UNIVERSITÄT ZU LÜBECK

From the Institute for Electrical Engineering in Medicine
of the University of Lübeck
Director: Prof. Dr. Philipp Rostalski

Interaction-aware Model Predictive Control for Autonomous Highway Driving

Dissertation
for Fulfillment of
Requirements
for the Doctoral Degree
of the University of Lübeck

from the Department of Computer Sciences and Engineering

Submitted by

Xiaorong Zhang

from Xi'an, China

Lübeck 2026

First referee: Herr Prof. Dr. Georg Schildbach

Second referee: Herr Prof. Dr. rer. nat. habil. Floris Ernst

Date of oral examination: May 13, 2026

Approved for printing: Lübeck, May 19, 2026

ABSTRACT

Automated driving technology has developed rapidly over the past decades, profoundly transforming traditional modes of transportation. It offers significant conveniences by replacing human drivers in simple scenarios and improves safety through driver assistance systems. However, despite these advances, achieving full automation still faces substantial challenges. One critical challenge is accurately predicting the motion of surrounding vehicles in real time and incorporating these predictions into the control of autonomous vehicles to ensure safety. The diversity in individual driving behaviors, coupled with the complexity of modeling interactions among traffic participants under uncertainties, makes this problem particularly difficult. To provide insights into these challenges, this thesis focuses on interaction-aware traffic prediction and the safe control of autonomous vehicles on highways. Highways are chosen as the study environment due to their structured lanes and lower traffic density. The aim is to establish a research foundation in this setting that can later be extended to more complex traffic environments, such as urban roads. Specifically, this thesis investigates two primary research directions to address these challenges. The first direction separates the tasks of interaction-aware traffic prediction and safe control of autonomous vehicles. The second direction aims to integrate these tasks into a unified control architecture for greater simplicity and efficiency.

In the first research direction, diverse intention-based models along with the Interacting Multiple Model Kalman Filter (IMM-KF) algorithm are employed to estimate and predict vehicle motion states while considering interactions. Possible vehicle motion maneuvers are represented by a finite number of normal scenarios. Additionally, a "worst-case" scenario is considered along with normal scenarios in a Scenario-based Model Predictive Control (SCMPC) architecture to ensure the safety of the autonomous vehicle. Moreover, to reduce the conservatism of the control strategy associated with considering the "worst-case" scenario, a new Contingency Model Predictive Control (CMPC) scheme is explored with time-varying prediction horizons and an enlarged terminal set. In the second research direction, the Minimizing Overall Braking Induced by Lane Change (MOBIL) model is employed to find out the possible lanes that vehicles may occupy based on vehicle interactions. With this lateral traffic prediction, the longitudinal states of surrounding vehicles

are modeled simultaneously with those of the autonomous vehicle within a Model Predictive Control (MPC) structure. They are determined by minimizing the collective control costs of all vehicles through dynamic interaction-aware mechanisms. All proposed control architectures are validated in a high-fidelity IPG CarMaker simulation environment. Simulation results demonstrate the effectiveness of these approaches in safely controlling autonomous vehicles while considering vehicles' interactions on highways.

The findings of this thesis provide new perspectives on addressing challenges in highway autonomous driving and establish the foundation for extending safe control strategies to more complex traffic scenarios involving interactive behaviors. This research advances the understanding of interaction modeling in multi-agent traffic environments, contributing structured approaches to represent and leverag vehicle interactions for safer autonomous control. The modular design of the proposed frameworks facilitates their integration with broader automated driving systems, such as perception and planning modules. Moreover, validation in a high-fidelity simulation environment offers a practical reference for a potential real-world implementation.

Keywords: *Interaction-aware*, *Traffic Prediction*, *Model Predictive Control*, *Safety*, *IPG CarMaker Validation*.

ZUSAMMENFASSUNG

Die Technologie des automatisierten Fahrens hat sich in den letzten Jahrzehnten rasant entwickelt und traditionelle Verkehrsmittel grundlegend verändert. Sie bietet enorme Vorteile, indem sie menschliche Fahrer in einfachen Szenarien ersetzt, und erhöht die Sicherheit durch Fahrerassistenzsysteme. Trotz dieser Fortschritte gibt es bei der vollständigen Automatisierung noch einige große Herausforderungen. Eine wichtige Herausforderung ist es, die Bewegungen der anderen Fahrzeuge in Echtzeit genau vorherzusagen und diese Vorhersagen in die Steuerung der autonomen Fahrzeuge einzubeziehen, um die Sicherheit zu gewährleisten. Die Vielfalt individueller Fahrweisen, kombiniert mit der Komplexität, Interaktionen zwischen Verkehrsteilnehmern unter Unsicherheiten zu modellieren, macht dieses Problem besonders schwierig. Um neue Einblicke in diese Herausforderungen zu gewinnen, konzentriert sich diese Arbeit auf interaktionsbewusste Verkehrsprognosen und die sichere Steuerung autonomer Fahrzeuge auf Autobahnen. Autobahnen werden als Untersuchungsumgebung gewählt, da sie über strukturierte Fahrspuren und eine geringere Verkehrsdichte verfügen. Ziel ist es, eine Forschungsgrundlage in diesem Umfeld zu schaffen, die auf komplexere Verkehrsumgebungen, wie innerstädtische Straßen, erweitert werden kann. Konkret untersucht diese Arbeit zwei zentrale Forschungsrichtungen zur Bewältigung dieser Herausforderungen. Die erste Richtung trennt die Aufgaben der interaktionsbewussten Verkehrsprognose und der sicheren Steuerung autonomer Fahrzeuge. Die zweite Richtung zielt darauf ab, diese Aufgaben in einer einheitlichen Steuerungsarchitektur zu integrieren, um größere Einfachheit und Effizienz zu erreichen.

In der ersten Forschungsrichtung werden verschiedene absichtsbasierte Modelle zusammen mit dem Interacting Multiple Model Kalman Filter (IMM-KF) Algorithmus eingesetzt, um Fahrzeugbewegungszustände unter Berücksichtigung von Interaktionen zu schätzen und vorherzusagen. Mögliche Fahrzeugmanöver werden durch eine endliche Anzahl normaler Szenarien dargestellt. Zusätzlich wird ein “worst-case” Szenario zusammen mit den normalen Szenarien in einer Scenario-based Model Predictive Control (SCMPC) Architektur berücksichtigt, um die Sicherheit autonomer Fahrze-uge zu gewährleisten. Um die mit der Berücksichtigung des “worst-case” Szenarios verbundene Vorsicht der

Steuerungsstrategie zu verringern, wird außerdem ein neuartiges Contingency Model Predictive Control (CMPC) Schema mit zeitlich variablen Vorhersagehorizonten und einem erweiterten Terminalset untersucht. In der zweiten Forschungsrichtung wird das Modell Minimizing Overall Braking Induced by Lane Change (MOBIL) verwendet, um mögliche Fahrspuren zu ermitteln, die Fahrzeuge basierend auf Interaktionen einnehmen könnten. Mit einer solchen lateralen Verkehrsp-rognose werden die longitudinalen Zustände umliegender Fahrzeuge gleichzeitig mit denen des autonomen Fahrzeugs innerhalb einer Model Predictive Control (MPC) Struktur modelliert. Diese Zustände werden durch Minimierung der kollektiven Steuerungsk-osten aller Fahrzeuge mittels dynamischer, interaktionsbewusster Mechanismen bestimmt. Die vorgeschlagenen Steuerungsarchitekturen werden in einer hochauflösenden IPG CarMaker Simulationsumgebung validiert. Die Simulationsergebnisse zeigen die Effektivität dieser Ansätze bei der sicheren Steuerung autonomer Fahrzeuge unter Berü-cksichtigung der Interaktionen von Fahrzeugen auf Autobahnen.

Die Ergebnisse dieser Arbeit bieten neue Perspektiven zur Bewältigung der Herausforderungen beim autonomen Fahren auf Autobahnen und schaffen die Grundlage für die Erweiterung sicherer Steuerungsstrategien auf komplexere Verkehrsszenarien mit interaktiven Verhaltensweisen. Diese Forschung trägt zum Verständnis der Interaktionsmodellierung in Multi-Agenten-Verkehrsumgebungen bei und liefert strukturierte Ansätze zur Darstellung und Nutzung von Fahrzeuginteraktionen für eine sicherere autonome Steuerung. Das modulare Design der vorgeschlagenen Frameworks erleichtert die Integration in umfassendere autonome Fahrsysteme, wie Wahrnehmungs- und Planungsmodule. Darüber hinaus liefert die Validierung in einer hochauflösenden Umgebung einen praktischen Bezug für eine potenzielle Umsetzung in der realen Welt.

Schlüsselwörter: *Interaktionsbewusst*, *Verkehrsprognose*, *Modellprädiktive Regelung*, *Sicherheit*, *IPG CarMaker Validierung*.

ACKNOWLEDGEMENT

This thesis presents the results of my time as a PhD student, studying at the Universität zu Lübeck (Uzl). I would like to express my gratitude to the people who joined me on this path over the last few years and contributed to my graduate career.

With immense pleasure and a deep sense of gratitude, I wish to express my sincere thanks to my supervisor, **Prof. Dr. Georg Schildbach**, Director of Autonomous System Lab (ASL), Institute for Electrical Engineering in Medicine (IME), Uzl. I truly appreciate that Georg gave me the opportunity to pursue my PhD dream. There are countless challenges throughout my studies both in research and personal life. Without his continuous encouragement and guidance, this research would not have been successfully completed. Georg is remarkably supportive of younger researchers. While he thoughtfully guides our research directions, he never restricts our freedom to explore. He consistently encourages us to pursue topics we are passionate about and has set a great example for my future research career. I still clearly remember Georg would prepare note pages for our weekly meetings. He never tired of patiently and repeatedly discussing the questions I struggled to understand. For my first paper, there were over 100 versions reviewed and supported by him. Thanks to his high standards for research quality, we often receive positive and exciting outcomes from paper submissions, which has greatly boosted my confidence in my academic work.

I am grateful to **Dr. Hossameldin Abbas**, IME, Uzl. He is always enthusiastic about helping anyone in the institute. I still remember that during my first experience as a teaching assistant, he spent many Friday afternoons going through the exercise sheets with me to ensure I was fully prepared for Monday's class. His continued support and encouragement were incredibly valuable for my teaching tasks. Hossam takes research details seriously and pursues the truth with tireless dedication. His attitude toward science and truth sets a great example for young researchers, including myself.

I express my sincere thanks to **Dr. Sahar Zeinali**. It was a great pleasure to work alongside her during her two years as a postdoctoral researcher at ASL. Sahar was always supportive of my research ideas, no matter how impractical they seemed. She generously gave her time and energy to discuss with me whenever I needed help. Beyond her academic support, Sahar has also been a trusted mentor and friend. I am especially grateful

for her kindness and encouragement during difficult moments, and I will always remember her generosity, whether it was staying late to help me with personal matters or being there when I felt frustrated in my research.

