputs are the steering rate u_1 and the acceleration u_2. Let us define the current state of the ego vehicle as $x = [s_{\text{x}}, s_{\text{y}}, \psi, \delta, v]^T$, and the input vector as $u = [u_1, u_2]^T$. Additionally, the dynamics of the ego vehicle depend on two parameters, the wheel base l, and the characteristic velocity[1] v_{ch}.

To consider the vehicle's physical limitations, as well as the safety-related ones, the following constraints on state and input variables are imposed:

$$0 \leq v \leq v_{\max}, \tag{4.8}$$

[1] The characteristic velocity v_{ch} depends on the mass and the cornering stiffness of the vehicle, which characterizes the dynamics of the bicycle model [194].

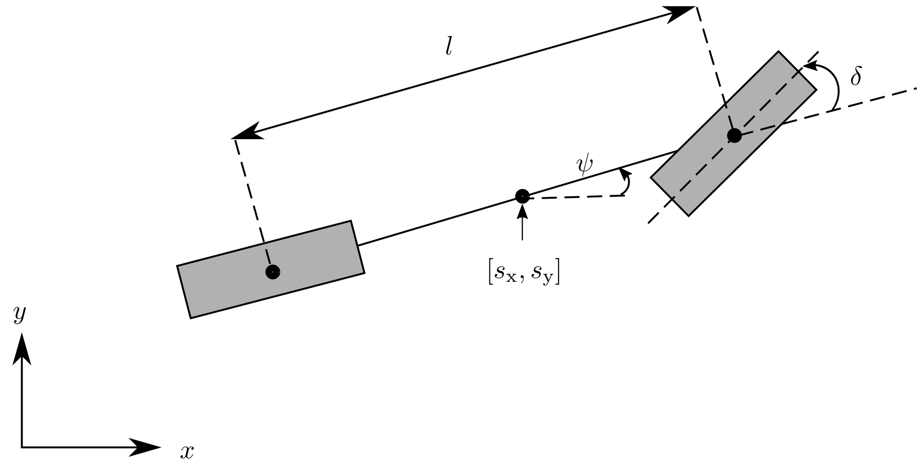


Figure 4.3: Kinematic bicycle model.

$$a_{min} \leq u_2 \leq a_{max}. \tag{4.9}$$

$$\Gamma(x_{\mathrm{ego}}, t) \in \mathit{lanes}, \tag{4.10}$$

$$\delta_{\min} \leq \delta \leq \delta_{\max}, \tag{4.11}$$

$$\dot{\delta}_{\min} \leq u_1 \leq \dot{\delta}_{\max}. \tag{4.12}$$

The inequalities (4.8), (4.9), (4.11), and (4.12) refer to physical constraints, i.e., the possible values of the velocity, acceleration, steering, and steering rate, respectively. The other constraint (4.10) refers to safety which states that the ego vehicle shall not exit the lanes' boundaries. The central safety constraint (see the formulated problem described in (4.2)) refers to the intersection between the occupancy set of the ego vehicle and the overapproximative occupancy sets of the surrounding vehicles.

The boundary parameters for the velocity $v_{\max}$, the acceleration $a_{\min}$, $a_{\max}$, the steering angle $\delta_{\min}$, $\delta_{\max}$, and steering rate $\dot{\delta}_{\min}$, $\dot{\delta}_{\max}$ are assumed to be given. These values can follow either from the vehicle's physical capabilities, or they can be defined as safety measures (e.g., maximum admissible velocity $v_{\max}$). In the following, we denote by $\mathcal{X} \subset \mathbb{R}^5$ the set of states, and by $\mathcal{U} \subset \mathbb{R}^2$ the set of inputs that satisfy the corresponding state/input inequalities (4.8)-(4.12).

4.4 Fail-safe Motion Planner

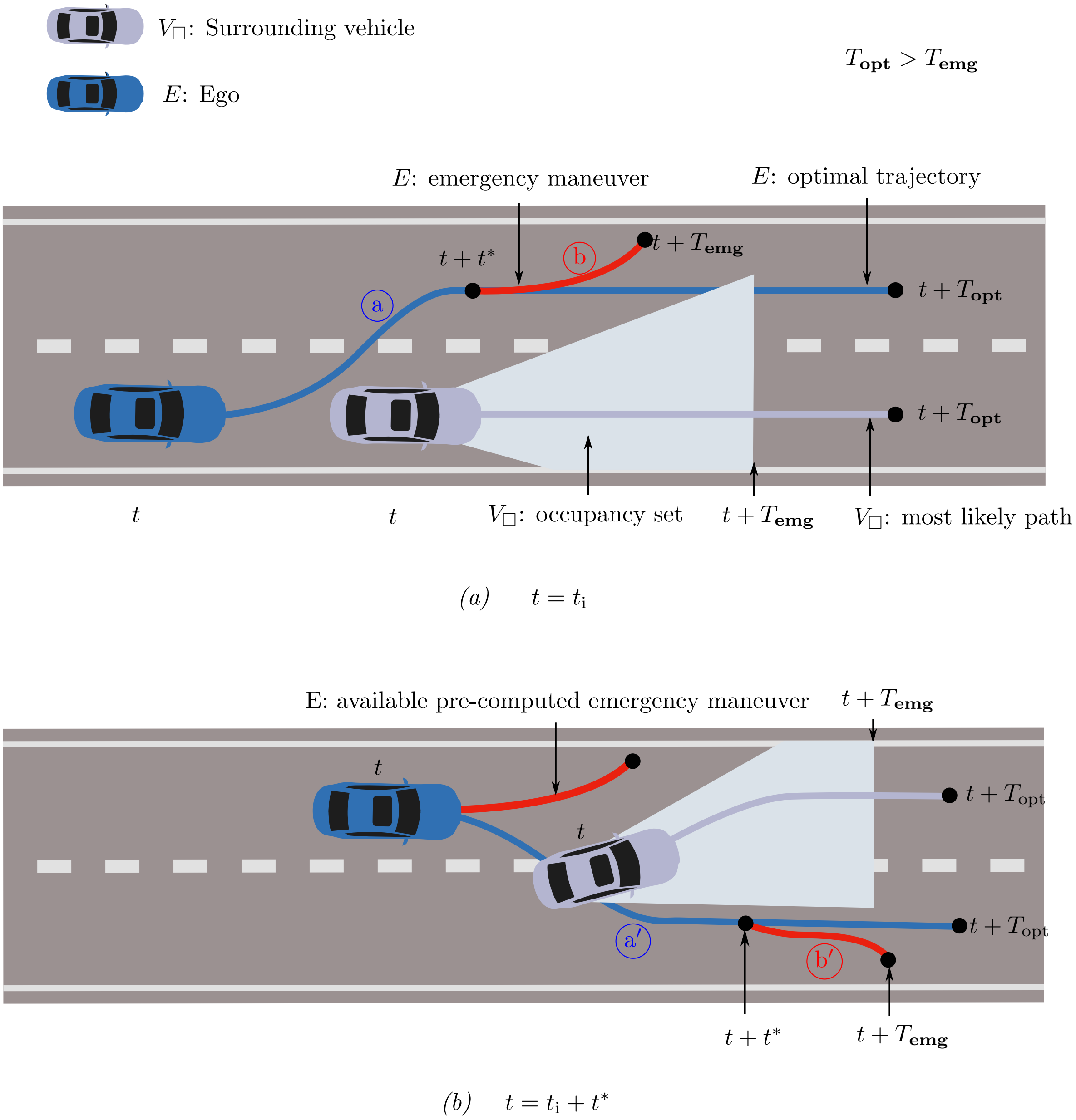
In this section, a framework for motion planning with safety guarantees of the ego vehicle is presented. Similar to adaptive cruise control with safety guarantees (see Chapter 2), the primary goal of the proposed motion planner is to achieve safety, by foreseeing possible unexpected maneuvers of other traffic participants and having the ability to avoid any collision with them. Moreover, when possible, comfort should not be jeopardized.

To consider both safety and comfort, a method similar to the one presented in Chapter 2 is proposed. However, unlike the adaptive cruise control approach, where the ego vehicle can only follow its own lane, here, the ego vehicle can perform lane changes as well. Therefore, a more sophisticated prediction of other traffic participants is required. To achieve comfort, a long-term trajectory is generated for the ego vehicle by considering the most likely behavior of other traffic participants. When the long-term trajectory is no longer feasible, a precomputed emergency maneuver that accounts for every possible behavior of the other traffic participants is engaged.

The main idea is to design a three-step motion planner that accounts for changes in the environment and at the same time, maintains comfortable driving. First, the most likely maneuver of each relevant surrounding vehicle is computed. Then, an optimal trajectory of the ego vehicle is generated for a given time horizon T_{opt}, so that no collision occurs according to the assumed behavior of surrounding vehicles, as illustrated in Fig. 4.4. In the second step, an emergency maneuver is generated that can bring the ego vehicle to a safe state (see Fig. 4.4). To guarantee safety, all possible future maneuvers of surrounding vehicles must be considered. To this end, an overapproximative occupancy set which encloses all possible occupancies is computed (as previously described in Chapter 3) for a given time horizon T_{emg}.

We denote the first part of the optimal trajectory computed for a given time interval $[t, t + t^*]$ as ⓐ (see Fig. 4.4(a)). Then, we generate a collision-free emergency maneuver ⓑ, such that the fail-safe trajectory determined by the concatenation of ⓐ and ⓑ does not intersect with the occupancy set of the surrounding vehicles, for any intermediate time interval up to the time horizon T_{emg}. Thus, for a given time horizon, no matter the trajectories of the surrounding vehicles, there exists an emergency maneuver that can safely bring the ego vehicle to a safe state, as depicted in Fig. 4.4(a). Here, an optimal control-based method is used to generate the emergency maneuver, however, any other approach that can guarantee the occupancy sets of the surrounding vehicle are not violated, can be used as well. Lastly, after the current traffic scenario is updated and new measurements are collected, the decision of whether the ego vehicle can still follow the optimal long-term trajectory is made, or else the precomputed emergency maneuver is engaged to avoid a possible collision.

To summarize, we first predict the most likely maneuvers of other traffic participants. Then, an optimal long-term trajectory is generated for the ego vehicle such that the corresponding occupancy sets of the ego vehicle, do not intersect with the occupancy sets associated with the most likely prediction of the surrounding vehicles. However, the long-term plan is not followed unless a fail-safe emergency maneuver is guaranteed to exist such that the ego vehicle will not collide with the surrounding vehicles. To check whether there exists a safe emergency maneuver, we compute the overapproximative sets of the other traffic participants, and verify if there is an intersection between these sets and the occupancy corresponding to the emergency maneuver of the ego vehicle. The general architecture of the presented



(a) $t = t_\mathrm{i}$

(b) $t = t_\mathrm{i} + t^*$

Figure 4.4: At each time step, an emergency maneuver that accounts for every possible maneuver of the leading vehicle is available. If there is no other further trajectory available, the emergency maneuver is applied, which can safely bring the ego vehicle to a standstill.

approach is illustrated in Fig. 4.5. In the following subsections, we describe the generation of the optimal and emergency trajectories in more detail.

4.4.1 Optimal Trajectory Generation

Since the long-term, optimal trajectory is being followed only after it is verified as safe (recall the online verification concept of a given trajectory in Fig. 3.2), formal predictions for the other traffic participants are not required for planning of this trajectory. Instead, we use the most likely maneuver of the relevant surrounding vehicles, as a non-formal, long-term prediction. Various approaches for computing the most probable trajectory of the leading vehicle already exist, for example, by assuming constant yaw rate and acceleration (CYRA) [24] or by using a maneuver recognition module (MRM) [75]. Since MRM shows higher accuracy compared to CYRA for a longer time horizon prediction [75], the MRM approach is subsequently used to generate the most probable trajectory for each surrounding vehicle. Nevertheless, any non-formal approach that predicts the most likely behavior of the surrounding vehicles can be used as well.

In [75], the goal is to generate a trajectory prediction based on the detection of the target lane, i.e. the lane towards which a vehicle is driving. Three basic maneuvers are considered: keep lane, change lane, and turn. Obviously, other possible maneuvers can be seen as a combination of those basic maneuvers. It is assumed that for any maneuver execution, the target position of a vehicle is along the center-line of a lane. Then, to compute the most likely trajectory of a surrounding vehicle, a comparison between the current path of the vehicle and the center-line of a given lane is performed. The prediction of the most likely trajectory is computed for each time t_i over a given time horizon T_{opt}. For more details, the reader is referred to [75]. Since the target architecture of the proposed framework is highly modular (see Fig. 4.5), the approach used for the detection of the most likely maneuver of the surrounding vehicles can be easily interchanged with another appropriate method. For each traffic participant, a polygon that represents the corresponding non-formal predicted occupancy is associated with each position of the computed trajectory $\Gamma(V_{\square}, t_i)$, for each time instance $\forall t_i,\ 0 < t_i \leq T_{\text{opt}}$. The predicted polygons of the surrounding vehicles are then embedded as constraints in the trajectory generation problem of the ego vehicle, as will be subsequently described.

Next, a long-term, optimal trajectory that considers the non-formal prediction of other traffic participants is generated for the ego vehicle, for a given time horizon T_{opt}. Generating trajectories that utilize optimal control or MPC, which must satisfy a set of given constraints, is already a mature research field. In this chapter, an approach inspired by [194] is used to generate the trajectory of the ego vehicle. Any other trajectory planning approach can be used as well (see the related approaches in Sec. 4.1). The goal of the work described in [194] is collision avoidance through velocity reduction. On the other hand, in this thesis, the aim is to generate a smooth trajectory that avoids high jerk values and that tracks

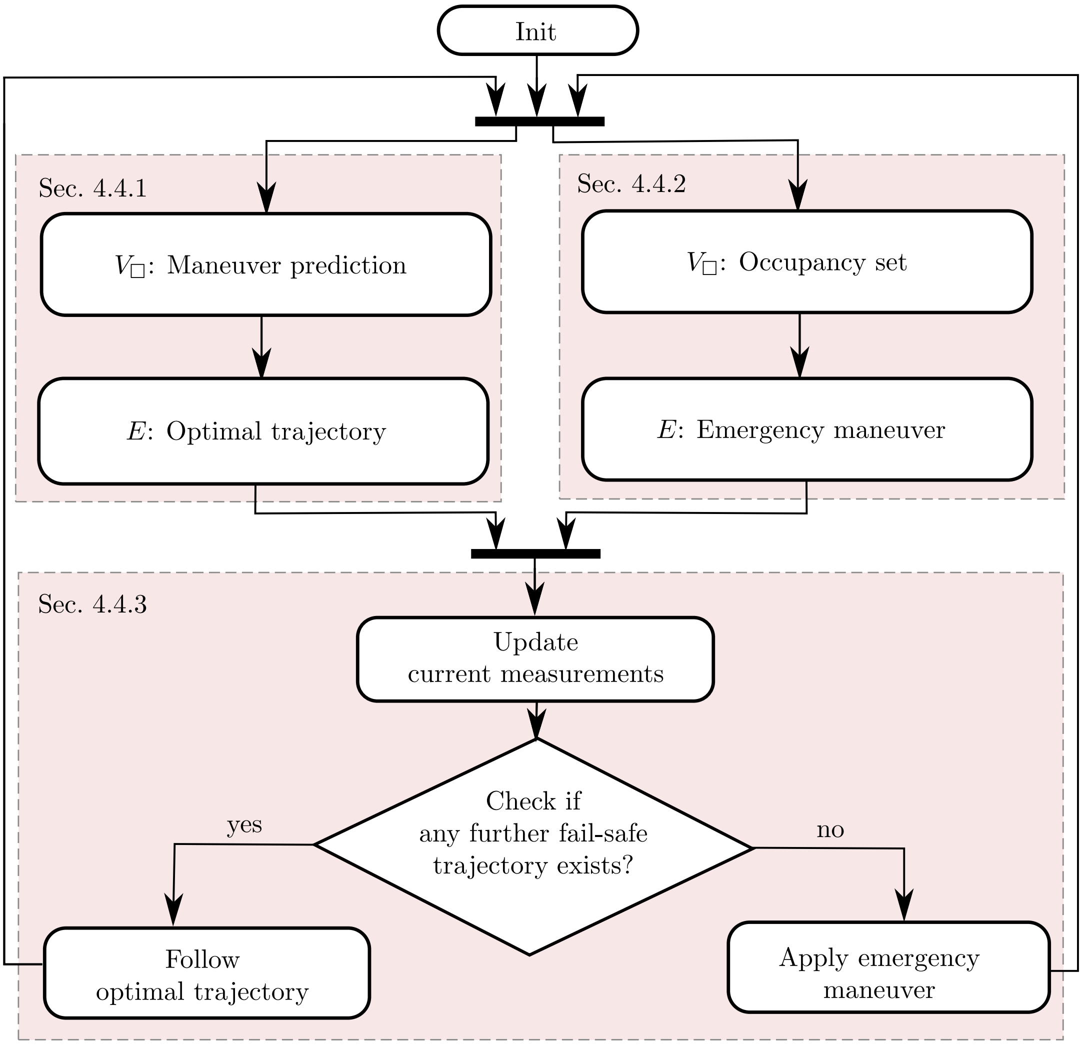

Init
Sec. 4.4.1
$V_\square$: Maneuver prediction
E: Optimal trajectory
Sec. 4.4.2
$V_\square$: Occupancy set
E: Emergency maneuver
Sec. 4.4.3
Update current measurements
Check if any further fail-safe trajectory exists?
yes
no
Follow optimal trajectory
Apply emergency maneuver

Figure 4.5: General architecture of the proposed approach for fail-safe motion planning.

the given reference path. Recall that we assume that a route planner is available to the ego vehicle, which provides the relevant reference path that must be followed in order to reach the navigation goal. Thus, the cost function from [194] is modified accordingly by penalizing any deviation from the reference trajectory (here, the reference trajectory is considered as the centerline of the target lane).