I would like to acknowledge the support provided by my colleagues, who have contributed in various ways throughout my research journey. **Maryam Nezami** has always been quick and resourceful in helping solve my technical questions. **Robin Kensbock** is enthusiastic and generous in offering support, backed by his strong scientific expertise. I am especially grateful to **Julia Sauer, Jonas Gruner, Dimitrios Karachalios, Carlos Castelar, Fabian Domberg, Ludmila Moshagen, Irina Wiebe, Ievgen Zhavzharov** for the daily lunch gatherings and the enjoyable interactions both inside and outside the office.

I wish to express my profound gratitude to **my parents, my sister, and my brother** for all the sacrifices they made during my research, as well as for their moral support and encouragement whenever I needed it.

感谢我的父母，姐姐和弟弟在我读博这几年的支持与关爱，也感谢你们一直包容我的任性与不懂事。我很抱歉让你们在我身处异国的这些年担心了。感谢姐姐弟弟在这几年里对父母的关心照顾，正是你们的支持让我安心求学。

I also want to thank myself for not giving up on my dream of pursuing a PhD. I want to tell myself that as long as I believe in my choices, move forward bravely, and follow my heart, I will reach the shore I long for.

Last but not least, I would like to thank my beloved, **Haowei wen**, for your constant encouragement and moral support along with patience and understanding. The past several years have been of great challenge for us because we have resided in different countries, separated by a seven-hour time difference. Nevertheless, these circumstances have never diminished your presence and support. You are always there whenever I need help. Our scholarly discussions have continually served as a profound source of inspiration, and your silent online companionship has provided invaluable comfort during moments of stress. I truly believe that, regardless of what the future holds, we will always be the strongest support for each other.

AI Usage Statement: I would also like to acknowledge the support of AI-based tools used during the preparation of this dissertation. ChatGPT assisted in refining the language and improving readability, primarily in Chapters 1 and 7, while DeepL was used to translate

the English abstract into German. All AI-assisted content was carefully reviewed for accuracy, and I take full responsibility for the final work.

Place: Xi'an, China

Date: 01/06/2025 **Xiaorong Zhang**

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$\bar{A}$	Compact system matrix (definition is chapter-dependent)
$\bar{a}_{\mathrm{c}}$	Longitudinal acceleration of the driver after lane change
$\bar{a}_{\mathrm{n}}$	Longitudinal acceleration of the new follower of the driver after the driver lane change
$\bar{a}_{\mathrm{o}}$	Longitudinal acceleration of the old follower of the driver after the driver lane change
Δa_{th}	Longitudinal acceleration threshold used for evaluating benefit of lane changes
$\hat{a}$	Modified acceleration
$\overline{a}$	Maximal value of acceleration
$\underline{a}$	Minimal value of acceleration
A	System matrix (definition is chapter-dependent)
a	Vehicle acceleration
a_{c}	Longitudinal accelerations of the driver before lane change
a_{n}	Longitudinal acceleration of the new follower of the driver before the driver lane change
a_{o}	Longitudinal acceleration of the old follower of the driver before the driver lane change
$\bar{B}$	Compact input matrix (definition is chapter-dependent)
B	Input matrix (definition is chapter-dependent)
b	Vehicle width
$\bar{C}$	Compact output matrix
$\tilde{c}^{(i)}$	Modified combined probability of switching from any previous mode to mode i

$c^{(i)}$ Combined probability of switching from any previous mode to mode i

Δd Minimal relative distance between two vehicles

$\underline{d}$ Safe distance between two vehicles

d Relative distance between two vehicles

d_{s} Minimal extra stopping distance

E System input vector (mode-dependent)

e_{lat} Lateral position error

F Time-varying system matrix (mode-dependent)

γ_{ref} Vector of reference flat output variables

γ Vector of flat output variables

θ Vehicle orientation angle

λ Target lane indicator, $\lambda = 1, 2, 3$

$\Delta\mu$ Different between original and modified mode probabilities

μ Probability of mode

$\tilde{\mu}$ Modified probability of mode

ν_k Vector of measurement noise variables

ξ Drivetrain efficiency factor

$\bar{\pi}_k^{(ij)}$ Modified transition probability matrix from mode i to mode j

$\pi^{(ij)}$ Transition probability from mode i to mode j

$\tilde{\pi}_k^{(ji)}$ Adaptively updated transition probability matrix from mode j to mode i

σ Steering angle

τ Time headway

ϕ Cumulative system matrix over prediction horizon

$\breve{\chi}$ Vector of contingency flat state variables

χ Vector of flat state variables

ω Vehicle orientation angular velocity

H System measurement matrix

i_g	Gear ratio
$\overline{j}$	Maximal value of jerk
$\underline{j}$	Minimal value of jerk
J	Cost value of the optimization problem
$J^\star$	Optimal value of the optimization problem
K	Kalman gain
k	Current time step
K_d	Control gain used for distance-keeping
K_v	Control gain used for velocity-tracking
K_{D}	Derivative gain
K_{P}	Proportional gain
$\tilde{L}^{(i)}$	Modified likelihood of the measurement under mode i
l	Vehicle length
$L^{(i)}$	Likelihood of the measurement under mode i
l_{lb}	Lower bound of lane
l_{ub}	Upper bound of lane
M	Total number of system motion modes
m	Vehicle's overall mass
ΔN	Minimal extra stopping horizon
$\underline{\mathrm{N}}$	Minimal stopping horizon of the EV
$\bar{P}$	Fused state error covariance estimation
$\hat{P}$	Estimation of state error covariance
$\overline{\mathrm{Pr}}(s)$	Normalized Probability of scenario s
$\mathrm{Pr}(s)$	Probability of scenario s
$\underline{P}$	Predefined scenario probability threshold
p	Vehicle position
p_{f}	Politeness factor

Q Covariance of process noise vector

$\hat{r}_{\text{ref},k}^{(i)-}$ Estimation of unknown reference variable

R Covariance of measurement noise vector

r_w Vehicle wheel radius

r_{ref} Unknown reference variable

S Innovation residual covariance

T Sampling/predicting time step

T_{dk} Decision-making time step

t_{s} Total simulation time

t_{gap} Reference time gap

$\breve{\mathbb{U}}$ Feasible set of contingency control variables/inputs

$\breve{u}$ Vector of contingency control variables/inputs

$\breve{U}$ Set of contingency control input sequences

$\mathbb{U}$ Feasible set of control variables/inputs (definition is chapter-dependent)

U Vector of control input sequences over the prediction horizon

u Vector of control variables/inputs (definition is chapter-dependent)

$U^{\star}$ Set of optimal control input sequences (definition is chapter-dependent)

$U^{(i)}$ Vector of control input sequences under control mode i over the prediction horizon

$U^{(i,j)}$ Vector of control input sequences under jth branch of control mode i over the prediction horizon

V Number of studied target vehicles

v Vehicle velocity

v_{ref} Reference velocity

W Engine/brake Torque

W_P Weight matrix used for penalizing terminal cost

W_Q Weight matrix used for penalizing stage cost

W_R Weight matrix used for penalizing control input cost

W_z Weight matrix used for penalizing the difference between the estimation and its modification of full state vector

w_k Vector of process noise variables

$\bar{x}$ Fused state estimation

$\breve{\mathbb{X}}$ Feasible set of contingency state variables

$\breve{\mathbb{X}}_f$ Terminal set of contingency state variables

$\breve{x}$ Vector of contingency state variables

$\hat{x}$ Estimation of state vector

$\mathbb{X}$ Feasible set of state variables (definition is chapter-dependent)

$\mathbb{X}^{(\text{safety},s)}$ Feasible state set under the safety constraints of scenario s

$\mathbb{X}_f$ Terminal set of state variables (definition is chapter-dependent)

x Vector of state variables (definition is chapter-dependent)

$x^\star$ Optimal solution of the optimization problem

x_{ref} Vector of reference state variables (definition is chapter-dependent)

$X^{(i)}$ Vector of state variables under control mode i over the prediction horizon

$X^{(i,j)}$ Vector of state variables under jth branch of control mode i over the prediction horizon

$\bar{y}$ Difference between full state vector estimation and its modification

Δy Innovation residual

y Vector of system measurement variables

$\breve{z}$ Modified estimation of full state vector

$\hat{z}$ Estimation of full state vector

z Vector of full state variables (combing x and r_{ref})

LIST OF TERMS AND ABBREVIATIONS

SMPC Stochastic MPC 11

SVM Support Vector Machine 9

TV Target Vehicle 8

V2V Vehicle-to-Vehicle 5

V2X Vehicle-to-Everything 5

VT Velocity Tracking 36

CHAPTER 1

Introduction

This chapter presents the research motivation, historical development, and current challenges of automated driving. It also reviews the relevant background and concludes by highlighting the thesis contributions and outlining its overall structure.

1.1 Motivation

Over the past few decades, automated driving technology has advanced rapidly, bringing transformative developments to the transportation sector [4, 5]. The implementation of automated driving technology enhances travel safety, alleviates congestion, reduces energy consumption, and improves resource efficiency while generating economic benefits [6]. Given the complexity of real-world traffic environments, significant research efforts continue to address the key challenges in automated driving to further advance the technology.

Unlike urban environments, highways feature structured lanes, lack pedestrians, and are characterized by high speeds and distinct traffic patterns, such as frequent lane changes, merging, and platooning. These characteristics make highways a suitable environment for the early deployment of fully automated driving systems [7, 8]. Consequently, within the broader domain of automated driving technology, an increasing amount of research has focused on highway scenarios.

Since vehicles usually operate at high speeds on highways, often exceeding 60 mph (97 $\mathrm{km/h}$) [9], sudden lane changes or braking could result in severe accidents if the Autonomous Vehicle (AV) fails to anticipate potential dangers in advance. Reliable prediction of surrounding vehicles' future motions is therefore essential for safe AV control. However, motion prediction of surrounding vehicles is inherently coupled with the AV's own actions at each time step, as vehicles continuously influence one another's behavior over time. This mutual influence between vehicles is commonly referred to as **interaction**. As defined in [10], interaction arises when the behaviors of two or more road users depend on the possibility that they may attempt to occupy the same region of space in the near future. Despite its importance, many existing automated driving systems assume independent vehicle motions, resulting in overly simplistic and sometimes unrealis-

tic predictions [11]. Incorporating interaction enables motion forecasts that better reflect actual driving behavior [12], and integrating these predictions within the control loop allows AVs to plan more adaptive and less excessively cautious maneuvers [13]. Nevertheless, interaction-aware prediction remains challenging for automated driving, particularly when it must operate in real-time within a control framework. Because the intentions and driving styles of surrounding vehicles are unknown, their future motions are inherently uncertain. Moreover, modeling multi-agent interactions in a form suitable for control integration, while preserving computational efficiency, introduces substantial complexity.

Beyond the challenges of interaction modeling and motion prediction, ensuring safety in uncertain and dynamic highway environments also poses significant challenges for control design. Overly strict safety constraints can result in conservative behaviors, reducing efficiency and limiting practical responsiveness [14]. Balancing conservatism and robustness in control strategies across diverse traffic scenarios represents an important research direction [15]. This thesis addresses these challenges by investigating methods for modeling vehicle interactions, and develops advanced control architectures for safe autonomous highway driving. By advancing the understanding of vehicle interactions, the proposed approaches aim to enhance the safety, computational efficiency, and reliability of future autonomous highway systems.

1.2 Historical perspective

1.2.1 History

According to the official classification document, SAE-J3016, issued by the Society of Automobile Engineers (SAE) in 2014, driving automation is categorized into six levels, from Level 0 to Level 5 [16]. Fig. 1.1 illustrates the main characteristics of these levels, ranging from no automation to full automation.

One of the earliest public demonstrations of automated vehicle technology dates back to the early 1970s, when the British Road Research Laboratory released a video of a guided driverless car prototype. Although it relied on dedicated road infrastructure, the demonstration attracted considerable attention from both academia and industry [5]. Subsequent efforts further advanced the field. In 1977, researchers at the Tsukuba Mechanical Engineering Laboratory in Japan developed one of the earliest vision-guided AV prototypes, which used on-board cameras and analog image-processing circuitry to detect road markings on a dedicated test track [17].

From the 1980s to the 1990s, the development of self-driving technology experienced its first major acceleration, driven by advances in computing power and the emergence of new sensing technologies such as cameras, Radio Detection and Ranging (Radar), and experimental Light Detection and Ranging (LiDAR) systems [18]. In the mid-1980s, Defense Advanced Research Projects Agency (DARPA) initiated the Autonomous Land

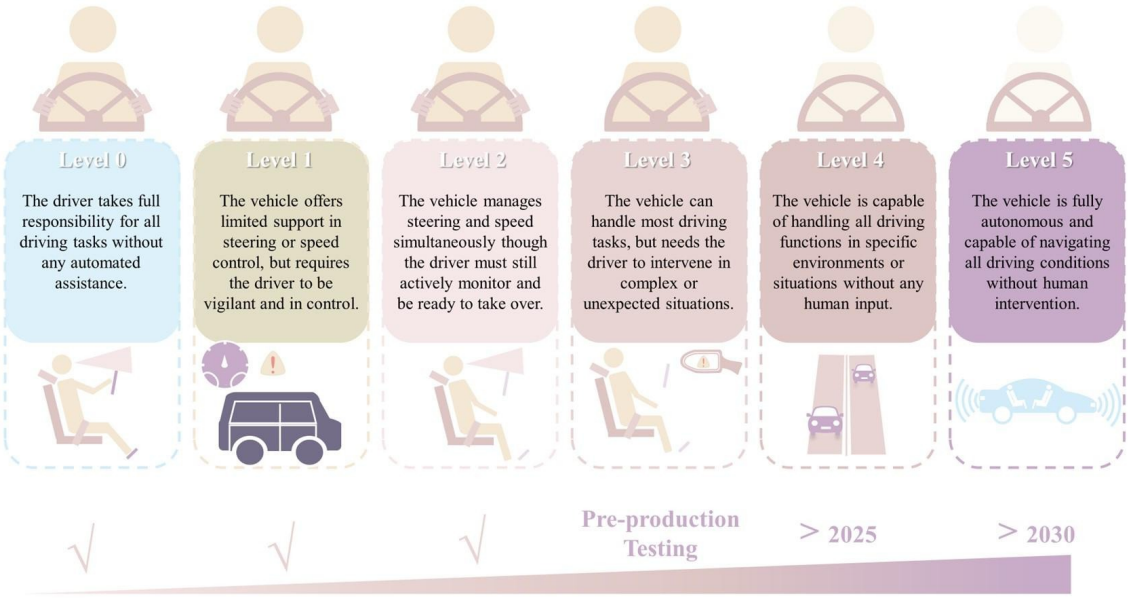


Fig. 1.1 Features of six levels of driving automation [1]. Level 3 systems are now available from BMW, Mercedes-Benz, and Honda under specific operating conditions and in limited regions [2].

Vehicle (ALV) program with substantial Army support [19]. Around the same time, Carnegie Mellon University launched the 'Navlab' project. The first 'Navlab' vehicle was constructed in 1986, using a combination of sensors and computing systems [20]. During this period, Ernst Dickmanns and his team at the Bundeswehr University Munich, Germany, developed a vision-guided autonomous Mercedes-Benz van. The system achieved autonomous driving at speeds up to $36\ \mathrm{km/h}$ on a closed test track and later reached $96\ \mathrm{km/h}$ on an empty highway segment in 1987 [5]. After approximately three years of work by DARPA, an ALV prototype was successfully demonstrated navigating controlled off-road terrain [21]. The 'Navlab' series continued to advance through the early 1990s. 'Navlab 5' completed a 3,100-mile cross-country trip in 1995, achieving 98.2% autonomous driving with minimal human intervention [22]. In 1997, an automated highway system was demonstrated in San Diego, illustrating coordinated vehicle platooning and automated control along dedicated highway test segments, marking another milestone in early large-scale automation research [23].

Starting in the early 2000s, DARPA launched the Grand Challenges for AVs. Teams from around the world participated, testing their autonomous systems under demanding real-world conditions [24]. In the first event in 2004, no team was able to complete the course [25]. However, substantial progress followed in 2005, when Stanford University's AV 'Stanley' successfully finished the course and won the competition [26]. The 2007 DARPA Urban Challenge further advanced the field by requiring vehicles to navigate complex urban-like environments, follow traffic rules, and interact with other AVs [27]. In 2009, Google launched its AV program, later spun off as Waymo, initiating major

efforts toward commercial and industrial development of AV technology [28].

In 2010, the University of Parma's VisLab conducted the VisLab Intercontinental Autonomous Challenge, completing a 15,900 km trip in 100 days from Parma, Italy to Shanghai, China [29]. In 2013, VisLab ran the PROUD project [30], which involved autonomous driving in mixed traffic on public roads through Parma, Italy [30]. Parallel European initiatives also advanced the field. The HAVEit, ABV, and CityMobil projects demonstrated autonomous highway and urban driving capabilities around 2011 [7], and platooning-based highway driving was achieved in the 2012 SARTRE Project [7]. In 2013, Daimler and the Karlsruhe Institute of Technology completed autonomous driving along Bertha Benz Memorial Route, showcasing both city and highway operation [31]. Alongside academic research, the autonomous driving industry grew rapidly, with numerous systems developed to assist human drivers [4, 32]. In 2014, Tesla introduced Autopilot, a consumer-oriented Advanced Driver Assistance System (ADAS) providing Level 1 automation [33]. Tesla subsequently expanded its capabilities, and by 2019 offered features marketed as Full Self-Driving, which correspond to Level 2 automation [32, 34]. Waymo also played a significant role in industrial progress, launching a public pilot ride-hailing program in the Phoenix area in 2017 with autonomous minivans [28], and expanding its fully autonomous robotaxi operations to additional U.S. cities beginning in 2022 [35, 36].

More than half a century of research and development has produced significant achievements in automated driving [5, 6, 37]. Despite these advancements, most commercially available automated driving systems remain at Level 2, requiring drivers to monitor the vehicle and take control when necessary [1]. More recently, Honda in Japan, Mercedes-Benz in Germany and select U.S. states, and BMW in Germany have introduced certified Level 3 systems, allowing drivers to disengage from monitoring the highway under specific conditions [2, 32]. Meanwhile, companies worldwide continue to advance higher levels of automation, with ongoing research, testing, and limited deployment of Level 4 and above autonomous systems [32].

1.2.2 Current state of technology and future challenges

ADAS is the most widely deployed form of automated driving technology, providing partial automation to support drivers [38, 39]. The system integrate sensors, control units, software algorithms, and vehicle actuators, such as steering and braking systems, to enhance safety and convenience. Current ADAS functionalities, corresponding primarily to SAE Levels 1 and 2 [1], include Adaptive Cruise Control (ACC), Lane Departure Warning (LDW), Automatic Emergency Braking (AEB), and Automated Parking (AP) [40]. In highway driving applications, ADAS focuses on enhancing safety by supporting longitudinal and lateral vehicle control, particularly in tasks related to maintaining stability and avoiding collisions. Commonly deployed ADAS features include ACC, AEB, LDW,

Lane Keeping Assist (LKA), Lane Change Assist (LCA), and Intelligent Speed Adaptation (ISA), which assist drivers in maintaining lane position, controlling speed, performing lane changes, and respecting speed regulations [7].

In the commercial mobility sector, automated driving technology is increasingly being deployed. Waymo operates autonomous ride-hailing services in several U.S. cities, allowing passengers to hail driverless robotaxis [35, 36], while autonomous shuttle services are offered in controlled environments such as campuses, airports, and urban areas, often under human supervision [41, 42]. Tesla's Full Self-Driving system provides Level 2 automation under specific conditions, gradually expanding driver-assistance capabilities [32, 34]. These examples demonstrate the growing adoption and versatility of automated driving technology and its potential to impact various industries and day-to-day activities.

Despite these advancements, significant challenges remain. Technical limitations, such as sensor constraints, computational delays, and perception uncertainty continue to affect system reliability [1]. Accurately predicting the motion behavior of surrounding vehicles is critical for safe AV operation, yet inherently difficult due to the interactive and dynamic nature of traffic. Interaction-aware motion prediction must account for multi-agent behaviors and the coupling between the AV's own actions and those of surrounding vehicles, introducing considerable uncertainty and complexity [13, 43].

Ethical and regulatory challenges further complicate the development. In situations where accidents are unavoidable, determining appropriate AV actions raises ethical dilemmas [4, 44, 45]. Legal frameworks must accommodate diverse traffic environments and evolving technology. Additionally, increased connectivity, such as Vehicle-to-Vehicle (V2V) and Vehicle-to-Everything (V2X) communication, exposes AVs to cybersecurity threats, which can compromise passenger safety and privacy [46, 47].