We denote by $\Gamma(V_\square, t_i)$ the occupancy sets corresponding to the most likely maneuver of the surrounding vehicle $V_\square$ at the time step t_i. To generate a trajectory for the ego vehicle that avoids collisions with the surrounding vehicles, we introduce constraints regarding the distance between the occupancy corresponding to the generated trajectory of the ego vehicle and the future predicted occupancy sets $\Gamma(V_\square, t_i)$ corresponding to the other traffic participants. Here, the minimum Euclidean distance d_i between the occupancy $\Gamma(E, t_i)$, which encloses the ego vehicle, and the predicted occupancy $\Gamma(V_\square, t_i)$, is given as (see Fig. 4.6):

$$d_{\mathrm{i}} = \min_{i} \; distance(\Gamma(E, t_i),\ \Gamma(V_{\square}, t_i)), \tag{4.13}$$

where both occupancy polygons $\Gamma(E, t_i)$ and $\Gamma(V_{\square}, t_i)$ are computed for each time step t_i.

To prevent collisions with the other traffic participants, the following constraint is set:

$$d_{\mathrm{i}} \geq d_{\min},\ \forall t_i,\ 0 < t_i \leq T_{\mathrm{opt}}, \tag{4.14}$$

where $d_{\min}$ is a given parameter. Of course, more sophisticated safe distance measures can be embedded here as well for both the longitudinal and lateral directions[1]. However, in this chapter, a simple parameter-based approach is used for both longitudinal and lateral directions, in order to focus on the novel aspects of the proposed framework.

Finally, the control inputs — steering rate u_1 and acceleration u_2 — are calculated such that the proposed cost function J_{opt} is minimized. Here, the cost function J_{opt} penalizes variations of the steering rate, acceleration, and steering angle, together with the deviation of the heading and position with respect to the reference trajectory[2]:

$$J_{\mathrm{opt}} = \int_{t}^{t+T_{\mathrm{opt}}} \Big[\gamma_1 u_1^2 + \gamma_2 u_2^2 + \gamma_3(\theta - \theta_r)^2 + \gamma_4 \delta^2 + \gamma_5 d_r^2\Big] d\tau,$$
$$\text{subject to: } (4.3), (4.8) - (4.12), (4.13),$$

where θ_r is the orientation of the reference trajectory, d_r is the distance to the reference trajectory, and $\gamma_i, i \in \{1, \ldots, 5\}$ are weighting parameters corresponding to each term of the cost function.

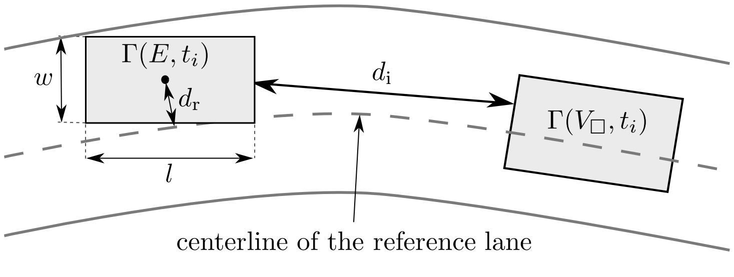


Figure 4.6: Obstacle avoidance constraint corresponding with time instance t_{i}.

[1] The approach for calculating the safe distance for the ACC setup could be used for the longitudinal direction. In contrast, a parameter-based approach may be more appropriate for the lateral direction.

[2] Here, the reference trajectory is considered the centerline of the desired lane. It is assumed that there exists a tactical planner that decides which is the desired lane.

4.4.2 Emergency Trajectory

The long-term, optimal trajectory does not consider all possible behaviors determined by infinitely many maneuvers[1] of the surrounding vehicles, but only the most probable one. However, not considering all possible maneuvers of the other traffic participants may result in an imminent collision. That is, the ego vehicle may get into a traffic situation where an emergency maneuver could only mitigate the collision effects.

In Chapter 3, we have presented a method for computing an overapproximative occupancy prediction for the other traffic participants that encloses all possible trajectories for a given time horizon. Recall that the overapproximative sets are calculated for a given time horizon T_{emg}, for consecutive time intervals τ_i, $\tau_i \leq T_{\text{emg}}$, for each surrounding vehicle. To this end, different abstracted models of the surrounding vehicles are considered. These models consider the constraints derived from the traffic rules listed in the Vienna Convention on Road Traffic [176] and physical constraints. Then, a collection of overapproximative reachable sets is calculated for each time interval τ_i, with respect to each considered abstracted model. In [6], it is proven that the intersection of the reachable sets corresponding to different abstracted models provides the overapproximative occupancy of the real model of other traffic participants. For a more detailed description, the reader is referred to [6, 7] and Chapter 3.

There is much research on emergency trajectory generation. However, a crucial issue common to the previous work on the generation of emergency maneuvers is that not all possible trajectories of the surrounding relevant vehicles are considered. Most of the previous work assumes that the surrounding vehicles are moving with constant acceleration and constant yaw rate or only considers static obstacles. Note that, even if a more sophisticated approach for predicting the future behavior of the other traffic participants is used, some unexpected maneuvers may still not be considered.

Similar to the previous step (see Sec. 4.4.1), after the prediction of the other vehicles is computed, a collision-free trajectory is generated for the ego vehicle. The major differences introduced by computing the emergency maneuver instead of an optimal trajectory are that the velocity must be reduced, and corresponding predicted occupancy set of the surrounding vehicles must be avoided.

To generate the emergency maneuver, an optimal control-based approach similar to the one presented in [194] is used. To guarantee safety, the predicted occupancy sets of other traffic participants are embedded in the constraint function. The cost function J_{emg} is similar to the one used for optimal trajectory generation, as described in (4.4.1). The difference is that driving along a reference trajectory is no longer desired, but rather minimizing the velocity v_{ego}:

[1] These infinitely many maneuvers are, of course, limited by the physical capabilities of the system.

$$\begin{aligned} J_{\text{emg}} = &\int\limits_{t}^{t+T_{\text{emg}}} \Big[\gamma_1 u_1^2 + \gamma_2 u_2^2 + \gamma_3 \delta^2 + \gamma_4 v_{\text{ego}}^2\Big] d\tau, \\ &\text{subject to: } (4.3), (4.8) - (4.12), \\ &\qquad\qquad \Gamma(E, t) \subset \text{lanes} \backslash Occ(V_{\square}, \tau). \end{aligned} \tag{4.15}$$

4.4.3 Maneuver Selection

In the previous two sections, Sec. 4.4.1 and Sec. 4.4.2, an optimal trajectory and an emergency maneuver are generated for the ego vehicle, considering the most probable future behavior and the predicted over-approximative occupancy sets of the surrounding vehicles, respectively, as illustrated in Fig. 4.4. The next step is to decide which maneuver must be applied at each time step to guarantee safety and ensure comfort.

After new measurements are received[1] (see Fig. 4.5), it is evaluated whether a dangerous traffic situation has occurred, and the precomputed emergency maneuver should be executed, or the optimal trajectory is still safe (with respect to the updated traffic scenario). If several feasible trajectories are found, the optimal one is chosen. Otherwise, if no further collision-free trajectory exists, then the previously computed emergency maneuver is engaged. In order to check whether the safety of the long-term optimal trajectory has been affected due to the updated traffic situation, let us first define what a safe trajectory represents.

Let us denote by $\mathcal{X}_C = \mathcal{X}_A \mathcal{X}_B$ the concatenation of trajectories $\mathcal{X}_A = [x_A(t_0), x_A(t_1), x_A(t_2), \ldots,$ $x_A(t_A)]$, $\mathcal{X}_B = [x_B(t_0), x_B(t_1), x_B(t_2), \ldots, x_B(t_B)]$, where t_A and t_B represent the time horizon corresponding to each trajectory, $\mathcal{X}_A$ and $\mathcal{X}_B$, respectively, such that the concatenated trajectory is $\mathcal{X}_C = [x_A(t_0),$ $x_A(t_1), x_A(t_2), \ldots, x_A(t_A), x_B(t_0), x_B(t_1), x_B(t_2), \ldots, x_B(t_B)]$.

Definition 4.1. [Safe trajectory] A given trajectory $\mathcal{X}_{\textcircled{a}}$ is verified as *safe* if a further emergency trajectory $\mathcal{X}_{\textcircled{b}}$ exists, which is concatenated with the previous one, can steer the ego vehicle into a safe state (e.g., standstill). The second condition that must be fulfilled is that the corresponding occupancy of both trajectories $\Gamma(\mathcal{X}_{\textcircled{a}})$ and $\Gamma(\mathcal{X}_{\textcircled{b}})$ must not intersect with the corresponding occupancy prediction sets of the surrounding vehicles, i.e.,

[1] New measurements with respect to the other traffic participants are assumed to be available to the ego vehicle at each time instance, through onboard sensors.

$$\begin{aligned}
&\mathcal{X}_{\text{safe}} = \mathcal{X}_{\textcircled{a}}\mathcal{X}_{\textcircled{b}}, \\
&\text{subject to} \quad \forall k \in \{1, \cdots, N\}\ \forall r \in \{0, \cdots, i-1\}, \\
&\qquad\qquad\quad \forall j \in \{i, \cdots, n\}, \exists \mathcal{X}_{\text{emg}}(t_i) : \\
&\qquad\qquad\quad \Gamma(\mathcal{X}_{\textcircled{a}}(\tau_r)) \cap Occ(V_k, \tau_r) = \varnothing \quad \wedge \\
&\qquad\qquad\quad \Gamma(\mathcal{X}_{\textcircled{b}}(\tau_j)) \cap Occ(V_k, \tau_j) = \varnothing.
\end{aligned} \tag{4.16}$$

If the ego vehicle follows the trajectory determined by concatenation of $\mathcal{X}_{\textcircled{a}}$ and $\mathcal{X}_{\textcircled{b}}$ (see Fig. 4.4), no collision will occur since the available emergency maneuver can safely bring the ego vehicle to a safe state, even if the surrounding vehicles perform unexpected maneuvers. To decide if the pre-computed emergency maneuver must be engaged, we first verify if there exists any further fail-safe trajectory at each time step t_i. Finally, we define the trajectory segment $\mathcal{X}_{\textcircled{a}}$=$\mathcal{X}_{\text{opt}}[t_0, t^*]$ as being safe if:

$$\forall t \in [t_0, t^*]\ \mathcal{X}_{opt}(t) \cap Occ(V_k, t) = \varnothing \wedge \forall t \in [t^*, T_{emg}]\ \mathcal{X}_{emg}(t) \cap Occ(V_k, t) = \varnothing \tag{4.17}$$

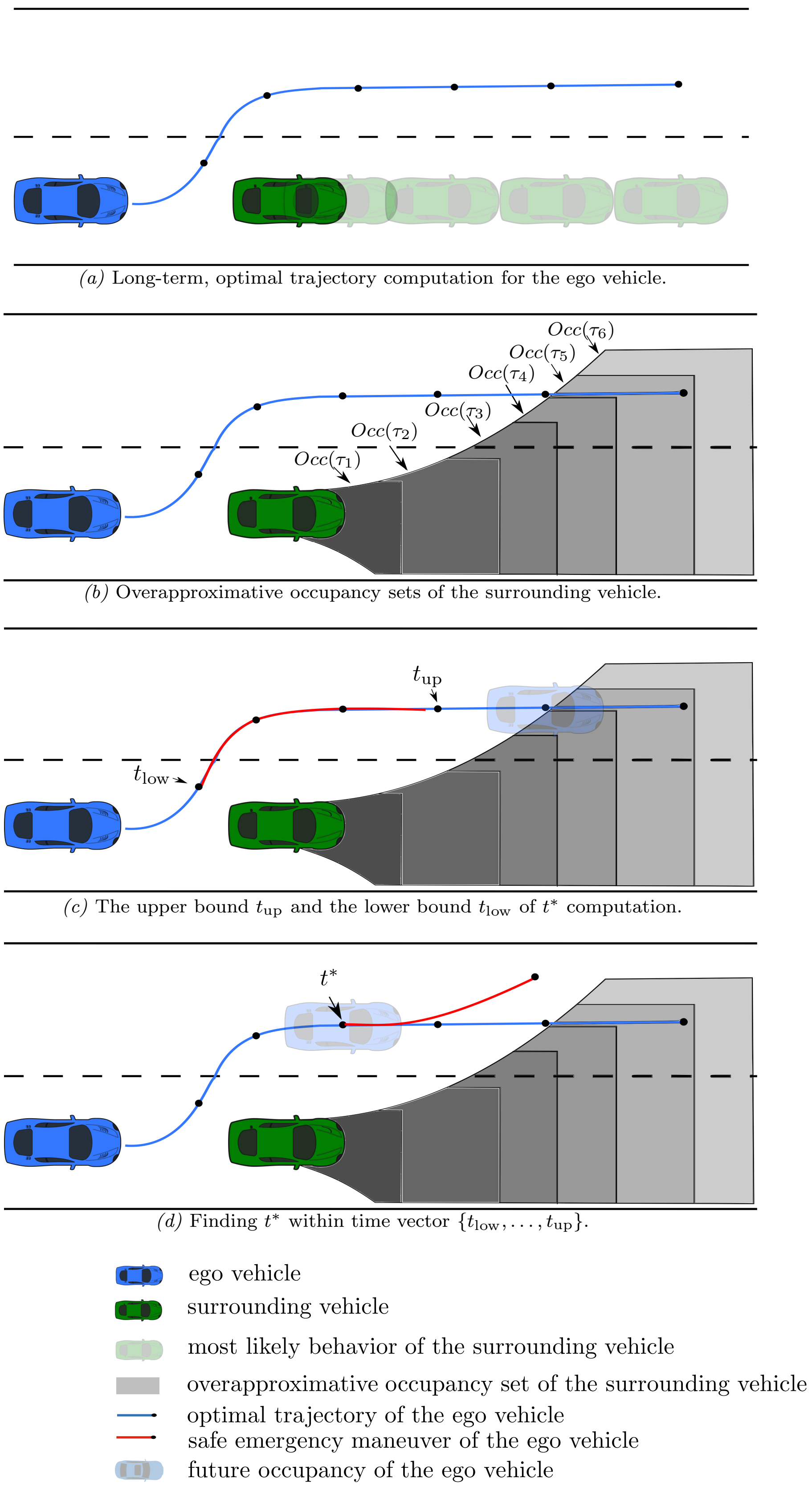
where the value of t^* can be initially set to Δt.