1.3 Literature review and fundamentals of automated driving

1.3.1 System architectures for automated driving

From the perspective of a layered architecture [1], an automated driving system is typically divided into three layers: the perception and Simultaneous Localization and Mapping (SLAM) layer, the planning layer, and the control layer, as illustrated in Fig. 1.2. The layered architecture functions as follows: in the first layer, the surrounding environment of the AV, such as objects, buildings, and traffic signs, is captured and processed as data by sensors [48, 49]. In addition to Global Positioning System (GPS) and Inertial Measurement Units (IMUs), camera, LiDAR, and Radar are commonly used to detect, track, and localize objects of interest [50, 51]. Then, the important messages, like the states of objects (e.g., position, velocity, attitude, etc.), are estimated and integrated into a representation of the environment [52, 53]. The first-order mapping subsystem, part of

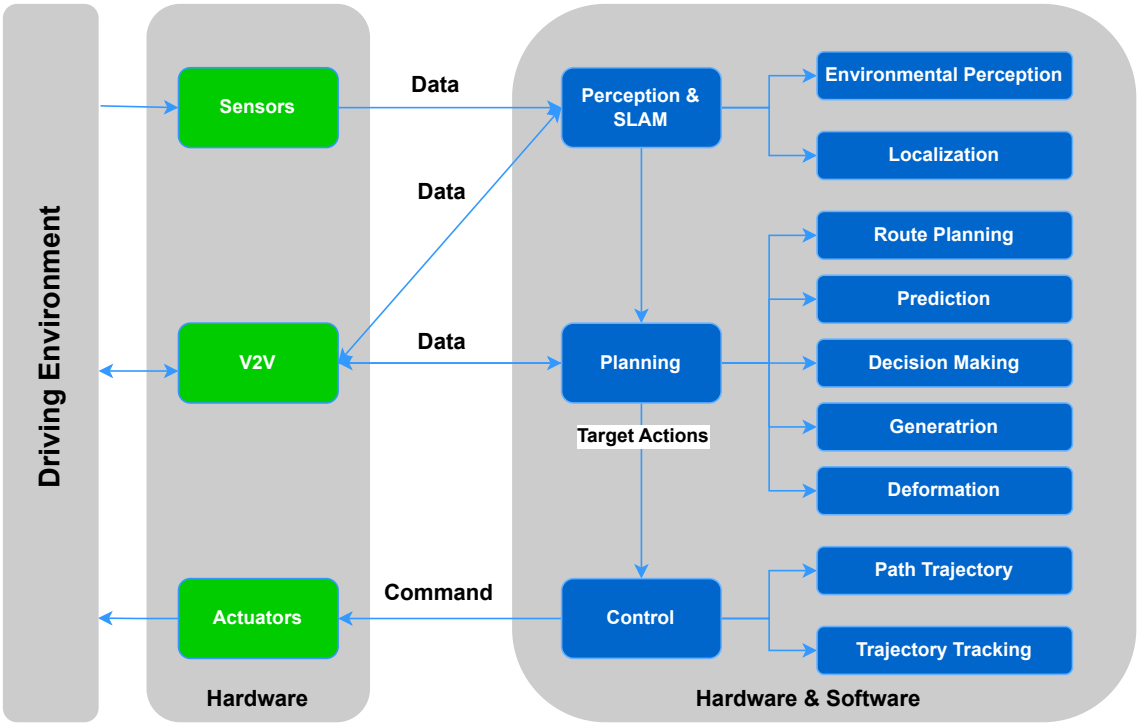


Fig. 1.2 Automated driving system structure [1]

the SLAM system, receives offline maps and the vehicle's state as input to produce online maps. This enhances the information available to the decision-making system [1]. In the subsequent layers, decision-making and planning for the AV are performed using the integrated algorithms based on the processed data [54–56]. Finally, after generating a reference path, the control unit guides the AV to follow it and achieve its goal through actuators [57–59].

Perception and SLAM Layer: Environmental perception in automated driving relies heavily on advancements in sensor technologies [48, 49]. Specifically, the camera excels at interpreting subtle nuances of textures and colors with remarkable precision, making it indispensable for critical automated navigation tasks such as lane detection and traffic sign recognition [1]. However, processing the rich information captured by cameras requires significant computational resources. The development of Machine Learning (ML) algorithms, particularly Deep Learning (DL), has become the foundation for meeting these computational demands. These advanced algorithms utilize Neural Network (NN) infrastructure as their computational backbones, enabling the extraction of rich information through complex neural networks, which enhances both accuracy and efficiency in perception tasks [1, 52, 60]. LiDAR, provides high-resolution and high-precision information, i.e., point clouds, even in dynamic environments. Its capabilities are crucial for tasks like navigation, obstacle detection, and collision avoidance in complex traffic scenarios. The raw data collected by LiDAR is processed through advanced computational techniques, including point cloud processing, SLAM algorithms, and DL-based approaches.

These technologies enable the creation of detailed 3D maps, providing a comprehensive representation of the surrounding environment [61–63]. Despite the acknowledged precision, LiDAR performance can degrade under adverse weather conditions such as heavy rain or fog, and its relatively high cost and physical size still limit widespread adoption [64]. In contrast, the Radar is robust under adverse weather conditions and excels at detecting objects over considerable distances [65]. Automotive Radar transmits frequency-modulated-continuous waveform at millimeter-wave frequencies, which supports high-resolution target range and velocity estimation at a much lower cost than LiDAR technology [66]. Although Radar offers lower spatial and angular resolution, it remains widely deployed in features such as ACC and collision-avoidance systems [1]. Additionally, ML algorithms and deep NNs have also been applied in automotive Radar for target recognition and classification [66].

Planning Layer: Followed by the perception layer, path planning/trajectory planning, and decision-making are executed in this layer using the processed data. According to the classification, most studies categorize this layer based on three levels of skills and tasks: strategical (planning), tactical (maneuvering), and operational (control) [67]. Five main functions in the motion strategy hierarchy are accordingly defined: (i) route planning, (ii) traffic prediction, (iii) decision-making, (iv) trajectory generation, and (v) trajectory deformation [7]. It works as follows: firstly, a general long-term path is generated through the road network from the initial position to the desired destination [68]. Then, the AV's surrounding objects' motion behaviors are estimated and predicted. The associated long-term risks are evaluated, and the path is re-planned for the AV [11]. Subsequently, under high-level predictive planning based on risk evaluation, criteria minimization, and constraint submission, the decision-making module chooses the best behavior motion from all candidates' generations [69]. Finally, low-level reactive planning deforms the generated motion from the high-level planning according to a deformation function [70]. The goal of this layer is summarized as to build an admissible path/trajectory from the starting point to the desired destination point smoothly, and constrained by safety, traffic rules, and other relevant factors that influence safe driving conditions. Traditional path and trajectory planning algorithms are generally classified as search-based [71, 72], probability-based [73, 74], AI logic-based [1, 75], parametric and semi-parametric curve based [76, 77], and numerical optimization-based [78, 79].

Control Layer: The role of the control layer is to follow the planned trajectory generated from the last layer as a reference in the presence of modeling error and other forms of uncertainties [59]. The Pure Pursuit controller and its variants are among the earliest approaches applied to vehicle control [80, 81]. The method follows a predefined path by steering toward a dynamically selected look-ahead point. Although the method faces challenges in dealing with sharp curves due to its use of a fixed look-ahead distance, it has been practically used in early AVs because of its simple implementation and great

performance [26, 27]. The rear wheel position-based feedback method is another solution, in which the control law regulates the steering angle based on the position of the rear-axle center (i.e., the midpoint of the rear axle) relative to a reference path, and the vehicle dynamics. The method provides good stability at low-speed but limited performance at high-speed, which makes it suitable for parking or low-speed navigation [59,82]. A related approach is front wheel position-based feedback control, which regulates both the front-wheel position error and the heading error. Because it better accounts for vehicle geometry and heading dynamics, this method generally provides improved stability at higher speeds compared with rear wheel position-based feedback [26, 59, 82]. However, on slippery roads or in emergency situations, the above control methods fail to generate sufficiently sophisticated control actions due to the simplicity of the underlying vehicle model [59]. Model Predictive Control (MPC) is an effective methodology for addressing this limitation [83]. Vehicle models with actuation limitations can be incorporated as hard constraints in the optimization problem formulation, commonly referred to as a Constrained Finite-time Optimal Control Problem (CFTOCP). Driving performance goals, such as ensuring safety and enhancing driver comfort, can be included as soft constraints [84]. In addition to handling multiple constraints, MPC enables the achievement of various control objectives by designing the objective function to minimize the difference between decision variables and reference values, such as trajectory tracking and energy efficiency. Due to its compact and straightforward structure, MPC has been successfully applied to vehicle control in various traffic scenarios [85]. AI-based approaches offer another solution, leveraging vast datasets of recorded human driving behaviors to train networks capable of producing human-like control actions for AVs [57, 86]. Though AI-based methods are widely applied in real-world AV applications, their further development is hindered by the lack of safety guarantees [87], the need for interpretable control outputs, and the requirement for reliable datasets [88].

Summary: A layer-structured automated driving system organizes driving tasks into a systematic hierarchy to support efficient execution. Although the boundaries between layers are not always strict, the architecture highlights the importance of coordinated interactions among them. For example, decision-making processes may span both the motion-planning and control layers rather than residing in a single layer. However, the use of multiple interconnected layers can also increase computational complexity and introduce delays in the system's overall response [1].

1.3.2 Terminology

Relevant terminology and scenario context used in this thesis are defined below.

Ego Vehicle (EV): The AV employing the proposed control method is referred to as the EV.

Target Vehicle (TV): A vehicle surrounding the EV that may interact with it is referred

to as the TV. The TV is assumed to be human-driven.

Leading Vehicle (LV): The TV drives in front of the EV is referred to as the LV.

Interaction-aware control: This term refers to control methods that account for the mutual influence between the EV and surrounding TVs (see Section 1.1 for a definition of interaction).

Interaction awareness: This term refers to the capability of a vehicle receiving and responding other surrounding vehicles' motion decisions.

The definition of the term "interaction-aware" is not consistent in the literature and may depend on the specific application context. The authors in [89] summarized studies on predicting vehicle lane change decision-making on highways under the term "intention-aware", where collected road users' information is used to infer human-driven vehicles' intentions at three sub-levels: strategic, tactical, and control levels. The methods for describing intentions are categorized into four types: generic models [90–93], such as Hidden Markov Model (HMM), Artificial Neural Network (ANN), and Bayesian Network (BN); discriminative models [94, 95], like Support Vector Machine (SVM); DL methods [96–98], including deep Convolutional Neural Network (CNN); and cognitive models [99, 100], which are built based on the Adaptive Control of Thought-Rational framework. "Behavior-aware" is introduced in [101] to describe DL-based prediction methods with the different input types [102, 103], such as history data of vehicles [104, 105], and raw sensor data [106, 107]. In [108], motion trajectories are predicted by incorporating vehicle interactions and additional contextual information to enhance DL models, ensuring interpretability and explainability in automated driving systems. These driving-related factors are collectively termed as "driving knowledge". Some proposed concepts combine these key terms. For instance, in [109], "behavioral-intention-aware" is defined to summarize prediction methods that simulate human reasoning to anticipate the behaviors of vehicles and pedestrians in urban settings. Additionally, "context-aware" and "situation-aware" are widely used in current literature to describe the vehicle awareness of the surrounding environment from the EV's perspective [110–112].

1.3.3 MPC for automated driving

Compared to traditional control methods and advanced AI-based control methods, MPC is more suitable for developing a simple and compact control structure that explicitly incorporates interactions between vehicles [83]. Its optimization-based formulation allows multiple inputs, objectives, and constraints to be handled in a unified manner, enabling vehicle interactions to be represented directly and efficiently within the constraint formulation, with decisions that remain transparent and interpretable. Under the MPC framework, predicting states over a horizon allows the controller to account for the future motion of both the EV and surrounding TVs, further possibly ensure the EV's safety under uncertain environments. Based on whether TV motion prediction and EV control are addressed as

separate modules or are integrated into a unified framework, the existing literature can be broadly categorized into two control approaches:

- *MPC with interaction-aware predictions*: The MPC controller uses precomputed predictions of TVs' behaviors to generate control actions.
- *Fully interaction-aware MPC*: Vehicle interactions are incorporated directly within the MPC optimization to compute control actions for the EV, considering how TVs may respond.

1.3.3.1 MPC with interaction-aware predictions

For traffic prediction, there are different approaches to model interaction awareness of vehicles [113, 114]. One representative method is to use interaction-aware models to simultaneously predict the motion behaviors of TVs and the EV [115]. Car-following models are widely used as the longitudinally interaction-aware models, like the Intelligent Driver Model (IDM), the Optimal Velocity (OV) model, etc [116–118]. In the lateral direction, due to the uncertainties of the lane change decision, lane change time, and the way to do lane change, different interaction-aware lane change models are developed, like the Gipps model, the Minimizing Overall Braking Induced by Lane Change (MOBIL) model, etc [119, 120]. The above car-following and lane-change models are usually built based on unknown parameters. These parameters are usually estimated or learned to represent the driver's personality [115, 121]. Moreover, vehicles' interactions can be effectively and compactly encoded in the vehicle's motion modeling, and also in Markov chain transitions under the Interacting Multiple Model Kalman Filter (IMM-KF) framework. The IMM-KF algorithm has been successfully applied in predicting highway scenarios [122, 123].

In addition to the model-based traffic prediction approaches, model-free traffic prediction approaches have also been intensively studied [124, 125]. Game theory is a popular choice, where traffic participants are assumed to make decisions by maximizing their utility [126, 127]. Vehicle interactions are represented by enumerating the possible maneuvers at each time step [128, 129]. Game trees are then formulated considering the probability of maneuvers from different aspects, like drivers' intentions, observations, or associated risks. However, solving for an equilibrium is still a challenge.

NN based approaches have been extensively investigated due to the development of perception systems, which makes obtaining data from various driving agents and the environment easier [10]. Spatial and temporal interaction of vehicles are modeled based on the dataset, using methods such as Recurrent Neural Networks (RNNs) [130], and Graph Neural Networks (GNNs) [131]. A DL-based method is applied in [132], where a graph NN encodes the interaction feature by modeling the traffic scene as a graph, and assuming traffic agents and their interconnection as nodes and edges, respectively.

Traffic predictions are then incorporated into the controller design to prevent the EV from colliding with TVs. Within the MPC framework, traffic predictions, along with the EV's dynamics and actuation limitations, are systematically formulated as constraints. Unknown driving intentions and styles of the drivers of TVs introduce uncertainties to traffic predictions. The corresponding uncertainty in real future motion behaviors of TVs has to be reflected in the formulation of the constraints of the MPC. **Stochastic MPC (SMPC)** handles uncertainties by modeling them as statistical probability distributions, and computes control actions that minimize the expected cost [133]. The resulting performance, however, strongly depends on the accuracy of the underlying probabilistic model [134]. In practice, constructing such stochastic models can be difficult or even infeasible for certain control applications, limiting the applicability of SMPC. [135]. The **Scenario-based MPC (SCMPC)** approach addresses uncertainty based on a limited number of scenarios [136, 137], and it is easily compatible with the traffic prediction component. It has been widely applied in various autonomous highway driving scenarios [115, 129].

1.3.3.2 Fully interaction-aware MPC

Some studies propose methods for directly determining the states of the EV and TVs considering their possible interaction models, by solving an optimization problem formulated within MPC framework [43, 138]. However, the amount of existing literature on this approach is significantly smaller compared to studies focusing on separate traffic prediction and MPC control methods. Since the states of both the EV and TVs are treated as decision variables, it is challenging to appropriately update vehicles' states, and propagate the uncertainties of their interactions within the prediction horizon. The formulation of the problem plays a crucial role in determining the complexity of solving the corresponding optimization problem. One possible solution is **SMPC**, where the EV's motion is modeled as a stochastic process based on the interaction with TVs, and mode transitions are modeled as a Markov chain. The transition model is either learned or estimated from previous EV and TVs measurements [139]. Another approach is **Branch MPC**, which employs a scenario tree and a trajectory tree to represent the possible motion behaviors of the EV and TVs. The branch probability of both trees influences each other based on their interactions [140]. However, the method proposed in [140] considers only one EV and one TV, and an increment in the number of TVs incredibly complicates the interaction formulation. Additionally, one method representing the EV and TV states simultaneously in an Affine Time-Varying (ATV) system within MPC is introduced in [141], where the system model and interactions are processed as nodes and edges in the Long Short-Term Memory (LSTM) networks. Despite their advantages, these methods suffer from high computational complexity due to the complicated interaction formulation within MPC.

1.3.3.3 Ensuring safety in MPC for automated driving

Safety is one of the most critical aspects of controlling the EV. This feature becomes even more challenging in emergency scenarios, such as an unexpected deceleration of the LV or a sudden cut-in by a TV. Considering these emergency scenarios is essential to ensure safety-guaranteed control actions. These circumstances are identified as a Safety-Critical-Event (SCE) [142], where the EV immediately brakes against the crash, potentially leading to a standstill due to rapid deceleration [143]. ACC is an example of a driver assistance system for resolving SCEs. In this system, the EV reacts based on information about the LV. However, a large time headway in this algorithm may lead to over-conservative actions [144]. [145] proposes a safety controller based on ACC, where the EV uses a predefined deceleration profile. Another representative solution to SCEs is the rigorous formalizing mathematical model of Responsibility-Sensitive-Safety (RSS) [146]. In this model, a safety distance is defined by assuming a "worst-case" scenario, and the EV responds to an SCE by decelerating at a predefined rate without full braking. This approach might be sensitive to the parameter design, a subtle change of the parameter set might lead to a different decision strategy [147].

Moreover, incorporating the safety guarantee into the MPC framework is crucial yet challenging due to the existing uncertainties. **Contingency MPC (CMPC)** offers a promising direction, as it integrates contingency resolutions alongside nominal ones under its framework to achieve safety. Meanwhile, uncertainties arising from predicting TVs motion behaviors can be presented by a finite number of nominal scenarios. The key to designing a CMPC controller lies in defining the contingent scenario, obtaining nominal scenarios, and ensuring safety, which also distinguishes different studies. A CMPC method proposed by [148], introduces two prediction horizons of different lengths during nominal and contingent scenario generations. Two sets of control actions are computed to meet the nominal and contingent objectives, respectively. Safety is guaranteed by aligning both types of control actions to be identical in the first step. Meanwhile, to regulate the trade-off between conservativeness and performance incentives, two tunable weight variables are employed, with their sum constrained to 1. However, selecting appropriate weights is non-trivial and requires expertise or prior knowledge. SMPC typically ensures safety probabilistically by constraining the risk probability within a predefined threshold using chance constraints. One approach fulfills the stochastic safety guarantee by proving the generalization bound for the predictive model through post-bloating, SVM, and conformal analysis methods, where a set of all possible TV trajectories within a given scenario is obtained by the trained predictive model [149]. However, a key limitation of SMPC is that chance constraints inherently allow for a small probability of safety violations. To address this problem, [14] proposes to exploit SMPC and fail-safe trajectory planning sequentially. Initially, SMPC is used to generate a safe trajectory based on a pre-

defined risk threshold. To further ensure safety, a backup trajectory is incorporated based on reachable sets, which is activated when the primary SMPC trajectory risks violating safety constraints. Despite its advantages, SMPC faces computational challenges due to the probabilistic formulation in the cost function and the complexity of solving schance constraints efficiently.

1.4 Contributions

The multi-layer structure of the automated driving architecture can introduce delays between processing stages, which in turn add to the overall computational burden [1]. To address this computational challenge, this thesis integrates planning and control into a unified framework. This integrated component, referred to as the control implementation for EVs, constitutes the main focus of the thesis and is highlighted in the red box in Fig. 1.3.

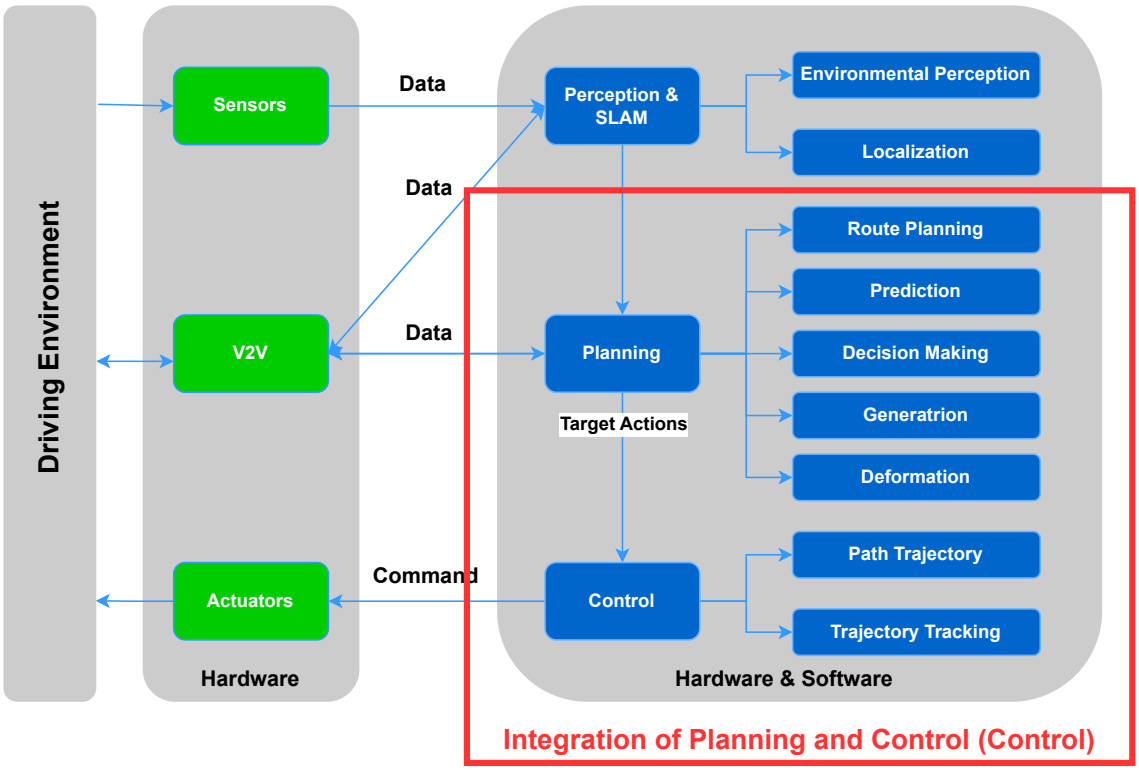


Fig. 1.3 Research scope of the thesis

At the same time, this thesis focuses on the real-time prediction of surrounding TVs' motion maneuvers based on their interactions, with the goal of ensuring safe EV operation. Building on the identified challenges and opportunities in modeling vehicle interactions and achieving safe control for autonomous highway driving, the following research questions are formulated to be addressed:

- Questions on control methodology and algorithm derivation:

1. How can a control framework be developed to enable the implementation of both planning and control for the EV?

2. How can interactions between the EV and TVs be modeled and predicted under the proposed control framework?

3. How can decision-making be realized for the EV under the proposed control framework?

- Question on the feasibility of solving optimization problems:

  4. How can the recursive feasibility of MPC be guaranteed?

- Questions on controller performance for autonomous highway driving:

  5. What are the key performance characteristics of the proposed control approaches?

The main contributions of this thesis are summarized as follows. A computationally efficient IMM-KF-based algorithm is introduced to produce possible interaction-aware scenarios, simplifying interaction modeling with the MPC framework [122] (Chapter 3). Moreover, this thesis proposes an improved IMM-KF-based algorithm to enhance the interaction-aware traffic prediction by adaptively updating the transition probability matrix according to vehicle's motion mode switch, and further improve accuracy of model likelihoods (Chapter 3). A SCMPC controller is proposed to compute collision-free actions for the EV by considering the generated nominal scenarios (Chapter 4). Unlike methods that include the decision-making of lane-change for the EV during the interaction process [139–141], this approach simplifies the decision-making by considering only two control modes, 'lane-keeping' and 'lane-change', and selecting the mode with the lower associated cost. To guarantee the safety of the control actions, a "worst-case" scenario is introduced into the SCMPC formulation, and the recursive feasibility is rigorously proved.

A CMPC controller is also proposed to reduce the conservativeness arising from considering the "worst-case" scenario (Chapter 5). For the nonlinear vehicle model used in CMPC, a differential flatness based method is employed to obtain a feedback linearized representation. This ensures that the vehicle model is locally controllable, in contrast to the negative controllability result observed for its linearization at an arbitrary equilibrium. This method also simplifies the optimization problem while maintaining accuracy in describing the vehicle's motion. Compared to the terminal set introduced in the SCMPC controller, the considered one within the CMPC scheme is enlarged. Additionally, this approach reduces the number of decision variables by using time-varying prediction horizons.

In addition, this thesis proposes a fully interaction-aware control approach by fusing the motion states of the EV and the longitudinal motion states of TVs within the MPC

framework (Chapter 6). Instead of predefining the interaction mechanism of TVs, they are optimally determined by minimizing the total effort of all vehicles across various traffic scenarios. For the lateral motion uncertainties of vehicles, a rule-based MOBIL model, is proposed to predict the possible lanes vehicles will occupy, based on the overall benefit to traffic flow. Unlike approaches that consider all possible TV motion behaviors at each prediction step, this method incorporates the causality of lane changes at each decision-making step, leading to fewer and more realistic predicted traffic scenarios. In the proposed control structure, the MPC formulation eliminates the need for information about all states of TVs, requiring only the possible lateral positions. Also, by introducing different control modes for the EV, decision-making, planning, and control are integrated into a single module for autonomous driving. The interaction of vehicles over the prediction horizon is considered from the viewpoint of the entire traffic environment. The proposed method is computationally efficient because of the linearity of the optimization problem and the compact control structure.

Evaluating the designed controllers in an appropriate environment is essential for demonstrating the effectiveness of algorithms. On the one hand, during controller design, a simplified vehicle model is usually used because fully accounting for the vehicle's real complex dynamics within the MPC framework is nearly impossible. The highly nonlinear vehicle dynamics also pose challenges for solving the associated optimization problem. Therefore, applying the calculated control actions in an environment that reflects real vehicle dynamics is both important and necessary. On the other hand, we focus on modeling the interactions between vehicles to obtain safety-guaranteed control actions. It is thus crucial to ensure that the simulated TVs in the environment interact with the EV similarly to how real human drivers would. Considering these requirements, we choose IPG CarMaker to evaluate the developed control approaches in this thesis. In particular, the EV's dynamics are represented by a realistic, real-time capable model, which can be expanded to 280 degrees of freedom. Traffic scenarios can be configured arbitrarily or generated based on real-world data. TVs' motion behaviors are modeled in automated driving mode using human-like models, such as the Human Driver Model (HDM). Additionally, IPG CarMaker's compatibility with other interfaces, particularly with Simulink, facilitates seamless implementation.

1.5 Thesis outline

Fig. 1.4 illustrates the overall structure of the thesis. Chapter 2 reviews the theoretical foundations and relevant knowledge for MPC (Section 2.1) and MPC-based control methods (Section 2.2), with a particular focus on SCMPC (Section 2.3.2) and CMPC (Section 2.3.3).

Considering the distinctive advantages of the control methods, several controllers for

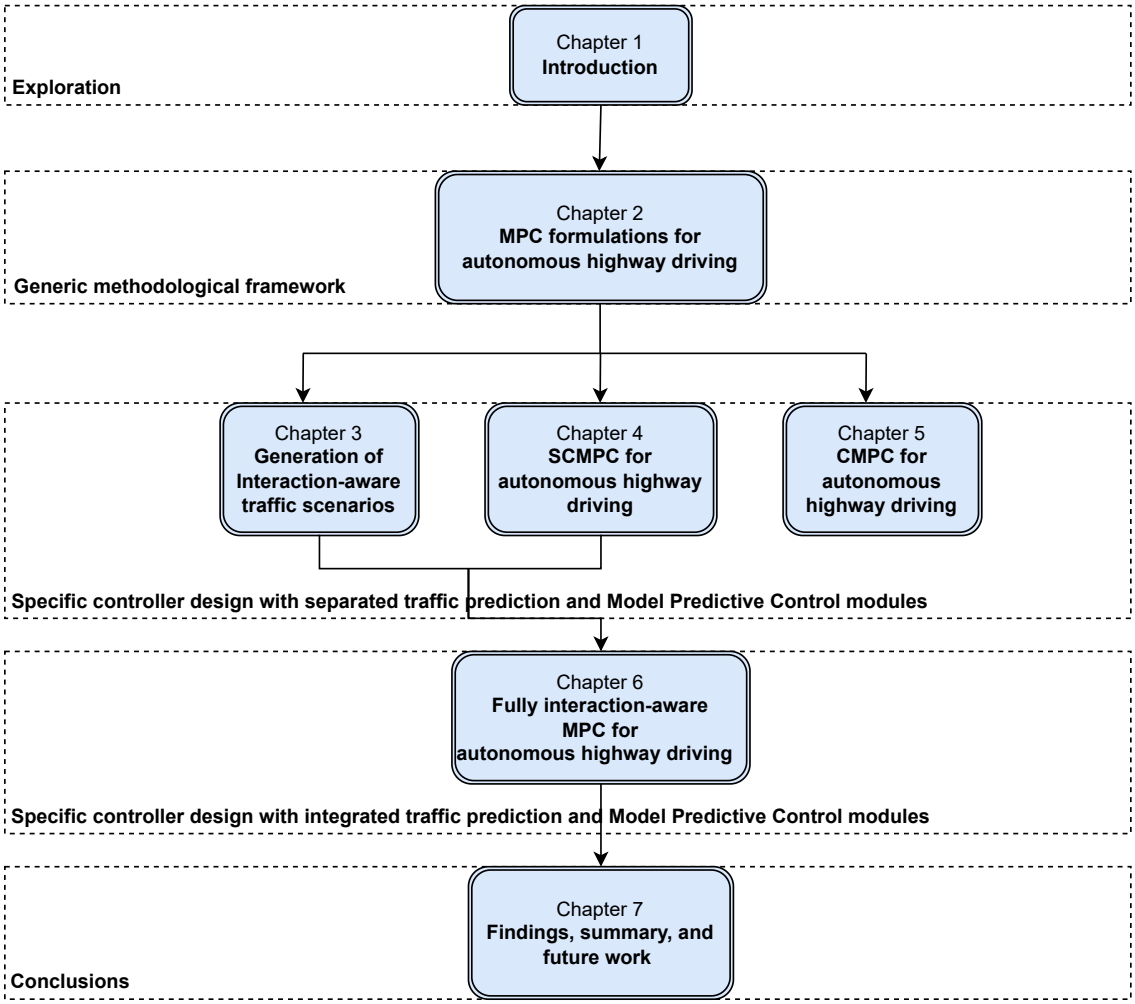


Fig. 1.4 Overall structure of the thesis

autonomous highway vehicles are designed in the following chapters with different design considerations and objectives. In particular, Chapters 3, 4, and 5 develop MPC-based methods with interaction-aware predictions, sequentially examining interaction-aware traffic predictions, integrated planning, decision-making and control, and recursive feasibility of the associated optimization problems to ensure safety. The logical flow between Chapters 3 and 4 is illustrated in Fig. 1.5. The corresponding results are summarized in: X. Zhang, S. Zeinali and G. Schildbach, "Interaction-aware traffic prediction and Scenario-based Model Predictive Control for autonomous vehicles on highways," in *2024 European Control Conference (ECC)*, IEEE, 2024, pp. 3351–3357 [115], and in the extended journal version: X. Zhang, S. Zeinali and G. Schildbach, "Interaction-aware traffic prediction and Scenario-based Model Predictive Control for autonomous vehicles on highways," *IEEE Transactions on Control Systems Technology*, vol. 33, no. 4, pp. 1235–1245, 2025 [150]. Chapters 3 and 5 also share a similar setup based on the separate control structure, which corresponds to the following manuscript currently under revision: X. Zhang, H. Wen, H. Abbas, and G. Schildbach, "Efficient Contingency Model Predictive Control for safe and less conservative autonomous driving," manuscript under revision for resubmission.