4.5 Computing the Maximum Time Horizon t^* to Safely Follow a Trajectory

In the previous section, we presented a method that verifies whether a given trajectory can be safely followed. Next, instead of verifying at every time step whether the state of the trajectory to be followed is safe, we determine the maximum time horizon t^* to safely follow the given trajectory, as formulated in (4.2).

To calculate the aforementioned maximum safe time horizon t^*, we propose a four-step algorithm, as illustrated in Fig. 4.7. In Sec. 4.4, we already solved the first two steps, that is, generating both a long-term trajectory $\mathcal{X}_{\text{opt}}(t)$ based on the most likely trajectory of other traffic participants, and an emergency maneuver $\mathcal{X}_{\text{emg}}(t)$ calculated by considering the overapproximative occupancy sets of other traffic participants (see Fig. 4.7a and Fig. 4.7b). Since it is neither desirable, nor required, to generate an emergency maneuver at each time step, we first prune the time interval bounded by the current time step t_0 and the end of the planning horizon T_{opt}. The resulting bounded time interval is determined by a lower and upper bound, between which it is guaranteed that an emergency maneuver exists.

The upper bound t_{up} of the time interval $[t_0, T_{\text{opt}}]$ represents the latest time for which the ego vehicle can follow the long-term trajectory without intersecting the corresponding occupancy set of the other vehicles. However, there is no guarantee that an emergency maneuver exists that can bring the ego vehicle to a safe state at time t_{up}. Thus, all states within the long-term trajectory corresponding to $t > t_{\text{up}}$ are not safe, since an intersection between the occupancies of the intended trajectory and of other traffic


$Occ(\tau_6)$
$Occ(\tau_5)$
$Occ(\tau_4)$
$Occ(\tau_3)$
$Occ(\tau_2)$
$Occ(\tau_1)$
t_{up}
t_{low}
t^*
ego vehicle
surrounding vehicle
most likely behavior of the surrounding vehicle
overapproximative occupancy set of the surrounding vehicle
optimal trajectory of the ego vehicle
safe emergency maneuver of the ego vehicle
future occupancy of the ego vehicle

(a) Long-term, optimal trajectory computation for the ego vehicle.

(b) Overapproximative occupancy sets of the surrounding vehicle.

(c) The upper bound t_{up} and the lower bound t_{low} of t^* computation.

(d) Finding t^* within time vector $\{t_{\text{low}}, \dots, t_{\text{up}}\}$.

Figure 4.7: Computation of the maximum time horizon t^* to safely follow a trajectory.

participants exists. The lower bound t_{low} represents the latest time at which full braking can be initiated so that standstill is reached before or at t_{up}. Both t_{up} and t_{low} are illustrated in Fig. 4.7c. Computing the lower and the upper time, t_{low} and t_{up}, corresponds to the third step within the proposed algorithm.

Finally, in the fourth step, the maximum time horizon t^* is calculated using binary search [99] within the interval $[t_{\text{low}}, t_{\text{up}}]$ (see Fig. 4.7d). In the following, a detailed description of every step of the proposed algorithm (as illustrated in Fig. 4.8) is provided, such that the problem described in (4.2) is solved.

4.5.1 Computation of the Upper Bound t_{up}

The upper bound t_{up} represents the latest possible time for which the planned trajectory of the ego vehicle does not intersect with the corresponding occupancy prediction of the surrounding vehicles, which is described as follows:

$$\begin{aligned} t_{\text{up}} \;=\; & \max_{0 \le i \le m} \quad t_i, \\ \text{subject to} \quad & \forall k \in \{1, \cdots, p\}, \forall r \in \{0, \cdots, i-1\}: \\ & \Gamma(x_{\text{opt}}(\tau_r)) \cap Occ_{\text{k}}(\tau_r) = \varnothing. \end{aligned} \tag{4.18}$$

To calculate t_{up}, we iteratively check for collisions for each time step Δt starting at t_0. If no intersection exists at the current time step, we further check for $t_{\text{up}} := t_{\text{up}} + \Delta t$ until $t_{\text{up}} = T_{\text{opt}}$. If the entire time horizon is reached at $t_{\text{up}} = T_{\text{opt}}$ and no collision is found, each value within the time interval $[t_0, T_{\text{opt}}]$ is a possible candidate for the maximum time t^*. However, to further prune this time interval, a lower bound is later computed, as presented in Sec. 4.5.2.

Moreover, if no further emergency maneuver is found and thus, $t_{\text{up}} := t_0$, the current trajectory is not verified as safe, and the precomputed emergency maneuver must be engaged, as illustrated in Fig. 4.8.

4.5.2 Computation of the Lower Bound t_{low}

The lower bound t_{low} represents the latest time the ego vehicle can initiate an emergency maneuver along the optimal trajectory, such that it can safely stop before t_{up}. Thus, the value of t_{low} already corresponds to a safe state. In comparison, the state corresponding to t_{up} has no safety guarantees, since one cannot guarantee that an emergency maneuver exists at that point in time. Although the lower bound t_{low} is safe, it is not necessarily the optimal value of t^*, that is, the last safe emergency maneuver could be found at a later point in time $t_{\text{low}} \le t^* \le t_{\text{up}}$. Thus, every state $x_{\text{opt}}(t), t_0 \le t \le t_{\text{low}}$ is safe, since there exists at least one safe emergency maneuver starting at t_{low}. Hence, those states can be discarded from the search interval of t^*.

To find t_{low}, we generate the optimal velocity profile backwards in time along the given path (the optimal trajectory $\mathcal{X}_{\text{opt}}$). For this, we assume that $v(t_{\text{up}}) = 0$. The generated optimal velocity profile guarantees minimal time when traveling along the given path starting from a standstill at the position

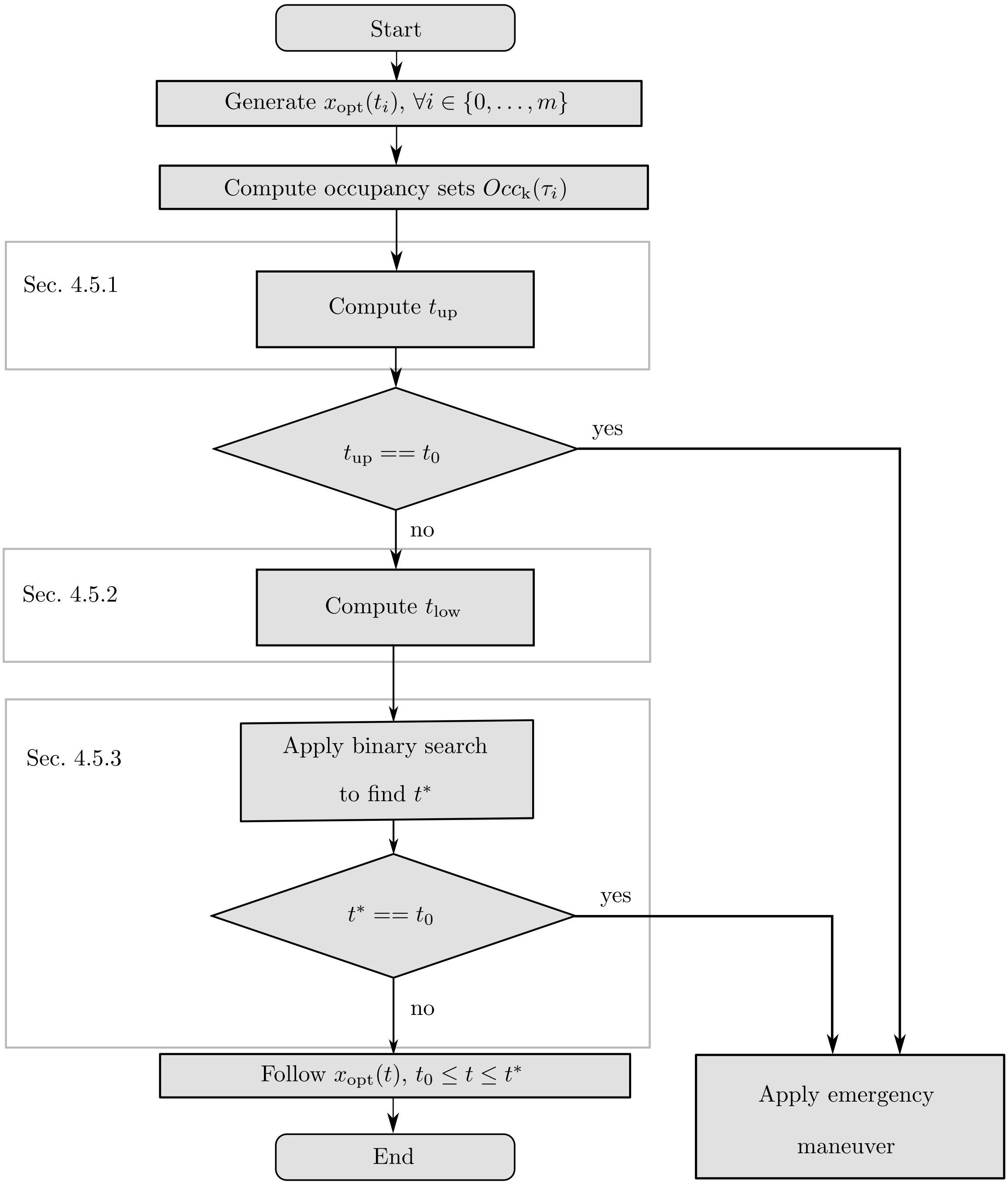

Start
Generate $x_{\text{opt}}(t_i)$, $\forall i \in \{0, \ldots, m\}$
Compute occupancy sets $Occ_{\text{k}}(\tau_i)$
Sec. 4.5.1
Compute t_{up}
$t_{\text{up}} == t_0$
yes
no
Sec. 4.5.2
Compute t_{low}
Sec. 4.5.3
Apply binary search to find t^*
$t^* == t_0$
yes
no
Follow $x_{\text{opt}}(t)$, $t_0 \leq t \leq t^*$
End
Apply emergency maneuver

Figure 4.8: Algorithm for computing the maximum time horizon t^* to safely follow a trajectory.

of $x_{\text{opt}}(t_{\text{up}})$, towards the position of $x_{\text{opt}}(t_0)$. To generate this optimal velocity profile that provides the shortest trip time for curved paths, the method presented in [187] is used. For a path with negligible

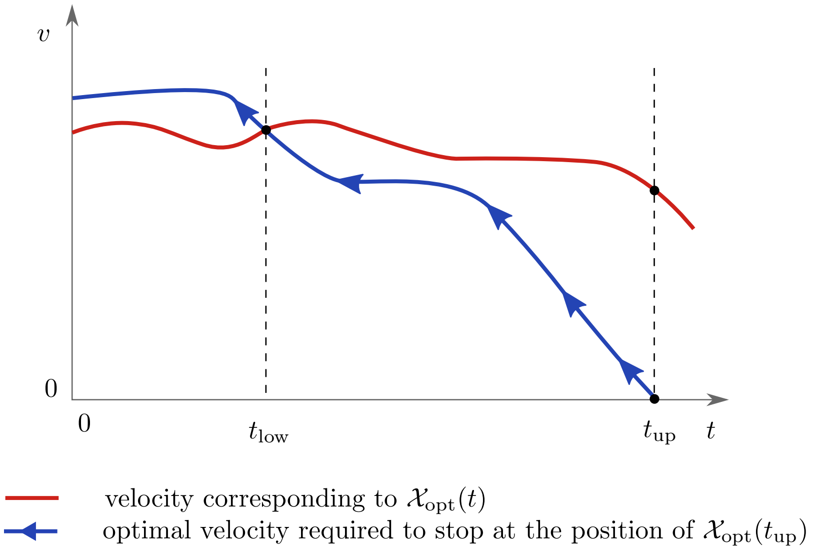


Figure 4.9: Computation of t_{low}.

curvature, full deceleration a_{min} would provide the fastest emergency maneuver. Finally, the lower bound t_{low} represents the time where the velocity profile of the optimal trajectory x_{opt} intersects with the optimal velocity for decelerating to a standstill at the position of $x_{\mathrm{opt}}(t_{\mathrm{up}})$ as shown in Fig. 4.9.

4.5.3 Binary Search of t^*

After the interval around t^* is pruned to $[t_{\mathrm{low}}, t_{\mathrm{up}}]$, we search for the maximum time within which a feasible emergency maneuver exists. To find t^* within the interval $[t_{\mathrm{low}}, t_{\mathrm{up}}]$, any search algorithm can be used. Here, a binary search [99, Sec. 6.2.1] is applied due to its efficiency ($\mathcal{O}(log\ n)$ complexity) compared to the sequential search ($\mathcal{O}(n)$ complexity).

The search interval is denoted by $t_{\mathrm{searchVect}} = [t_{\mathrm{low}}, t_{\mathrm{low}} + \Delta t, \cdots, t_{\mathrm{up}}]$. Let us now introduce the operator idx(e, v), which provides the *index of the element* e within a vector v. We denote by $\lfloor \cdot \rfloor$ the *floor operator*, which provides the greatest integer less than or equal to the given parameter.

Subsequently, in Alg. 3, the computation of t^* is explained in detail. First, an initialization is done by adding a label $\mathit{visit}(t_i) \leftarrow 0$, which is a Boolean variable, for each element $t_i \in t_{\mathrm{searchVect}}$. An element $t_i \in t_{\mathrm{searchVect}}$ is assigned the label $\mathit{visit}(t_i) \leftarrow 1$ if an emergency maneuver starting with that element is verified whether or not to exist.

To check if an emergency maneuver exists, an optimal control-based method already described in Sec. 4.4.2 is used. The considered objective function to be minimized is provided in (4.15). To solve (4.15), we apply Sequential Quadratic Programming (SQP) [144], since this method provides an efficient solution of constrained nonlinear optimization problems.

Algorithm 3 Search of t^* within $[t_{\text{low}}, t_{\text{up}}]$.

```
Input: $t_{\text{up}},\ t_{\text{low}},\ \Delta t,\ timeVect$
Output: $t^*$
forall $t_i \in t_{searchVect}$ do
    $visit(t_i) \leftarrow 0$
while $true$ do
    $i_{\text{low}} \leftarrow idx(t_{\text{low}}, t_{\text{searchVect}})$
    $i_{\text{up}} \leftarrow idx(t_{\text{up}}, t_{\text{searchVect}})$
    $i_{\text{mid}} \leftarrow \left\lfloor \frac{i_{\text{low}} + i_{\text{up}}}{2} \right\rfloor$
    $t_{\text{mid}} \leftarrow t_{\text{searchVect}}[i_{\text{mid}}]$
    if $\exists(\mathcal{X}_{emg}(t_{mid}))$ then
        $s \leftarrow idx(t_{\text{mid}} + \Delta t, t_{\text{searchVect}})$
        if $visit(t_s) == 1$ then
            $t^* \leftarrow t_{\text{mid}}$
            break
        else
            $t_{\text{low}} \leftarrow t_{\text{mid}}$
            $t_{\text{temp}} \leftarrow t_{\text{searchVect}}\left[\left\lfloor \frac{i_{\text{low}} + i_{\text{up}}}{2} \right\rfloor\right]$
            $visit(t_{\text{temp}}) \leftarrow 1$
    else
        $s \leftarrow idx(t_{\text{mid}} - \Delta t, t_{\text{searchVect}})$
        if $visit(t_s) == 1$ then
            $t^* \leftarrow t_{\text{mid}} - \Delta t$
            break
        else
            $t_{\text{up}} \leftarrow t_{\text{mid}}$
            $t_{\text{temp}} \leftarrow t_{\text{searchVect}}\left[\left\lfloor \frac{i_{\text{low}} + i_{\text{up}}}{2} \right\rfloor\right]$
            $visit(t_{\text{temp}}) \leftarrow 1$
return $t^*$
```

4.6 Numerical Experiments

We demonstrated our proposed approach for fail-safe motion planning by using real traffic data. To this end, detailed vehicle trajectories are used, that are taken from a dataset that is part of the Federal

Highway Administration's (FHWA) Next Generation Simulation (NGSIM)[1] project, as already described in Sec. 3.5.3. We continuously plan a fail-safe motion accounting for all possible behaviors of other traffic participants (captured by our set-based prediction, as described in Chapter 3) to guarantee collision avoidance. In this demonstration, we use a standstill as a safe final state of the fail-safe trajectory.