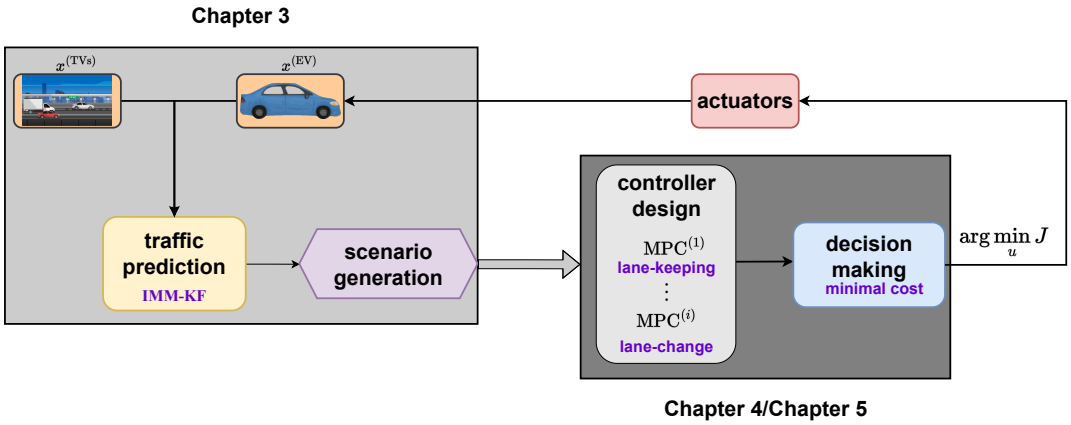


Fig. 1.5 Structure of the MPC with interaction-aware predictions

Chapter 6 proposes a fully interaction-aware MPC, integrating interaction-aware traffic prediction and EV control within the MPC framework. A corresponding overview is presented in Fig. 1.6, and the results are reported in: X. Zhang, S. Zeinali, H. Wen, and G. Schildbach, "MOBIL-based traffic prediction and interaction-aware Model Predictive Control for autonomous highway driving," *Control Engineering Practice*, vol. 164, pp. 106434, 2025. Finally, Chapter 7 concludes the thesis by summarizing the main findings and outlining their implications.

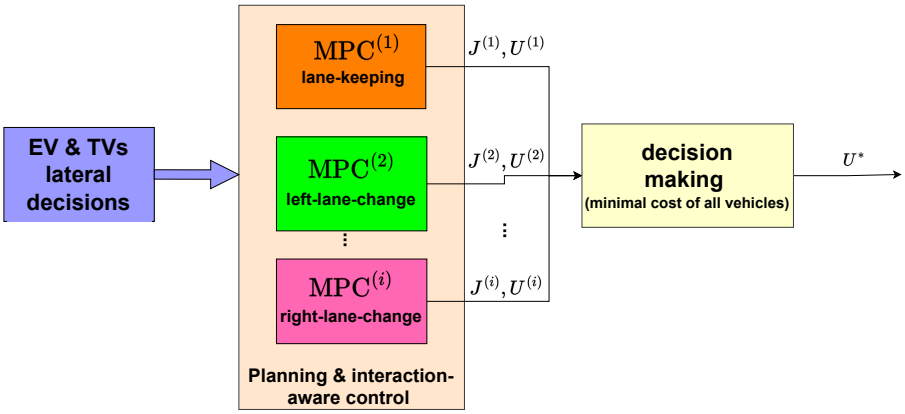


Fig. 1.6 Structure of the fully interaction-aware MPC (Chapter 6)

CHAPTER 2

MPC Formulations for autonomous highway driving

MPC-based methods are extensively employed in automated driving, as they provide a systematic framework to account for vehicle dynamics, interactions with surrounding traffic, and operational constraints [7, 11, 84]. This chapter introduces the relevant theoretical background and controller formulations.

2.1 Optimization and constrained optimal control

2.1.1 Numerical optimization

Convexity: A set $\mathbb{X} \subseteq \mathbb{R}^{n_x}$ is convex if

$$\lambda x_1 + (1-\lambda)x_2 \in \mathbb{X} \quad \text{for all } \; x_1, x_2 \in \mathbb{X}, \; \text{ and } \; \lambda \in [0,1]. \tag{2.1}$$

A function $f : \mathbb{X} \to \mathbb{R}$ is convex if $\mathbb{X}$ is convex and it satisfies the inequality

$$f(\lambda x_1 + (1-\lambda)x_2) \leq \; \lambda f(x_1) + (1-\lambda) f(x_2) \; \text{ for all } \; x_1, x_2 \in \mathbb{X}, \; \text{ and } \; \lambda \in [0,1]. \tag{2.2}$$

A function $f : \mathbb{X} \to \mathbb{R}$ is concave if $\mathbb{X}$ is convex, and $-f$ is convex.

The *convex combination* of a finite set of points $x_1, \ldots, x_k \in \mathbb{R}^{n_x}$ is defined as any point of the form $\lambda_1 x_1 + \cdots + \lambda_k x_k$, where $\lambda_i \geq 0$ for all $i = 1, \ldots, k$, and $\sum_{i=1}^{k} \lambda_i = 1$. The *convex hull* of a set $\mathcal{K} \subseteq \mathbb{R}^{n_x}$ is the set of all convex combinations of points in $\mathcal{K}$, and it is denoted by $\text{conv}(\mathcal{K})$ [3]:

$$\text{conv}(\mathcal{K}) = \Big\{ \sum_{i=1}^{k} \lambda_i x_i \;\Big|\; x_i \in \mathcal{K}, \; \lambda_i \geq 0, \; \sum_{i=1}^{k} \lambda_i = 1 \Big\}. \tag{2.3}$$

Polyhedron and polytope: A polyhedron $\mathcal{P}$ in $\mathbb{R}^{n_x}$ is defined as the intersection of a finite set of closed halfspaces in $\mathbb{R}^{n_x}$:

$$\mathcal{P} = \{x \in \mathbb{R}^{n_x} : Ax \leq b\}, \tag{2.4}$$

where $Ax \leq b$ represents a finite set of linear inequalities. A is the constraint matrix with

rows $a_1^\top, a_2^\top, \ldots, a_{n_m}^\top$, and $b \in \mathbb{R}^{n_m}$ is a vector of corresponding dimension. A polytope is a bounded polyhedron.

An optimization problem is generally formulated as:

$$\min_x \quad J(x) \tag{2.5a}$$

$$\text{s.t. } x \in \mathbb{X} \subseteq \mathrm{dom}(J), \tag{2.5b}$$

where x denotes the decision variables, $\mathrm{dom}(J) \subseteq \mathbb{R}^{n_x}$ is the domain of the objective/cost function J, and $\mathbb{X} \subseteq \mathrm{dom}(J)$ is the set of feasible or admissible decisions. The objective function J assigns a real-valued cost $J(x) \in \mathbb{R}$ to each decision x.

Solving the problem (2.5) means to find a decision $x^\star \in \mathbb{X}$ that minimizes $J(x)$. The minimal cost is denoted by $J^\star$, referred to as the optimal value, and the corresponding decision $x^\star$ achieving $J^\star$ is called an optimal solution. If the feasible set $\mathbb{X}$ is empty, the problem is infeasible, and by convention, we set $J^\star = +\infty$. If the feasible set coincides with the domain of the objective function, i.e., $\mathbb{X} = \mathrm{dom}(J)$, the problem is unconstrained.

A standard formulation of an optimization problem is given by:

$$\min_x \quad J(x) \tag{2.6a}$$

$$\text{s.t. } g_i(x) \leq 0, \quad i = 1, \ldots, n_m, \tag{2.6b}$$

$$h_j(x) = 0, \quad j = 1, \ldots, n_p. \tag{2.6c}$$

Here, $g_1, g_2, \ldots, g_{n_m}$ and $h_1, h_2, \ldots, h_{n_p}$ represent the inequality and equality constraint functions, respectively, where $g_i : \mathbb{R}^{n_x} \to \mathbb{R}$, $h_j : \mathbb{R}^{n_x} \to \mathbb{R}$. The feasible set $\mathbb{X}$ is defined by the intersection of $\mathrm{dom}(J)$ with the region satisfying all constraints. The optimization problem (2.6) is convex if J is convex and the feasible set $\mathbb{X}$ is convex. A key property of convex optimization problems is that any locally optimal solution is also globally optimal. Common classes of optimization problems include Linear Program (LP)s, Quadratic Program (QP)s, and Mixed Integer Program (MIP)s.

When the objective function and constraint functions of the optimization problem (2.6) are affine, the problem is called an LP. A commonly used standard formulation of an LP is given by:

$$\min_x \quad c^\top x \tag{2.7a}$$

$$\text{s.t. } Ax \leq b. \tag{2.7b}$$

Here, $x \in \mathbb{R}^{n_x}$ is the decision variable, $c \in \mathbb{R}^{n_x}$ is the linear coefficient vector. $A \in \mathbb{R}^{n_m \times n_x}$ is the constraint coefficient matrix, $b \in \mathbb{R}^{n_m}$ is the constraint bound vector. Since both the objective function and constraints are affine, LPs are convex optimization prob-

lems.

The optimization problem (2.6) is called a QP when the objective function is a convex quadratic function, and the constraint functions are affine. Its general form is given by:

$$\min_{x} \quad \frac{1}{2}x^\top Hx + c^\top x + r \tag{2.8a}$$

$$\text{s.t.} \quad Ax \leq b, \tag{2.8b}$$

where $x \in \mathbb{R}^{n_x}$ is the decision variable, $H \in \mathbb{R}^{n_x \times n_x}$ is a symmetric positive semidefinite matrix ($H = H^\top \succeq 0$), $c \in \mathbb{R}^{n_x}$ is a linear coefficient vector. $A \in \mathbb{R}^{n_m \times n_x}$ and $b \in \mathbb{R}^{n_m}$ define the linear inequality constraints. The term r is a constant, and does not affect the optimal solution, so it can be omitted if only the optimal x is of interest. Since H is positive semidefinite, QPs are convex optimization problems.