Table 4.1: Parameters used for fail-safe motion planning simulations.

Parameter (unit)	Δt (s)	l (m)	w (m)	v_{ch} (m/s)
Value	0.1	4	2	20

4.6.1 Simulation Results: Fail-Safe Motion Planning

In our simulation setup, the vehicles whose trajectories were recorded are considered to be the surrounding vehicles. The ego vehicle is positioned behind the surrounding vehicle(s) and the initial velocity and acceleration are arbitrarily chosen within the given limits. In Table 4.1, the parameters used for the generation of fail-safe trajectories are given. In Table 4.2, the admissible ranges of variables are listed.

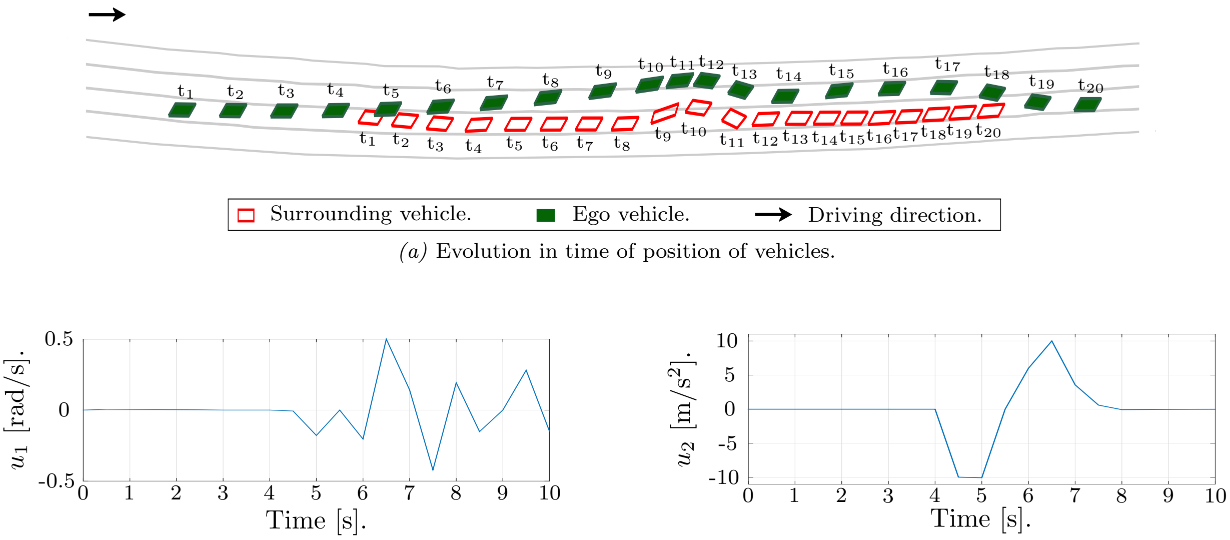
Table 4.2: Ranges of variables used for fail-safe motion planning simulations.

Parameter (unit)	v (m/s)	δ (rad)	$\dot{\delta}$ (rads)	a (m/s^2)
Interval	[0,30]	[-π/2,π/2]	[-π/8, π/8]	[-10,10]

Next, we consider three traffic scenarios based on the NGSIM dataset, as follows: In the first scenario, only one surrounding vehicle is considered and in the second and third investigated scenarios, multiple surrounding vehicles are considered.

Scenario 1. Here, a scenario with only one surrounding vehicle is considered. Initially, both the ego and the other vehicle are travelling in the same lane at a 37m apart. The initial velocity for the ego and the lead vehicle is 20m/s and 13.5m/s, respectively. At time 4.5s (t_9), the other vehicle begins to perform an unexpected maneuver and steers towards the left lane, where the ego vehicle is driving in, at that time. However, at each time step, a feasible emergency maneuver is available for the ego vehicle, which considers all possible maneuvers of the surrounding vehicle. The unexpected behavior of the other traffic participant triggers an emergency maneuver for the ego vehicle, such that a possible collision is avoided, as depicted in Fig. 4.10a. The inputs used to control the ego vehicle are presented in Fig. 4.10. The control inputs u_1 and u_2 are large due to the fail-safe maneuver, which is only executed in an emergency situation.

[1] http://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm



(a) Evolution in time of position of vehicles.

(b) Steering rate u_1.

(c) Acceleration u_2.

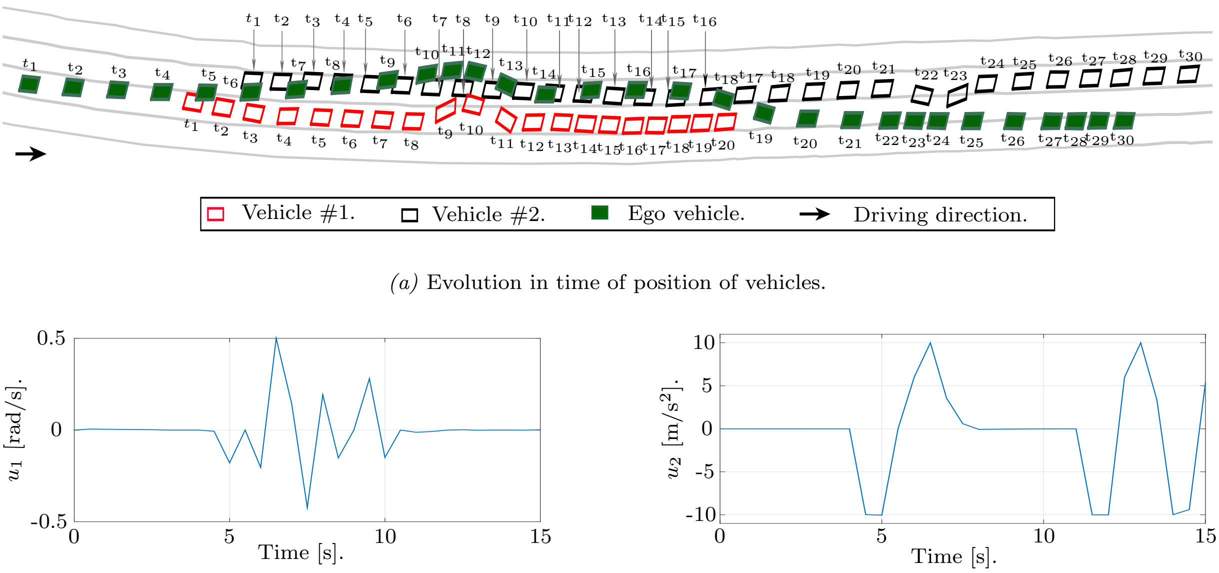
Figure 4.10: Scenario 1. Simulation results.

Scenario 2. In this scenario, two surrounding traffic participants are taken into account. The scenario considers the initial distances between the ego vehicle and the surrounding vehicle as 37m and 49m, and the initial velocities of other vehicles as 13.5m/s and 13m/s, respectively. Fig. 4.11a shows the measured position of the surrounding vehicles at each time step, together with the generated path of the ego vehicle. The values of control inputs u_1 and u_2 are presented in Fig. 4.11.

The lead vehicle #1 (see Fig. 4.11a) performs an unexpected maneuver at time t_9 towards the left lane, where the ego vehicle is driving. The ego vehicle successfully avoids the collision by applying the available emergency maneuver. Next, at time t_{22} the lead vehicle #2 starts a lane change maneuver. At the next time step, the lane change maneuver is aborted, and the ego vehicle continues driving along the planned trajectory.

Scenario 3. The basis of the third scenario originates from the US 101 dataset with two surrounding vehicles (vehicle #1 and vehicle #2) and the ego vehicle, as shown in Fig. 4.12. The initial velocities of vehicle #1, vehicle #2, and the ego vehicle are are 12.0 m/s, 15.9 m/s, and 20 m/s, respectively. The time horizons considered for the occupancy prediction and the long-term trajectory are $t_{short} = 0.5$ s, $t_{long} = 5$ s, respectively. The simulation of the whole scenario with continuous updates of the occupancy prediction is performed for 10 s. Our intended trajectories consist of the first 0.2 s of the long-term trajectory followed by the planned fail-safe trajectory. Our objective function for the long-term plan penalizes the distance from the center of the lane and the variation of the yaw angle whereas the objective function for the fail-safe trajectory penalizes the velocity.

In order to create a more challenging scenario, we partially modified the recorded motion of vehicle #1,



(a) Evolution in time of position of vehicles.

(b) Steering rate u_1.

(c) Acceleration u_2.

Figure 4.11: Scenario 2. Simulation results.

while the recorded motion of vehicle #2 is unchanged. The first modification occurs at time step 2 when vehicle #1 suddenly steers towards the left lane and quickly returns to its original lane (see Fig. 4.12). However, the verification procedure of the ego vehicle determines that the long-term trajectory continues to be safe so that it is continued. At time step 8, the second modification occurs: vehicle #1 again steers towards the left lane and quickly returns to its original lane which causes a dangerous situation. This time, the ego vehicle must execute its fail-safe trajectory to avoid a potential collision. The fail-safe trajectory is executed until a new safe trajectory is generated for the ego vehicle. This demonstration shows the benefit of keeping a fail-safe trajectory based on a set-based prediction of other traffic participants available.

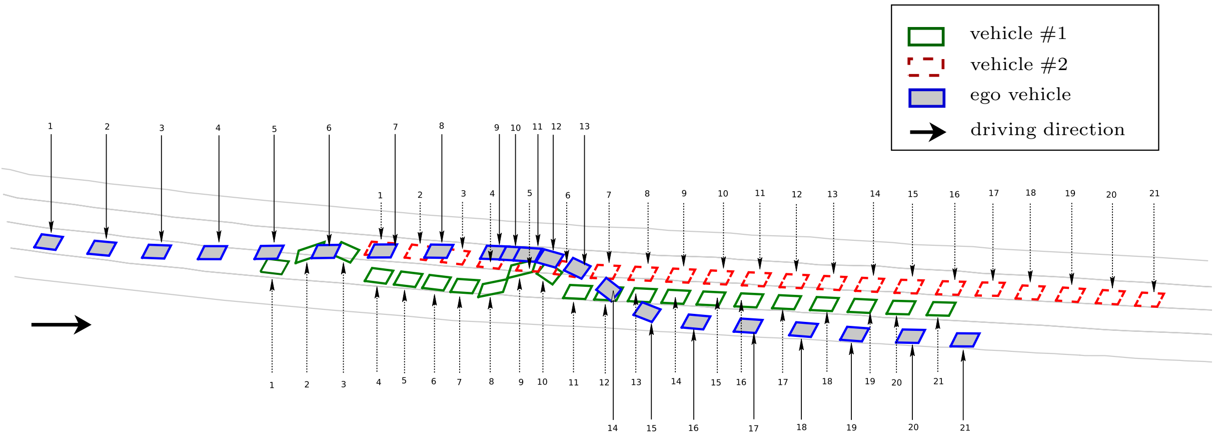
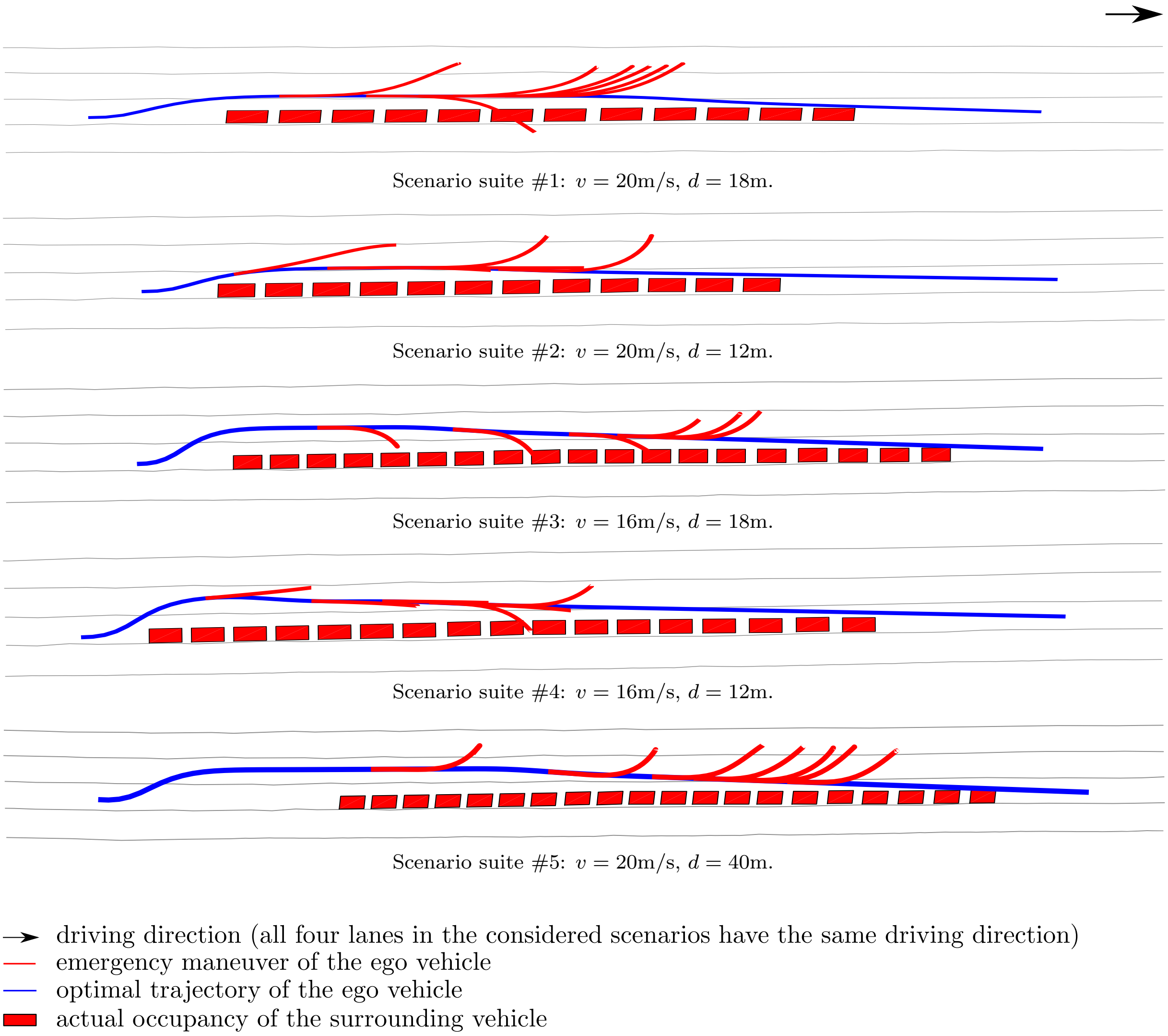


Figure 4.12: Scenario 3. Fail-safe motion planning using predicted occupancy sets; the numbers represent the time steps t_i.



Scenario suite #1: $v = 20\text{m/s}$, $d = 18\text{m}$.

Scenario suite #2: $v = 20\text{m/s}$, $d = 12\text{m}$.

Scenario suite #3: $v = 16\text{m/s}$, $d = 18\text{m}$.

Scenario suite #4: $v = 16\text{m/s}$, $d = 12\text{m}$.

Scenario suite #5: $v = 20\text{m/s}$, $d = 40\text{m}$.

→ driving direction (all four lanes in the considered scenarios have the same driving direction)
— emergency maneuver of the ego vehicle
— optimal trajectory of the ego vehicle
🟥 actual occupancy of the surrounding vehicle

Figure 4.13: Simulation results for scenario suites #1 - #5.

4.6.2 Simulation Results: Maximum Safe Time Horizon

4.6.2.0.1 Simulation setup. Six scenario suites are analyzed to validate our approach, that is, to calculate the latest point in time where an emergency maneuver exists (so-called maximum safe time horizon). The first five suites consider only one surrounding vehicle, whereas the last one considers three surrounding vehicles. We start by selecting an initial frame from the data set and mark it as the initial simulation time. Then, we extract the corresponding information regarding the other traffic participants at each future time step t_i.

The constraints on the velocity, steering rate, and acceleration, as well as the length, width, and characteristic velocity used for the ego vehicle, are shown in Table 4.1. Additionally, the weighting parameters used in the optimization problem are set to the following values: $\gamma_1 = 80$, $\gamma_2 = 10$, $\gamma_3 = 1$,

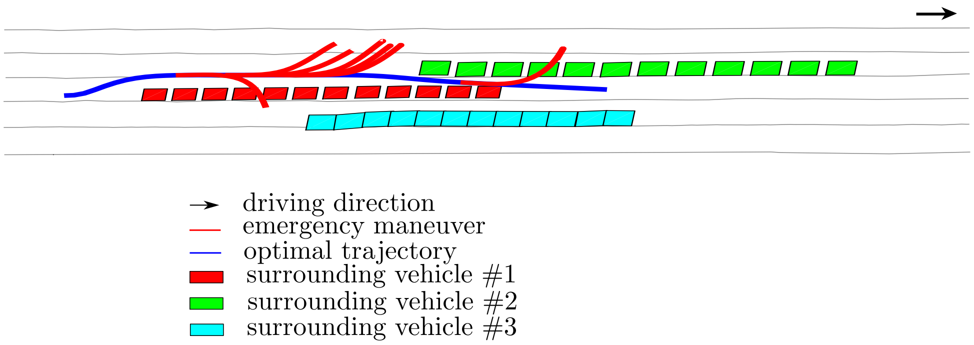


Figure 4.14: Simulation result for scenario suite # 6.

and $\gamma_4 = 0$. The initial velocity v of the ego vehicle and the initial smallest distance d to any surrounding vehicle are defined by the user. In the simulation, different initial values of v and d are considered. In scenario suite #1 and #2, v is set to 20m/s and d is 18m and 12m, respectively, while in scenario suite #3 and #4, v is set to 16m/s and d is 18m and 12m. The initial velocity of the surrounding vehicle is 12.19m/s in scenario suites #1 − #4. In scenario suite #5, v is 20m/s and d is 40m.

In scenario suite #6, the ego vehicle has an initial velocity $v = 20$m/s. We obtain the initial velocities of surrounding vehicles #1, #2, and #3 from the NGSIM dataset, which are 12.19m/s, 14.62m/s, and 10.92m/s, respectively. The distance between the ego vehicle and the closest surrounding vehicle is 18m. Fig. 4.13 and 4.14 show the simulation results for each considered scenario suite. For illustration purposes, the position of the surrounding vehicle is shown only for every 0.5 seconds, and not for every Δt. Here, to generate a long-term plan for the ego vehicle, an RRT-based method [104] is applied. Other trajectory generating algorithms can be used as well.

Simulation Results. Fig. 4.15 illustrates the computed values of t^* for each scenario suite, which represents the latest point in time when an emergency maneuver would be required. Taking scenario suite #3 as an example, the ego vehicle can safely follow the long-term trajectory over the following time interval [0, 2] since an emergency maneuver is kept available at $t_1^* = 2$s. Meanwhile, new measurements are collected, and a new safe interval is computed as [2, 3.4]. A safe emergency maneuver is pre-computed after $t_2^* = 1.4$s. Considering the new acquired information regarding the behavior of the other traffic participants, we compute a new safe time horizon, as well as an evasive maneuver starting at the end of the computed time interval, which guarantees to bring the ego vehicle to a safe state.

If we would have used the approach from our previous work [120], a total of 95 emergency maneuvers would have been generated, one for each time step. However, with the method proposed in this chapter, only 7 emergency maneuvers are needed for the same scenario suite. Therefore, our algorithm is indeed capable of significantly reducing the required number of emergency maneuvers. Similarly, in scenario

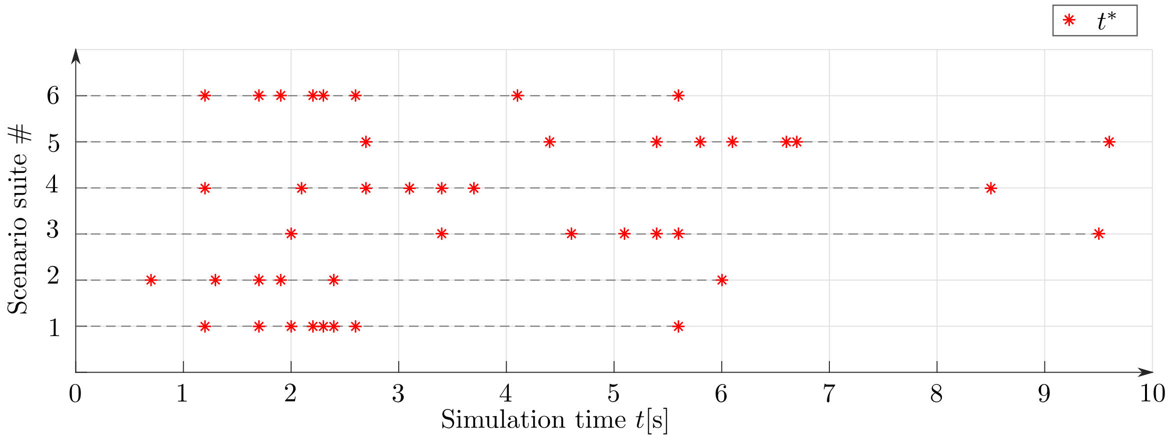


Figure 4.15: Values of t^* for the considered scenarios suite. After each t^*, the current traffic scenario is updated and we check if the current optimal trajectory is still feasible. If it is not, and there exists no other new optimal trajectory, the precomputed emergency maneuver is engaged until a new optimal trajectory is found. Otherwise, a new t^* is computed, during which the ego vehicle can safely follow the optimal trajectory.

suite #6, we only need to generate 8 emergency maneuvers, while by using our previous approach, it would have required 56 maneuvers, since one maneuver is generated at each time step.

In conclusion, the simulation results demonstrate that our algorithm can indeed automatically adapt the frequency of generating emergency maneuvers, based on the current traffic situation (e.g., distance to the surrounding vehicles and relative velocity), which can be seen when comparing scenario suites #1 to #2 and #3 to #4. When the ego vehicle gets closer to a surrounding vehicle, t^* becomes smaller. More frequent emergency maneuvers are generated in order to account for the future possible behavior of other traffic participants. Scenario suite #5 shows this more obviously — when the ego vehicle is approaching the surrounding vehicle in the longitudinal direction, emergency maneuvers are generated more often, to account for a possible dangerous situation.

4.7 Summary

Guaranteeing safety and comfort for motion planning in a dynamic environment is a major challenge due to the uncertainties introduced by the infinite number of possible trajectories of other traffic participants. However, most of the time, comfort and safety are contradictory requirements. While much work already exists on both emergency maneuver and optimal trajectory generation, motion planning that simultaneously considers safety and comfort has not yet been addressed.

In this chapter, a fail-safe motion planning approach for autonomous vehicles is presented, that considers both comfort and safety. The optimality of the generated trajectory is achieved by considering the most likely trajectory of the surrounding vehicles. Moreover, safety is guaranteed by keeping an emergency maneuver available that accounts for every possible trajectory of the surrounding vehicles over a given time horizon. Thus, the main asset of our technique is that we are able to bring the ego vehicle to a safe state, no matter what the future maneuvers of other traffic participants may be.

Since keeping available an emergency maneuver at each time step is computationally expensive, and often not required, an algorithm for computing the maximum time horizon during which a vehicle can safely follow a given trajectory is proposed. Instead of generating emergency maneuvers at each time step, we determine the maximum time horizon during which the ego vehicle can follow a trajectory without need of applying an emergency maneuver. To truncate the search interval for finding this maximum time horizon, a lower and an upper bound are computed. After the search interval is pruned, we apply binary search to find the maximum time horizon for which a given trajectory can be safely followed. At the end of this time horizon, only one emergency maneuver is generated.

The fail-safe motion planning approach was tested using real traffic data, and it shows that safety can indeed be achieved by considering all possible maneuvers of the surrounding vehicles. Furthermore, the frequency of the generated emergency maneuvers is highly reduced, while safety is still guaranteed. The simulation results show that our approach performs well under different traffic scenarios by adapting the frequency of generating emergency maneuvers according to the current traffic situation.

Chapter 5

Concluding Remarks

Lastly, a summary of this thesis is presented. First, the major contributions introduced in each chapter are highlighted. Then, the possible future research directions conclude this thesis.

5.1 Summary and Contributions

In this thesis, we have motivated the need for new methods and decision algorithms for motion planning of autonomous vehicles that can formally guarantee safety. Although motion planning, in general, is a mature research field, the existing planning techniques for autonomous vehicles in mixed traffic scenarios cannot guarantee safety with respect to the other traffic participants.

There are two major reasons for this problem. On the one hand, when generating a motion plan for an autonomous vehicle, only a limited set of the most likely behaviors of the other traffic participants is taken into consideration. Therefore, in the situation that a misinterpretation of the future behavior of other vehicles occurs, the ego vehicle may end up in an unsafe situation. On the other hand, when a dangerous traffic situation occurs, an emergency maneuver should be generated. However, it is possible at that point in time that no further emergency maneuver capable of avoiding a possible collision exists; instead, the generated maneuver would just decrease the impact of the collision. Collision mitigation is its own research field with the main goal of minimizing the collision results, rather than preventing hazardous situations from occurring. These issues have been already identified in self-driven prototype vehicles [190], but also in vehicles equipped with driving assistance systems[1].

Therefore, in this thesis, we propose a method for motion planning of autonomous vehicles that can formally guarantee safety, by considering all possible future behaviors of other traffic participants and keeping available an emergency maneuver that can bring the controlled vehicle to a safe state.

[1] https://www.wabco-auto.com/americas_en/Our-Solutions/Bus-solutions/Bus-Safety/Advanced-Driver-Assistance-Systems/OnGuard-Collision-Mitigation-System-Bus

Safe Longitudinal Control of Autonomous Vehicles

The objective of this chapter is to provide a framework that generates longitudinal control input for an autonomous vehicle in a traffic scenario where maintaining the current lane is the desired maneuver. Although there already exists systems that assist a human-driver with longitudinal driving functions (e.g.: cruise control or adaptive cruise control), there is a distinct lack of formal safety guarantees. That is, those existing systems cannot prove safety in an unexpected traffic situation. They can, at most, activate a mitigation maneuver, and hand further responsibility over to the human-driver. For vehicles with a higher level of autonomy, however, the system takes over the driving responsibility from the driver. Therefore systems that can formally guarantee safety are required.

The primary contribution in this chapter is represented by a novel control scheme that consists of a nominal controller, which is supervised by an emergency controller that is activated once the nominal controller is no longer feasible. To generate an emergency maneuver that can bring the ego vehicle into a safe state, we first calculate the required safe distance with respect to the vehicle driving ahead. To do this, we not only consider the most likely behavior of the leading vehicle, but the worst-case situation where a full braking maneuver is engaged. Therefore, even in this case, the precomputed emergency maneuver can guarantee that the ego vehicle is brought to a safe state.

In a typical leading-follower setup (e.g.: ACC systems), the ego vehicle (the follower) keeps a safe distance only to the vehicle driving ahead. However, in the situation that the closest leading vehicle is performing a lane change due to a slower vehicle or a static obstacle ahead, the ego vehicle must be prepared to react as well, to avoid a potential collision. To guarantee that we can still find an emergency maneuver that is able to bring the ego vehicle to a safe state, we not only calculate a safe longitudinal input corresponding to the closest leading vehicle, but consider all traffic participants that can affect the safe distance of the ego vehicle.

Another traffic scenario where the safe distance may be violated is represented by cut-in vehicles aiming for the inter-vehicle gap between the ego vehicle and its closest leading vehicle, but also in a general case, where a vehicle driving in an adjacent lane is performing a lane change towards the ego vehicle's lane. First, in order to have a smoother response to a cut-in maneuver, the ego vehicle should react before the lane change is actually completed. Therefore, when a cut-in maneuver is predicted, the calculation of a clearance acceleration has already begun, and the safe distance could be restored in a specified time horizon.

Finally, since a control input is calculated for each relevant vehicle, we select the one which does not jeopardize the safe distance with respect to the other traffic participants. Therefore, we select the acceleration command that produces the most defensive behavior to achieve safety corresponding to all relevant surrounding obstacles.

Overapproximative Occupancy Set Computation of Traffic Participants

The drawback of the framework presented in the previous chapter is that only the longitudinal motion problem is addressed. For more complex traffic scenarios (e.g. the ego vehicle performs a lane change maneuver), however, it is not enough to guarantee safety only with respect to the vehicles driving in the ego lane. Therefore, a more complex prediction of the behavior of other traffic participants is required. In previous work, this problem was solved by generating either the most likely behavior — determined by one or more possible maneuvers or by calculating the probability distribution of the future occupancy of other traffic participants over time. Since the prediction generated by these approaches does not consider all possible behaviors of the other traffic participants, the generated trajectory of the ego vehicle cannot be guaranteed to be collision-free.

In this chapter, we address the above-mentioned issue. Instead of calculating only a set of the most likely maneuvers, we calculate consecutive overapproximative occupancy sets that contain all possible behaviors of surrounding traffic participants, for a given time interval.

This chapter is an extension of the work published by Matthias Althoff in [7, 9], where it has been shown that the overapproximative occupancy sets representing the prediction of other traffic participants can be calculated in a formal manner. This is computed by intersecting the occupancy corresponding to different abstractions of the vehicle model. In the previous work, however, only single lane roads were considered. The main contribution presented in this chapter is represented by a framework that can formally calculate the predicted occupancy sets of surrounding traffic participants on arbitrary road networks.

Another improvement presented in this chapter represents an algorithm that calculates in an efficient way an overapproximation of the fastest way through a lane, whereas in the previous work, the shortest path through a lane was assumed to exist.

Finally, we have validated our approach for computing overapproximative occupancy prediction sets against real traffic data in order to show that our prediction always encloses the actual occupancy resulting from the recorded trajectories.

Fail-Safe Motion Planning

After an efficient algorithm to calculate the overapproximative occupancy sets of other traffic participants was presented in the previous chapter, here, the main focus is to include the computed prediction sets into the planning algorithm, such that we can guarantee safety.

Our proposed method for fail-safe motion planning of the ego vehicle is similar to the one presented in Chapter 2 and consists of two main steps. First, a long-term, optimal trajectory is calculated considering the most likely behavior of the other traffic participants. This trajectory, however, cannot be guaranteed to be collision-free. Moreover, when a dangerous situation is detected, an emergency maneuver may not exist anymore during replanning. To avoid this, before following the generated optimal trajectory, we check if it is verified as safe. We define an intended trajectory as safe if there exists another trajectory corresponding to an emergency maneuver that, concatenated with the intended trajectory, can bring the ego vehicle to a safe state, no matter what the future behavior of the other traffic participants is.

In order to generate the trajectory corresponding to the emergency maneuver, we use the predicted overapproximative occupancy sets of the other traffic participants as constraints, such that any possible collision can be avoided, even when the surrounding vehicles perform unexpected maneuvers.

Therefore, an emergency maneuver starting at each state of the optimal trajectory is generated and kept available. This precomputed maneuver is engaged, however, only if no further safe emergency maneuver is found. Safety is thus guaranteed, since the previous emergency maneuver accounts for all possible behaviors of the surrounding traffic participants, by incorporating the overapproximative predicted sets as constraints.

Nevertheless, keeping an emergency maneuver available at each time instance can be computationally expensive and most of the time it is not required. The second contribution presented in this chapter represents an algorithm that calculates the maximum time horizon the ego vehicle can safely follow a trajectory with the guarantee that at the end of this time horizon an emergency maneuver that can bring the ego vehicle to a safe state exists.

5.2 Possible Future Directions

In the following, the possible future directions are discussed separately for each chapter.

Safe Longitudinal Control of Autonomous Vehicles

The main asset of the proposed algorithm is to guarantee safety with respect to the other traffic participants. The switch between the nominal controller that achieves comfortable driving and the engaged emergency maneuver to ensure safety is mainly triggered by how accurate the assumptions on the behavior of the surrounding vehicles are. Therefore, the more accurate the prediction of future maneuvers is, the less frequently the emergency maneuver is engaged. Consequently, one obvious direction would be to use more sophisticated algorithms for the non-formal prediction. Lately, trajectory prediction approaches based on artificial intelligence methods [94, 114] has gained a lot of attention due to their ability to consider a variety of external factors and to cope with different levels of uncertainty.

Since selfish driving actions lead to increased traffic congestion on highways [160], another possible improvement would be to combine the proposed safe longitudinal control with cooperative motion planning, for not only trucks but also passenger vehicles.

Overapproximative Occupancy Set Computation of Traffic Participants

It is clear that the approaches mentioned above for prediction (e.g., based on artificial intelligence methods) cannot replace the formal prediction introduced in this chapter. Nevertheless, the proposed method can improve the accuracy of its results by considering the following points.

As already mentioned in the corresponding chapter, the more traffic rules are considered, the more accurate the predicted occupancy sets are. Therefore, to reduce the error introduced by the overapproximative prediction approach, one may consider adding multiple traffic rules. For example, we could integrate the safe distance calculation presented in Chapter 2 as a required traffic rule that must be kept with respect to the leading vehicle. Rules that result from markings on the road (e.g., solid lines) could further prune the resulting occupancy sets. Another future direction could consider the interaction between surrounding vehicles (started in [100]) such that traffic behavior is anticipated more accurately. This would allow the ego vehicle to plan a smoother and more comfortable trajectory.

Different traffic regulations may be considered depending on the type of vehicle the future occupancy is predicted for. Various traffic signs imply different regulations for separate categories of vehicles (e.g., truck vs. passenger vehicle). In the same direction, different sets of parameters (e.g., maximum deceleration or acceleration values) could be considered depending on the type of the vehicle, or on the perceived environment conditions (which provides the friction coefficient).

Fail-Safe Motion Planning

This chapter proposed a method for fail-safe motion planning, where an evasive maneuver is kept available, that can steer the ego vehicle to a safe state if required. The main objective of the evasive maneuver was to reduce the velocity of the ego vehicle. However, there may be situations where the best reaction to a dangerous situation is not to engage a braking maneuver, but rather to accelerate to leave the unsafe area. Hence, a possible future direction would be to integrate more sophisticated methods for generating evasive maneuvers into the proposed fail-safe motion planning framework, depending on the current traffic scenario.

In our approach, we assumed that the desired lane where the ego vehicle should drive, is available. There exists a lot of work on deciding which maneuver (keeping the current lane or changing the lane) is more beneficial, when concerning improved comfort or optimization the trip's duration. These methods, however, could also be integrated into the evasive maneuver generation to enhance overall comfort and safety. Moreover, traffic regulations such as driving in the rightmost lane, when possible [34, Section 1], could be integrated in the decision-making towards which lane the evasive maneuver should be performed.

Appendix

Subsequently, the evaluated hypothesis, the considered traffic scenarios, and the questionnaire used in the user study — which are taken from [123] — are listed.

Scenarios considered in the user study

Scenario 1. [Emergency brake of the first leading vehicle] The ego vehicle drives behind another vehicle that suddenly performs an emergency brake, until it reaches standstill.

Scenario 2. [Standstill second leading vehicle] The ego vehicle drives behind another vehicle. Due to a standing-still second leading vehicle, the first leading vehicle performs a lane change. This results in a situation where the ego vehicle must react to a static obstacle that is entering the field of view.

Scenario 3. [Aggressive cut-in maneuver] A vehicle that was initially driving on an adjacent lane performs an aggressive cut-in maneuver, aiming for the inter-vehicle gap between the ego vehicle and the leading vehicle. During the maneuver, the cut-in vehicle is also braking.

Scenario 4. [Long cut-in maneuver] This traffic scenario is similar to the previous. The difference is that the cut-in vehicle accelerates during the lane change maneuver.

Scenario 5. [Entering a traffic jam] The ego vehicle reaches the beginning of a traffic jam area.

User study questionnaire

1. How do you rate the distance to the leading vehicle: *very short — short — appropriate — large — very large* ?

2. How do you rate the safety feeling provided by the algorithm: *very low — low — appropriate — high — very high*?

3. Did you want to intervene, but in the end not do so: *yes — no*?

4. Did the algorithm slow down unnecessarily or slow down too strongly: *yes — no* ?

5. How do you rate the comfort of this algorithm: *very low — low — appropriate — high — very high*?

6. How do you assess the timing of the braking maneuver: *very early — early — appropriate — late — very late*?

The possible answers to the questions #1, #2, #5, #6 have been encoded as numerical values from 1 to 5, whereas for the questions #3 and #4 the possible answers have been encoded as 0 and 1.

Evaluated hypothesis

Hypothesis 1: By using the safe ACC the user feels at least as safe as by using the state-of-the-art ACC;

Hypothesis 2: The comfort feeling provided by the safe ACC is at least as high as by using the state-of-the-art ACC.

Bibliography

[1] M. K. Agoston. *Computer Graphics and Geometric Modeling: Implementation and Algorithms.* Springer, 2005. 60

[2] A. A. Alam, A. Gattami, and K. H. Johansson. An experimental study on the fuel reduction potential of heavy duty vehicle platooning. In *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, pages 306 – 311, 2010. 2, 11

[3] R. W. Allen, H. T. Szostak, D. H. Klyde, T. J. Rosenthal, and K. J. Owens. Vehicle dynamic stability and rollover. Final Report DOT HS 807 956, U.S. Department of Transportation, 1992. 73

[4] M. Althoff, D. Althoff, D. Wollherr, and M. Buss. Safety verification of autonomous vehicles for coordinated evasive maneuvers. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 1078–1083, 2010. 56

[5] M. Althoff and J. M. Dolan. Reachability computation of low-order models for the safety verification of high-order road vehicle models. In *Proc. of the American Control Conference*, pages 3559–3566, 2012. xi, 73, 74

[6] M. Althoff and J. M. Dolan. Online verification of automated road vehicles using reachability analysis. *IEEE Transactions on Robotics*, 30(4):903–918, 2014. 46, 48, 49, 55, 59, 61, 64, 65, 94

[7] M. Althoff, D. Heß, and F. Gambert. Road occupancy prediction of traffic participants. In *Proc. of IEEE Conference on Intelligent Transportation Systems*, pages 99–105, 2013. 6, 48, 49, 94, 111

[8] M. Althoff and B. H. Krogh. Zonotope bundles for the efficient computation of reachable sets. In *Proc. of the 50th IEEE Conference on Decision and Control*, pages 6814–6821, 2011. 48

[9] M. Althoff and B. H. Krogh. Reachability analysis of nonlinear differential-algebraic systems. *IEEE Transactions on Automatic Control*, 59(2):371–383, 2014. 111

[10] M. Althoff and S. Magdici. Set-based prediction of traffic participants on arbitrary road networks. *IEEE Transactions on Intelligent Vehicles*, 1(2):187 – 202, 2016. 1, 6, 45, 46, 50, 52, 53, 85

[11] M. Althoff and A. Mergel. Comparison of Markov chain abstraction and Monte Carlo simulation for the safety assessment of autonomous cars. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1237–1247, 2011. 47

[12] M. Althoff, O. Stursberg, and M. Buss. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In *Proc. of the 47th IEEE Conference on Decision and Control*, pages 4042–4048, 2008. 48

[13] M. Althoff, O. Stursberg, and M. Buss. Safety assessment for stochastic linear systems using enclosing hulls of probability density functions. In *Proc. of the European Control Conference*, pages 625–630, 2009. 47

[14] R. Alur, C. Coucoubetis, N. Halbwachs, T. A. Henzinger, P. H. Ho, X. Nicolin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical Computer Science*, 138:3–34, 1995. 58

[15] A. Ames, J. W. Grizzle, and P. Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *Proc. of the IEEE Conference on Decision and Control*, pages 6271 – 6278, 2014. 12

[16] S. J. Anderson, S. C. Peters, T. E. Pilutti, and K. Iagnemma. An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios. *Transactions on International Journal of Vehicle Autonomous Systems*, 8(234):190 – 216, 2010. 80, 81

[17] G. S. Aoude, V. R. Desaraju, L. H. Stephens, and J. P. How. Behavior classification algorithms at intersections and validation using naturalistic data. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 601–606, 2011. 13

[18] M. Aramrattana, T. Larsson, C. Englund, J. Jansson, and A. Nabo. Simulation of cut-in by manually driven vehicles in platooning scenarios. In *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, pages 1–6, 2017. 13

[19] M. Ardelt, C. Coester, and N. Kaempchen. Highly automated driving on freeways in real traffic using a probabilistic framework. *IEEE Transactions on Intelligent Transportation Systems*, 13(4):1576–1585, 2012. 83

[20] B. Asadi and A. Vadihi. Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time. *IEEE Transactions on Control Systems Technology*, 19(3):707–714, 2011. 2, 11

[21] D. Baehring, S. Simon, W. Niehsen, and C. Stiller. Detection of close cut-in and overtaking vehicles for driver assistance based on planar parallax. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 290–295, 2005. 13

[22] V. L. Bageshwar, W. L. Garrard, and R. Rajamani. Model predictive control of transitional maneuvers for adaptive cruise control vehicles. *IEEE Transactions on Vehicular Technology*, 53(5):1573–1585, 2004. 12

[23] C. R. Baker and J. M. Dolan. Traffic interaction in the urban challenge: Putting boss on its best behavior. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1752–1758, Sep. 2008. 83

[24] A. Barth and U. Franke. Where will the oncoming vehicle be the next second? In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 1068–1073, 2008. 46, 91

[25] K. E. Bekris, K. I. Tsianos, and L. E. Kavraki. Safe and distributed kinodynamic replanning for vehicular networks. *Mobile Networks and Applications*, 14(3):292–308, 2009. 82

[26] K. Belghith, F. Kabanza, L. Hartman, and R. Nkambou. Anytime dynamic path-planning with flexible probabilistic roadmaps. In *Proc. of the IEEE Conference on Robotics and Automation*, pages 2372 – 2377, 2006. 80, 81

[27] P. Bender, J. Ziegler, and C. Stiller. Lanelets: Efficient map representation for autonomous driving. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 420–425, 2014. 53, 84

[28] A. Bhatia and E. Frazzoli. Incremental search methods for reachability analysis of continuous and hybrid systems. In *Hybrid Systems: Computation and Control*, LNCS 2993, pages 142–156. Springer, 2004. 72, 74

[29] P. Bhattacharya and M. L. Gavrilova. Voronoi diagram in optimal path planning. In *4th International Symposium on Voronoi Diagrams in Science and Engineering (ISVD 2007)*, pages 38–47, July 2007. 80

[30] S. Bouraine, T. Fraichard, and H. Salhi. Provably safe navigation for mobile robots with limited field-of-views in dynamic environments. *Autonomous Robots*, 32(3):267–283, 2012. 48

[31] M. Brännström, E. Coelingh, and J. Sjöberg. Model-based threat assessment for avoiding arbitrary vehicle collisions. *IEEE Transactions on Intelligent Transportation Systems*, 11(3):658–669, 2010. 46

[32] A. E. Broadhurst, S. Baker, and T. Kanade. Monte Carlo road safety reasoning. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 319–324, 2005. 47

[33] J. Bruce and M. Veloso. Real-time randomized path planning for robot navigation. In *Proc. of the IEEE Conference on Intelligent Robots and Systems*, volume 3, pages 2383 – 2388, 2002. 80, 81

[34] Bundesrepublik-Deutschland. Straßenverkehrs-Ordnung. `www.stvo.de/strassenverkehrsordnung`, 2013. 3, 6, 11, 114

[35] C. F. Chung, T. Furukawa, and A. H. Göktogan. Coordinated control for capturing a highly maneuverable evader using forward reachable sets. In *Proc. of the IEEE International Conference on Robotics and Automation*, pages 1336–1341, 2006. 48

[36] A. Chutinan and B. H. Krogh. Computational techniques for hybrid system verification. *IEEE Transactions on Automatic Control*, 48(1):64–75, 2003. 48

[37] P. A. Cook. Stable control of vehicle convoys for safety and comfort. *IEEE Transactions on Automatic Control*, 52(3):526 – 531, 2007. 13

[38] D. Corona, M. Lazar, B. D. Schutter, and M. Heemels. A hybrid MPC approach to the design of a smart adaptive cruise controller. In *Proc. of the IEEE International Conference on Control Applications*, pages 231–236, 2006. 12, 13

[39] D. Corona and B. D. Schutter. Adaptive cruise control for a SMART car: A benchmark for MPC-PWA control methods. *IEEE Transactions on Control Sytems Technology*, 16(2):365–372, 2008. 12, 41

[40] I. Dagli and D. Reichardt. Motivation-based approach to behavior prediction. In *Proc. of the Intelligent Vehicles Symposium*, pages 227–233, 2002. 46

[41] T. Dang and O. Maler. Reachability analysis via face lifting. In *Hybrid Systems: Computation and Control*, pages 96–109, 1998. 48

[42] S. Danielsson, L. Petersson, and A. Eidehall. Monte Carlo based threat assessment: Analysis and improvements. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 233–238, 2007. 47

[43] Y. Dingyi, W. Haiyan, and Y. Kaiming. State-of-the-art and trends of autonomous driving technology. In *IEEE International Symposium on Innovation and Entrepreneurship*, pages 1–8, March 2018. 1

[44] A. Eidehall. Multi-target threat assessment for automotive applications. In *Proc. of the 14th Int. IEEE Conference on Intelligent Transportation Systems*, pages 433–438, 2011. 46

[45] A. Eidehall and L. Petersson. Statistical threat assessment for general road scenes using Monte Carlo sampling. *IEEE Transactions on Intelligent Transportation Systems*, 9(1):137–147, 2008. 47

[46] P. Falcone, M. Ali, and J. Sjöberg. Predictive threat assessment via reachability analysis and set invariance theory. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1352–1361, 2011. 48

[47] J. Firl, H. Stübing, S. A. Huss, and C. Stiller. Predictive maneuver evaluation for enhancement of car-to-x mobility data. In *IEEE Intelligent Vehicles Symposium*, pages 558–564, 2012. 13

[48] L. Fletcher, S. Teller, E. Olson, D. Moore, Y. Kuwata, J. How, J. Leonard, I. Miller, M. Campbell, D. Huttenlocher, A. Nathan, and F.-R. Kline. The MIT - Cornell collision and why it happened. *Journal of Field Robotics*, 25(10):775–807, Oct. 2008. 83

[49] T. Fraichard and H. Asama. Inevitable collision states. A step towards safer robots? In *Proc. of the IEEE Conference on Intelligent Robots and Systems*, pages 388 – 393, 2003. 31

[50] E. Frazzoli. *Robust Hybrid Control for Autonomous Vehicle Motion Planning.* PhD thesis, Massachusetts Institute of Technology, 2001. 80, 81

[51] E. Frazzoli, M. A. Dahleh, and E. Feron. Maneuver-based motion planning for nonlinear systems with symmetries. *IEEE Trans. on Robotics*, 21(6):1077–1091, 2005. 81

[52] E. Frew, J. Langelaan, and M. Stachura. Adaptive planning horizon based on information velocity for vision-based navigation. In *Guidance, Navigation and Control Conference*, pages 3822–3841, 2007. 82

[53] A. Furda and L. Vlacic. Towards increased road safety: Real-time decision making for driverless city vehicles. In *2009 IEEE International Conference on Systems, Man and Cybernetics*, pages 2421–2426, Oct 2009. 83

[54] A. Furda and L. Vlacic. Enabling safe autonomous driving in real-world city traffic using multiple criteria decision making. *IEEE Intelligent Transportation Systems Magazine*, 3(1):4–17, 2011. 83

[55] M. Garcia Ortiz, J. Fritsch, F. Kummert, and A. Gepperth. Behavior prediction at multiple timescales in inner-city scenarios. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 1068–1073, 2011. 13

[56] S. K. Gehrig and F. J. Stein. Elastic bands to enhance vehicle following. In *Proc. of the IEEE Conference on Intelligent Transportation Systems*, pages 597 – 602, 2001. 81

[57] S. K. Ghosh and D. M. Mount. An output-sensitive algorithm for computing visibility graphs. *SIAM Journal on Computing*, 20(5):888–910, 1991. 66

[58] T. Gindele, S. Brechtel, and R. Dillmann. A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 1625–1631, 2010. 13

[59] T. Gindele, S. Brechtel, and R. Dillmann. Learning driver behavior models from traffic observations for decision making and planning. *IEEE Intelligent Transportation Systems Magazine*, 7(1):69–79, 2015. 47

[60] A. Girard and C. Le Guernic. Efficient reachability analysis for linear systems using support functions. In *Proc. of the 17th IFAC World Congress*, pages 8966–8971, 2008. 48

[61] A. Gray, Y. Gao, T. Lin, J. K. Hedrick, H. E. Tseng, and F. Borrelli. Predictive control for agile semi-autonomous ground vehicles using motion primitives. In *Proc. of the IEEE American Control Conference*, pages 4239 – 4244, 2012. 81

[62] D. Greene, J. Liu, J. Reich, Y. Hirokawa, A. Shinagawa, H. Ito, and T. Mikami. An efficient computational architecture for a collision early-warning system for vehicles, pedestrians, and bicyclists. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):942–953, 2011. 48

[63] G. Greiner and K. Hormann. Efficient clipping of arbitrary polygons. *ACM Transactions on Graphics*, 17(2):71–83, 1998. 60

[64] T. Gu, J. Atwood, C. Dong, J. M. Dolan, and J.-W. Lee. Tunable and stable real-time trajectory planning for urban autonomous driving. In *Proc. of the IEEE Conference on Intelligent Robots and Systems*, pages 250 – 256, 2015. 81

[65] J. Halkias and J. Colyar. Interstate 80 freeway dataset. Technical report, U.S. Department of Transportation, Federal Highway Administration, 2006. 6, 7, 76

[66] K. Hauser. Adaptive time stepping in real-time motion planning. In *Springer Tracts in Advanced Robotics*, volume 68, pages 139–155, 01 2010. 82

[67] E. Hellström, M. Ivarsson, J. Åslund, and L. Nielsen. Look-ahead control for heavy trucks to minimize trip time and fuel consumption. *Control Engineering Practice*, 17(2):245–254, 2009. 11

[68] T. A. Henzinger and R. Majumdar. Symbolic model checking for rectangular hybrid systems. In *Tools and Algorithms for the Construction and Analysis of Systems*, LNCS 1785, pages 142–156. Springer, 2000. 48

[69] C. Hermes, C. Wöhler, K. Schenk, and F. Kummert. Long-term vehicle motion prediction. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 652–657, 2009. 47

[70] J. Hershberger and S. Suri. An optimal algorithm for Euclidean shortest paths in the plane. *SIAM Journal on Computing*, 28(6):2215–2256, 1999. 66

[71] D. Heß, M. Althoff, and T. Sattel. Comparison of trajectory tracking controllers for emergency situations. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 163–170, 2013. 72

[72] D. Heß, M. Althoff, and T. Sattel. Formal verification of maneuver automata for parameterized motion primitives. In *Proc. of the IEEE International Conference on Intelligent Robots and Systems*, pages 1474 – 1481, 2014. 81

[73] J. Hilgert, K. Hirsch, T. Bertram, and M. Hiller. Emergency path planning for autonomous vehicles using elastic band theory. In *Proc. of the IEEE Conference on Advanced Intelligent Mechatronics*, pages 1390 – 1395, 2003. 81

[74] L. L. Hoberock. A survey of longitudinal acceleration comfort studies in ground transportation vehicles. Technical report, University of Washington, Department of Transportation, 1976. 39

[75] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao. Vehicle trajectory prediction based on motion model and maneuver recognition. In *Proc. of the IEEE Conference on Intelligent Robots and Systems*, pages 4363 – 4369, 2013. 91

[76] M. Hoy, A. S. Matveev, and A. V. Savkin. Algorithms for collision-free navigation of mobile robots in complex cluttered environments: a survey. *Robotica*, 33(3):463–497, 2015. 80

[77] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. Maybank. A system for learning statistical motion patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:1450–1464, 2006. 46

[78] H.-P. Huang and S.-Y. Chung. Dynamic visibility graph for path planning. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2813–2818, 2004. 66

[79] P. Ioannou, Z. Xu, S. Eckert, D. Clemons, and T. Sieja. Intelligent cruise control: Theory and experiment. In *Proc. of the IEEE Conference on Decision and Control*, volume 2, pages 1885 – 1890, 1993. 2, 11

[80] P. A. Ioannou and C. C. Chien. Autonomous intelligent cruise control. *IEEE Transactions on Vehicular Technology*, 42(4):657 – 672, 1993. 2, 11

[81] P. A. Ioannou and M. Stefanovic. Evaluation of acc vehicles in mixed traffic: lane change effects and sensitivity analysis. *IEEE Transactions on Intelligent Transportation Systems*, 6(1):79–89, 2005. 14

[82] ISO. ISO 15622:2010, Inteligent transport systems – Adaptive cruise control systems – Performance requirements and test procedures. Standard, 2010. 2, 11

[83] ISO. ISO 26262-1:2011, Road vehicles – Functional safety. Standard, 2018. 3

[84] ISO. ISO/PAS 21448:2019 Road vehicles – Safety Of The Intended Functionality. Standard, 2019. 3

[85] N. Kaempchen, B. Schiele, and K. Dietmayer. Situation assessment of an autonomous emergency brake for arbitrary vehicle-to-vehicle collision scenarios. *IEEE Transactions on Intelligent Transportation Systems*, 10:678–687, 2009. 47

[86] N. Kalra and S. M. Paddock. Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability? Technical report, Santa Monica, CA: RAND Corporation, 2016. 4

[87] N. Kämpchen. *Feature-level fusion of laser scanner and video data for advanced driver assistance systems.* PhD thesis, Universität Ulm, 2007. 16

[88] D. Kasper, G. Weidl, T. Dang, G. Breuel, A. Tamke, and W. Rosenstiel. Object-oriented bayesian networks for detection of lane change maneuvers. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 673–678, June 2011. 13

[89] L. E. Kavraki, M. N. Kolountzakis, and J. C. Latombe. Analysis of probabilistic roadmaps for path planning. *IEEE Trans. on Robotics and Automation*, 14(1):166 – 171, 1998. 80, 81

[90] L. E. Kavraki, P. Svestka, J. C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. on Robotics and Automation*, 12(4):566 – 580, 1996. 80, 81

[91] H. Kazemi, H. N. Mahjoub, A. Tahmasbi-Sarvestani, and Y. Fallah. A learning-based stochastic mpc design for cooperative adaptive cruise control to handle interfering vehicles. *IEEE Transactions on Intelligent Vehicles*, pages 1–1, 2018. 14

[92] R. Kianfar, P. Falcone, and J. Fredriksson. A receiding horizon approach to string stable cooperative adaptive cruise control. In *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, pages 734 – 739, 2011. 18

[93] R. Kianfar, P. Falcone, and J. Fredriksson. Safety verification of automated driving systems. *IEEE Intelligent Transportation Systems Magazine*, 5:73–86, 2013. 46, 48

[94] B. Kim, C. M. Kang, J. Kim, S. H. Lee, C. C. Chung, and J. W. Choi. Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network. In *IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 399–404, 2017. 113

[95] J. Kim and D. Kum. Collision risk assessment algorithm via lane-based probabilistic motion prediction of surrounding vehicles. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–12, 2017. 47

[96] J.-H. Kim and D.-S. Kum. Threat prediction algorithm based on local path candidates and surrounding vehicle trajectory predictions for automated driving vehicles. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 1220 – 1225, 2015. 46

[97] K. Kim and P. R. Kumar. An MPC-based approach to provable system-wide safety and liveness of autonomous ground traffic. *IEEE Transactions on Automatic Control*, 59(12):3341–3356, Dec 2014. 81

[98] T. Kim and H.-Y. Jeong. A novel algorithm for crash detection under general road scenes using crash probabilities and an interactive multiple model particle filter. *IEEE Transactions on Intelligent Transportation Systems*, 15(6):2480–2490, 2014. 47

[99] D. E. Knuth. *The Art of Computer Programming. Volume 3. Sorting and Searching.* Addison-Wesley, 1998. 98, 100

[100] M. Koschi and M. Althoff. Interaction-aware occupancy prediction of road vehicles. In *Proc. of the 20th IEEE International Conference on Intelligent Transportation Systems*, pages 1885–1892, 2017. 113

[101] M. Koschi and M. Althoff. SPOT: A tool for set-based prediction of traffic participants. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 1679–1686, 2017. 85

[102] J. Kuffner and S. LaValle. RRT-connect: An efficient approach to single-query path planning. In *Proc. of the IEEE Conference on Robotics and Automation*, volume 2, pages 995 – 1001, 2000. 80, 81

[103] A. B. Kurzhanski and P. Varaiya. Ellipsoidal techniques for reachability analysis. In *Hybrid Systems: Computation and Control*, LNCS 1790, pages 202–214. Springer, 2000. 48

[104] Y. Kuwata, G. A. Fiore, J. Teo, E. Frazzoli, and J. P. How. Motion planning for urban driving using rrt. In *Proc. of the IEEE International Conference on Intelligent Robots and Systems*, pages 1681–1686, 2008. 106

[105] Y. Kuwata, J. Teo, G. Fiore, and S. Karaman. Real-time motion planning with applications to autonomous urban driving. *IEEE Trans. on Control Systems Techology*, 17(5):1105 – 1118, 2009. 80, 81

[106] A. Lambert, D. Gruyer, G. S. Pierre, and A. N. Ndjeng. Collision probability assessment for speed control. In *Proc. of the 11th International IEEE Conference on Intelligent Transportation Systems*, pages 1043–1048, 2008. 47

[107] S. M. LaValle. *Planning algorithms.* Cambridge University Press, 2006. 46, 80

[108] S. M. LaValle and J. J. Kuffner. Randomized kinodynamic planning. *International Journal of Robotics Research*, 20(5):378–400, 2001. 70, 72

[109] D. Lee, Y. P. Kwon, S. McMains, and J. K. Hedrick. Convolution neural network-based lane change intention prediction of surrounding vehicles for acc. In *Proc. of the IEEE Conference on Intelligent Transportation Systems*, pages 1–6, 2017. 13

[110] S. Lefèvre, C. Laugier, and J. I. nez Guzmán. Risk assessment at road intersections: Comparing intention and expectation. In *2012 IEEE Intelligent Vehicles Symposium*, pages 165–171, June 2012.

[111] S. Lefèvre, D. Vasquez, and C. Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH journal*, 1(1):1–14, 2014. 46

[112] S. Li, K. Li, R. Rajamani, and J. Wang. Model predictive multi-objective vehicular adaptive cruise control. *IEEE Transactions on Control Systems Technology*, 19(3):556 – 566, 2011. 12, 13

[113] M. Liebner, C. Ruhhammer, F. Klanner, and C. Stiller. Generic driver intent inference based on parametric models. In *Proc. of the 16th International IEEE Conference on Intelligent Transportation Systems*, pages 268–275, 2013. 46

[114] L. Lin, S. Gong, and T. Li. Deep learning-based human-driven vehicle trajectory prediction and its application for platoon control of connected and autonomous vehicles. In *The Autonomous Vehicles Symposium*, 07 2018. 113

[115] P. Liu and Ümit Özgüner. Predictive control of a vehicle convoy considering lane change behavious of the preceding vehicle. In *Proc. of the IEEE American Control Conference*, pages 4374 – 4379, 2015. 14

[116] S. Liu, L. Liu, J. Tang, B. Yu, Y. Wang, and W. Shi. Edge computing for autonomous driving: Opportunities and challenges. *Proceedings of the IEEE*, 107(8):1697–1716, Aug 2019. 1

[117] V. J. Lumelsky. On fast computation of distance between line segments. *Information Processing Letters*, 21(2):55–61, 1985. 67

[118] J. Lygeros, D. N. Godbole, and S. Sastry. Verified hybrid controllers for automated vehicles. *IEEE Transactions on Automatic Control*, 43(4):522–539, 1998. 12

[119] J. M. Maciejowski. *Predictive control with constraints.* Prentice Hall, 2002. 12

[120] S. Magdici and M. Althoff. Fail-safe motion planning of autonomous vehicles. In *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, pages 452–458, 2016. 7, 80, 82, 84, 106

[121] S. Magdici and M. Althoff. Adaptive cruise control with safety guarantees for autonomous vehicles. In *Proc. of the World Congress of the International Federation of Automatic Control*, pages 5774–5781, 2017. 6, 15, 20

[122] S. Magdici, Z. Ye, and M. Althoff. Determining the maximum time horizon for vehicles to safely follow a trajectory. In *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, pages 1–7, 2017. 7, 80, 84

[123] S. Maierhofer. Enhancement and test of a novel adaptive cruise control system with formal safety guarantees in a real driving simulator. Master's thesis, Technical University of Munich, 2018. xi, 43, 115

[124] J.-J. Martinez and C. C. de Wit. A safe longitudinal control for adaptive cruise control and stop-and-go scenarios. *IEEE Transactions on Control Systems Technology*, 15(2):246–258, 2007. 38

[125] J. Marzbanrad and N. Karimian. Space control law design in adaptive cruise control vehicles using model predictive control. *Journal of Automobile Engineering*, 225(7):870–884, 2011. 12

[126] M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner. *Autonomous Driving. Technical, Legal and Social Aspects*. Springer Open, 05 2016. 1, 3

[127] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000. 12

[128] R. Mayr and O. Bauer. Safety issues in intelligent cruise control. In *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, pages 970–975, 1999. 11

[129] A. Mehra, W.-L. Ma, F. Berg, P. Tabuada, J. Grizzle, and A. Ames. Adaptive cruise control: Experimental validation of advanced controllers on scale-model cars. In *Proc. of the IEEE American Control Conference*, pages 1411 – 1418, 2015. 12

[130] F. Meng, J. Su, C. Liu, and W. Chen. Dynamic decision making in lane change: Game theory with receding horizon. In *11th International Conference on Control*, pages 1–6, Aug 2016. 83

[131] M. M. Minderhoud and P. H. L. Bovy. Extended time-to-collision measures for road traffic safety assessment. *Accident Analysis & Prevention*, 33:89–97, 2001. 47

[132] S. Mitra, T. Wongpiromsarn, and R. M. Murray. Verifying cyber-physical interactions in safety-critical systems. *IEEE Security and Privacy*, 11(4):28–37, 2013. 46

[133] H. Morizane, H. Takenaga, Y. Kobayashi, and K. Nakamura. Cut-in vehicle recognition system. In *Proc. of the IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems*, pages 976–980, 1999. 13

[134] MSC Software, 2 MacArthur Place, Santa Ana, CA 92707. *Adams/Tire help*, April 2011. Documentation ID: DOC9805. 73

[135] W. Muenst, T. Rehder, L. Louis, and C. Icking. A novel method for providing situational awareness to longitudinal controllers in (semi-) automated vehicles. In *Proc. of the IEEE Conference on Intelligent Transportation Systems*, pages 984–989, 2016. 14

[136] G. Naus, J. Ploeg, M. V. de Molengraft, W. Heemels, and M. Steinbuch. Design and implementation of parametrized adaptive cruise control: An explicit model predictive control approach. *Control Engineering Practice*, 18:882–892, 2010. 12

[137] C. B. Neas and M. Farhood. A hybrid architecture for maneuver-based motion planning and control of agile vehicles. In *Proc. of the IFAC World Congress*, volume 18, pages 3521–3526, 2011. 81

[138] B. Németh and P. Gáspár. Design of low conflict cruise control for safety critical vehicle interactions . In *Proc. of the 19th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pages 1186–1191, 2015. 12

[139] D. C. K. Ngai and N. H. C. Yung. A multiple-goal reinforcement learning method for complex vehicle overtaking maneuvers. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):509–522, June 2011. 83

[140] J. Nilsson, J. Silvlin, M. Brannstrom, E. Coelingh, and J. Fredriksson. If, when, and how to perform lane change maneuvers on highways. *IEEE Intelligent Transportation Systems Magazine*, 8(4):68–78, winter 2016. 83

[141] J. Nilsson and J. Sjöberg. Strategic decision making for automated driving on two-lane, one way roads using model predictive control. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 1253–1258, June 2013. 83

[142] P. Nilsson, O. Hussien, A. Balkan, Y. Chen, A. Ames, J. Grizzle, N. Ozay, H. Peng, and P. Tabuada. Correct-by-construction adaptive cruise control: Two approaches. *IEEE Transactions on Control Systems Technology*, 24(4):1294–1307, 2016. 12

[143] P. Nilsson, O. Hussien, Y. Chen, A. Balkan, M. Rungger, A. Ames, J. Grizzle, N. Ozay, H. Peng, and P. Tabuada. Preliminary results on correct-by-construction control software synthesis for adaptive cruise control. In *Proc. of the IEEE Conference on Decision and Control*, pages 816 – 823, 2014. 12

[144] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer, 2006. 100

[145] S. Öncü, J. Ploeg, N. van de Wouw, and H. Nijmeijer. Cooperative adaptive cruise control: Network-aware analysis of string stability. *IEEE Transactions on Intelligent Transportation Systems*, 15(4):1527–1537, 2014. 13

[146] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on Intelligent Vehicles*, 1(1):33–55, 2016. 80

[147] J.-M. Park, D.-W. Kim, Y.-S. Yoon, H. Kim, and K.-S. Yi. Obstacle avoidance of autonomous vehicles based on model predictive control. *Proc. of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 223(12):1499–1516, 2009. 80, 81

[148] S. Park, H. Rakha, K. Ahn, and K. Moran. Fuel economy impacts of manual, conventional cruise control, and predictive eco-cruise control driving. *International Journal of Transportation Science and Technology*, 2(3):227 – 242, 2013. 11

[149] R. Parthasarathi and T. Fraichard. An inevitable collision state-checker for a car-like vehicle. In *Proc. of the IEEE International Conference on Robotics and Automation*, pages 3068–3073, 2007. 50

[150] S. Pendleton, H. Andersen, X. Du, X. Shen, M. Meghjani, Y. Eng, D. Rus, and M. Jr. Perception, planning, control, and coordination for autonomous vehicles. *Machines*, 5:6, 02 2017. 82

[151] A. Platzer and E. M. Clarke. The image computation problem in hybrid systems model checking. In *Hybrid Systems: Computation and Control*, LNCS 4416, pages 473 – 486. Springer, 2007. 51

[152] J. Ploeg, B. T. M. Scheepers, E. van Nunen, N. van de Wouw, and H. Nijmeijer. Design and experimental evaluation of cooperative adaptive cruise control. In *Proc. of the IEEE Conference on Intelligent Transportation Systems*, pages 260 – 265, 2011. 14

[153] J. Ploeg, N. van de Wouw, and H. Nijmeijer. Lp string stability of cascaded systems: Application to vehicle platooning. *IEEE Transactions on Control Systems Technology*, 22(2):786–793, 2014. 13

[154] S. Quinlan and O. Khatib. Elastic bands: Connecting path planning and control. In *Proc. of the IEEE Conference on Robotics and Automation*, volume 2, pages 802 – 807, 1993. 80, 81

[155] R. Rajamani. *Vehicle Dynamics and Control, 2nd edition.* Springer US, 2012. 14

[156] G. Rieger, J. Scheef, H. Becker, M. Stanzel, and R. Zobel. Active safety systems change accident environment of vehicles significantly - A challange for vehicle design. In *Proc. of the International Conference on Enhanced Safety of Vehicles*, 2005. 11

[157] A. Rizaldi and M. Althoff. Formalising traffic rules for accountability of autonomous vehicles. In *Proc. of the 18th IEEE International Conference on Intelligent Transportation Systems*, pages 1658 – 1665, 2015. 4, 56

[158] A. Rizaldi, F. Immler, and M. Althoff. A formally verified checker of the safe distance traffic rules for autonomous vehicles. In *NASA Formal Methods - 8th International Symposium, NFM 2016, Minneapolis, MN, USA, June 7-9, 2016, Proceedings*, pages 175–190, 2016. 22, 26, 33

[159] A. Rizaldi, J. Keinholz, M. Huber, J. Feldle, F. Immler, M. Althoff, E. Hilgendorf, and T. Nipkow. Formalising and monitoring traffic rules for autonomous vehicles in isabelle/hol. In *Integrated Formal Methods*, volume 10510 of *LNCS*, pages 50–66. Springer, 2017. 4, 56

[160] S. Robinson. The Price of Anarchy. *SIAM News*, 37(5), June 2004. 113

[161] SAE. J3016-201806 Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. Standard, 2018. 4

[162] K. Santhanakrishnan and R. Rajamani. On spacing policies for highway vehicle automation. *IEEE Transactions on Intelligent Transportation Systems*, 4(4):198 – 204, 2003. 12

[163] T. Sattel and T. Brandt. Ground vehicle guidance along collision-free trajectories using elastic bands. In *Proc. of the IEEE American Control Conference*, pages 4991 – 4996, 2005. 81

[164] C. Schmidt, F. Oechsle, and W. Branz. Research on trajectory planning in emergency situations with multiple objects. In *Proc. of the IEEE Intelligent Transportation Systems Conference*, pages 988–992, 2006. 48, 60

[165] R. Schubert. Evaluating the utility of driving: Toward automated decision making under uncertainty. *IEEE Transactions on Intelligent Transportation Systems*, 13(1):354–364, March 2012. 83

[166] L. Segel. *The Physics of Tire Traction*, chapter Tire Traction on Dry, Uncontaminated Surfaces, pages 65–98. Springer, 1974. 57

[167] M. S. Shaikh and P. E. Caines. On the hybrid optimal control problem: Theory and algorithms. *IEEE Trans. on Automatic Control*, 52(9):1587–1603, 2007. 81

[168] S. Shalev-Shwartz, S. Shammah, and A. Shashua. On a Formal Model of Safe and Scalable Self-driving Cars. *arXiv e-prints*, page arXiv:1708.06374, Aug 2017. 3, 80

[169] S. Singh. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. Technical report, National Highway Traffic Safety Administration, 2015. 1

[170] S. Sivaraman and M. M. Trivedi. Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis. *IEEE Transactions on Intelligent Transportation Systems*, 14(4):1773 – 1795, 2013. 13, 16

[171] S. Söntges and M. Althoff. Determining the nonexistence of evasive trajectories for collision avoidance systems. In *Proc. of the 18th IEEE International Conference on Intelligent Transportation Systems*, pages 956 – 961, 2015. 48

[172] T. Stanger and L. del Re. A model predictive cooperative adaptive cruise control approach. In *Proc. of the IEEE American Control Conference*, pages 1374 – 1379, 2013. 12, 13

[173] O. Stursberg, A. Fehnker, Z. Han, and B. H. Krogh. Specification-guided analysis of hybrid systems using a hierarchy of validation methods. In *Proc. of the 1st IFAC Conference on Analysis and Design of Hybrid Systems*, pages 289–295, 2003. 48

[174] N. Sumpter and A. J. Bulpitt. Learning spatio-temporal patterns for predicting object behaviour. *Image and Vision Computing*, 18:679–704, 2000. 46

[175] Z. Sun, D. Hsu, T. Jiang, H. Kurniawati, and J. H. Reif. Narrow passage sampling for probabilistic roadmap planning. *IEEE Trans. on Robotics*, 21(6):1105 – 1115, 2005. 80, 81

[176] L. M. Surhone, M. T. Timpledon, and S. F. Marseken, editors. *Vienna Convention on Road Traffic.* VDM Publishing, 2010. 56, 94

[177] P. Svestka and M. H. Overmars. Coordinated motion planning for multiple car-like robots using probabilistic roadmaps. In *Proc. of the IEEE Conference on Robotics and Automation*, volume 2, pages 1631 – 1636, 1995. 80, 81

[178] D. Swaroop, J. K. Hedrick, C. C. Chien, and P. A. Ioannou. A comparison of spacing and headway control laws for automatically controlled vehicles. *Journal of Vehicle System Dynamics*, 23(8):597–625, 1994. 12

[179] A. Tamke, T. Dang, and G. Breuel. A flexible method for criticality assessment in driver assistance systems. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 697–702, 2011. 47

[180] H. Täubig, U. Frese, C. Hertzberg, C. Lüth, S. Mohr, E. Vorobev, and D. Walter. Guaranteeing functional safety: design for provability and computer-aided verification. *Autonomous Robots*, 32(3):303–331, 2012. 48

[181] C. Tomlin, J. Lygeros, and S. Sastry. A game theoretic approach to controller design for hybrid systems. *Proceedings of the IEEE*, 88(7):949–970, 2000. 12

[182] A. Vahidi and A. Eskandarian. Research advances in intelligent collision avoidance and adaptive cruise control. *IEEE Transactions on Intelligent Transportation Systems*, 4(3):143–153, 2003. 11

[183] F. A. M.-B. M. W. B. van Arem; Riender Happee. Design and analysis of full range adaptive cruise control with integrated collision a voidance strategy. In *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, pages 308 – 315, 2016. 13

[184] J. van den Berg, D. Ferguson, and J. Kuffner. Anytime path planning and replanning in dynamic environments. In *Proc. of the International Conference on Robotics and Automation*, pages 2366–2371, 2006. 48

[185] B. Vanholme, D. Gruyer, B. Lusetti, S. Glaser, and S. Mammar. Highly automated driving on highways based on legal safety. *IEEE Transactions on Intelligent Transportation Systems*, 14(1):333–347, 2013. 48

[186] D. Vasquez, T. Fraichard, and C. Laugier. Incremental learning of statistical motion patterns with growing hidden Markov models. *IEEE Transactions on Intelligent Transportation Systems*, 10:403–416, 2009. 46

[187] E. Velenis and P. Tsiotras. Optimal velocity generation for given acceleration limits: Theoretical analysis. In *Proc. of the IEEE American Control Conference*, pages 1478–1483, 2005. 99

[188] K. Vogel. A comparison of headway and time to collision as safety indicators. *Accident Analysis & Prevention*, 35:427–433, 2003. 47

[189] C.-G. Wallman and H. Åström. Friction measurement methods and the correlation between road friction and traffic safety. In *VTI meddelande*. Swedish National Road and Transport Research Institute, 2001. 71

[190] H. Wang, Y. Huang, A. Khajepour, Y. Zhang, Y. Rasekhipour, and D. Cao. Crash mitigation in motion planning for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 20(9):3313–3323, Sep. 2019. 109

[191] J. Wei, J. M. Dolan, and B. Litkouhi. A prediction- and cost function-based algorithm for robust autonomous freeway driving. In *2010 IEEE Intelligent Vehicles Symposium*, pages 512–517, June 2010. 83

[192] J. Wei, J. M. Snider, T. Gu, J. M. Dolan, and B. Litkouhi. A behavioral planning framework for autonomous driving. In *Proc. of the IEEE Intelligent Vehicles Symposium*, June 2014. 46

[193] M. Werling, S. Kammel, J. Ziegler, and L. Gröll. Optimal trajectories for time-critical street scenarios using discretized terminal manifolds. *International Journal of Robotic Research*, 31(3):346–359, 2012. 80, 81

[194] M. Werling and D. Liccardo. Automatic collision avoidance using model-predictive online optimization. In *Proc. of the IEEE Conference on Decision and Control*, pages 6309 – 6314, 2012. 80, 81, 87, 91, 92, 94

[195] B. H. Wilson. How soon to brake and how hard to brake: Unified analysis of the envelope of opportunity for rear-end collsion warnings. Technical report, International Society of Automotive Engineers, 2001. 12

[196] A. Wu and J. P. How. Guaranteed infinite horizon avoidance of unpredictable, dynamically constrained obstacles. *Autonomous Robots*, 32(3):227–242, 2012. 48

[197] L. Xiao and F. Gao. A comprehensive review of the development of adaptive cruise control systems. *Vehicle System Dynamics*, 48(10):1167 – 1192, 2010. 11

[198] L. Xiao and F. Gao. Practical string stability of platoon of adaptive cruise control vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1184 –1194, 2011. 13

[199] Y. Xing, C. Lv, H. Wang, H. Wang, Y. Ai, D. Cao, E. Velenis, and F. Wang. Driver lane change intention inference for intelligent vehicles: Framework, survey, and challenges. *IEEE Transactions on Vehicular Technology*, 68(5):4377–4390, May 2019. 46

[200] D. Yanakiev and I. Kanellakopoulos. Nonlinear spacing policies for automated heavy-duty vehicles. *IEEE Transactions on Vehicular Technology*, 47(4):1365–1377, 1998. 12, 13, 41, 42

[201] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda. A survey of autonomous driving: Common practices and emerging technologies. *preprint arXiv:1906.05113v2*, abs/1906.05113, 2019. 1

[202] J. Zeisler, J. Cherepanov, and V. Haltakov. A driving path based target object prediction. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 316–321, 2015. 13

[203] Y. Zhang, Q. Lin, J. Wang, S. Verwer, and J. M. Dolan. Lane-change intention estimation for car-following control in autonomous driving. *IEEE Transactions on Intelligent Vehicles*, pages 1–1, 2018. 14

[204] J. Zhou and H. Peng. Range policy of adaptive cruise control vehicles for improved flow stability and string stability. *IEEE Transactions on Intelligent Transportation Systems*, 6(2):229 –237, 2005. 13

[205] J. Ziegler, P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. G. Keller, E. Kaus, R. G.Herrtwich, C. Rabe, D. Pfeiffer, F. Lindner, F. Stein, F. Erbs, M. Enzweiler, C. Knöppel, J. Hipp, M. Haueis, M. Trepte, C. Brenk, A. Tamke, M. Ghanaat, M. Braun, A. Joos, H. Fritz, H. Mock, M. Hein, and E. Zeeb. Making Bertha drive – an autonomous journey on a historic route. *IEEE Intelligent Transportation Systems Magazine*, 6(2):8–20, 2014. 54