The optimization problem (2.6) is called an MIP when the feasible set $\mathbb{X}$ is the Cartesian product of a binary set and a real Euclidean space, i.e., $\mathbb{X} \subseteq \{(x_c, x_b) : x_c \in \mathbb{R}^{n_c}, x_b \in \{0,1\}^{n_b}\}$. A standard formulation of an MIP is given by:

$$\min_{x_c, x_b} \quad J(x_c, x_b) \tag{2.9a}$$

$$\text{s.t.} \quad g_i(x_c, x_b) \leq 0, \quad i = 1, \ldots, n_m, \tag{2.9b}$$

$$h_j(x_c, x_b) = 0, \quad j = 1, \ldots, n_p, \tag{2.9c}$$

$$x_b \in \{0,1\}^{n_b}, \tag{2.9d}$$

where the objective function J, inequality constraint functions $g_1, g_2, \ldots, g_{n_m}$, and equality constraint functions $h_1, h_2, \ldots, h_{n_p}$ are real-valued functions defined over $\mathbb{X}$.

When the objective function of the optimization problem (2.9) is quadratic and convex, and the constraints are affine, the problem is called a Mixed Integer Quadratic Program (MIQP). A general formulation is given by:

$$\min_{x} \quad \frac{1}{2}x^\top Hx + c^\top x + r \tag{2.10a}$$

$$\text{s.t.} \quad A_c x_c + A_b x_b \leq b, \tag{2.10b}$$

$$x_b \in \{0,1\}^{n_b}, \tag{2.10c}$$

where $A_c \in \mathbb{R}^{n_m \times n_c}$, $A_b \in \mathbb{R}^{n_m \times n_b}$, $b \in \mathbb{R}^{n_m}$, and $n_x = n_c + n_b$. MIQPs are generally non-convex due to the presence of binary variables. When $H = 0$, the problem reduces to a Mixed Integer Linear Program (MILP), which remains computationally challenging because of the integer constraints.

2.1.2 Constrained optimal control

Consider the discrete linear time-invariant (DLTI) system

$$x(k+1) = f(x(k), u(k)) = Ax(k) + Bu(k), \tag{2.11}$$

where $x(k) \in \mathbb{R}^{n_x}$, $u(k) \in \mathbb{R}^{n_u}$ denote the state and input vectors, respectively. A and B are the system and input matrices of appropriate dimensions. The state and input must satisfy the admissible sets $\mathbb{X}$ and $\mathbb{U}$, respectively:

$$x(k) \in \mathbb{X}, \quad u(k) \in \mathbb{U}, \quad \forall k \geq 0. \tag{2.12}$$

The sets $\mathbb{X} \subseteq \mathbb{R}^{n_x}$, and $\mathbb{U} \subseteq \mathbb{R}^{n_u}$ are assumed to be polyhedra.

The objective function over the time horizon from time 0 to time N is defined as:

$$J(x(0), U) = J_f(x_N) + \sum_{k=0}^{N-1} J_q(x_k, u_k), \tag{2.13}$$

where the variable x_k denotes the predicted state at time k, which is obtained by starting from the measured state $x_0 = x(0)$, and applying the input sequences $u_0, \ldots, u_{N-1}$ to the system (2.11). The input sequence is collected in $U = \begin{bmatrix} u_0^\top & \ldots & u_{N-1}^\top \end{bmatrix}^\top$. The terms $J_f(x_N)$ and $J_q(x_k, u_k)$ are referred to as the terminal cost and stage cost, respectively. Both cost functions are assumed to be positive definite.

A CFTOCP is given by:

$$\begin{aligned}
J^\star(x_0) = \min_{U} \;\; & J_f(x_N) + \sum_{k=0}^{N-1} J_q(x_k, u_k) && \text{(2.14a)}\\
\text{s.t.} \;\; & x_{k+1} = Ax_k + Bu_k, \;\; k = 0, \ldots, N-1, && \text{(2.14b)}\\
& x_k \in \mathbb{X}, \;\; k = 1, \ldots, N, && \text{(2.14c)}\\
& u_k \in \mathbb{U}, \;\; k = 0, \ldots, N-1, && \text{(2.14d)}\\
& x_N \in \mathbb{X}_f, && \text{(2.14e)}\\
& x_0 = x(0). && \text{(2.14f)}
\end{aligned}$$

Here, $\mathbb{X}_f \subseteq \mathbb{R}^{n_x}$ denotes the terminal set, which the system state must reach at the end of the prediction horizon. We distinguish between the current state $x(k)$ of system (2.11) at time k, and the optimization variable x_k used in the optimization problem (2.14). x_k represents the predicted state at time k, obtained by propagating the initial state x_0 under input sequences $u_0, \ldots, u_{k-1}$ according to (2.14b). Similarly, $u(k)$ denotes the actual input applied to the system at time k, while u_k is the k-th decision variable in the optimization problem. When designing the objective function (2.14a), different norm choices

can be applied to represent various control objectives. In general, the state and terminal costs can be expressed as:

$$J_f(x_N) = \|W_P x_N\|_p, \quad J_q(x_k, u_k) = \|W_Q x_k\|_p + \|W_R u_k\|_p, \tag{2.15}$$

where $p \in \{1, 2, \infty\}$. The weighting matrices W_P, W_Q, W_R are assumed to have full column rank and serve as tuning parameters. When $p = 1$ or $p = \infty$, the objective function represents a linear or max-norm formulation, resulting in an LP problem. In contrast, when $p = 2$, the objective is based on the Euclidean norm. Using its squared form yields a smooth and convex quadratic objective function, leading to a QP formulation under linear constraints [3].

2.1.3 Reachability and invariance

Consider a DLTI system (2.11) with state and input constraints (2.12), the *successor set* from the set $\mathcal{S}$ is defined as [3]

$$\text{Suc}(\mathcal{S}) = \Big\{ x \in \mathbb{R}^{n_x} : \exists x_0 \in \mathcal{S},\ \exists u_0 \in \mathbb{U}\ s.t.\ x = f(x_0, u_0) \Big\}. \tag{2.16}$$

The *precursor set* to the set $\mathcal{S}$ is defined as [3]

$$\text{Pre}(\mathcal{S}) = \Big\{ x \in \mathbb{R}^{n_x} :\ \exists u \in \mathbb{U}\ s.t.\ f(x, u) \in \mathcal{S} \Big\}. \tag{2.17}$$

For a given initial set $\mathbb{X}_0 \subseteq \mathbb{X}$, the *N-step reachable set* $\mathbb{R}_N(\mathbb{X}_0)$ of the system (2.11) subject to the constraints (2.12) is defined as [3]

$$\mathcal{R}_{i+1}(\mathbb{X}_0) = \text{Suc}(\mathcal{R}_i(\mathbb{X}_0)) \cap \mathbb{X},\ \mathcal{R}_0(\mathbb{X}_0) = \mathbb{X}_0,\ i = 0, \ldots, N-1. \tag{2.18}$$

For every $x_0 \in \mathbb{X}_0$, there exists an admissible input sequence such that the state at time N belongs to $\mathbb{R}_N(\mathbb{X}_0)$.

For a given target set $\mathcal{S} \subseteq \mathbb{X}$, the *N-step controllable set* $\mathcal{K}_N(\mathcal{S})$ of the system (2.11) subject to the constraints (2.12) is defined recursively as [3]

$$\mathcal{K}_j(\mathcal{S}) = \text{Pre}(\mathcal{K}_{j-1}(\mathcal{S})) \cap \mathbb{X},\ \mathcal{K}_0(\mathcal{S}) = \mathcal{S},\ j \in \{1, \ldots, N\}. \tag{2.19}$$

All states $x_0 \in \mathcal{K}_N(\mathcal{S})$ can be driven to the target set $\mathcal{S}$ in N steps by a suitable choice of admissible inputs, while satisfying state constraints.

A set $\mathcal{C} \subset \mathbb{R}^n$ is said to be a *control invariant set* for the system (2.11) subject to the constraints (2.12), if $\mathcal{C} \subseteq \mathbb{X}$ and

$$x_k \in \mathcal{C} \ \Rightarrow\ \exists u_k \in \mathbb{U} : f(x_k, u_k) \in \mathcal{C}. \tag{2.20}$$

The set $\mathcal{C}_\infty \subset \mathbb{R}^n$ is called a *maximal control invariant set* if it is not a strict subset of any other control invariant set.

Consider a DLTI system with additive disturbance or uncertainty:

$$x_{k+1} = f(x_k, u_k, w_k) = Ax_k + Bu_k + w_k, \quad w_k \in \mathbb{W}, \tag{2.21}$$

subject to state and input constraints (2.12).

The *robust successor set* from the set $\mathcal{S}$ is defined as [3]

$$\text{Suc}(\mathcal{S}, \mathcal{W}) = \Big\{ x \in \mathbb{R}^{n_x} : \exists x_0 \in \mathcal{S},\ \exists u \in \mathbb{U},\ \exists w \in \mathcal{W},\ \text{such that } x = f(x_0, u, w) \Big\}. \tag{2.22}$$

Thus, the successor set $\text{Suc}(\mathcal{S}, \mathcal{W})$ contains all states that can be reached in one step from $x_0 \in \mathcal{S}$ under at least one admissible input and at least one admissible disturbance.

The *robust precursor set* to the set $\mathcal{S}$ is defined as [3]

$$\text{Pre}(\mathcal{S}, \mathcal{W}) = \Big\{ x \in \mathbb{R}^{n_x} : \exists u \in \mathbb{U}\ s.t.\ f(x, u, w) \in \mathcal{S},\ \forall w \in \mathcal{W} \Big\}. \tag{2.23}$$

For a system with inputs, $\text{Pre}(\mathcal{S}, \mathcal{W})$ is the set of states which can be robustly driven into the target set $\mathcal{S}$ in one time step for all admissible disturbances.

For a given initial set $\mathbb{X}_0 \subseteq \mathbb{X}$, the *N-step robust reachable set* $\mathbb{R}_N(\mathbb{X}_0, \mathcal{W})$ of the system (2.21) subject to the constraints (2.12) is defined recursively as [3]

$$\mathcal{R}_{i+1}(\mathbb{X}_0, \mathcal{W}) = \text{Suc}(\mathcal{R}_i(\mathbb{X}_0, \mathcal{W}), \mathcal{W}) \cap \mathbb{X},\ \mathcal{R}_0(\mathbb{X}_0, \mathcal{W}) = \mathbb{X}_0,\ i = 0, \ldots, N-1. \tag{2.24}$$

Thus, for every $x_0 \in \mathbb{X}_0$, there exists an admissible input sequence such that the state at time N lies in $\mathcal{R}_N(\mathbb{X}_0, \mathcal{W})$ for all disturbances.

For a given target set $\mathcal{S} \subseteq \mathbb{X}$, the *N-step robust controllable set* $\mathcal{K}_N(\mathcal{S}, \mathcal{W})$ of the system (2.21) subject to the constraints (2.12) is defined recursively as [3]

$$\mathcal{K}_j(\mathcal{S}, \mathcal{W}) = \text{Pre}(\mathcal{K}_{j-1}(\mathcal{S}, \mathcal{W}), \mathcal{W}) \cap \mathbb{X},\ \mathcal{K}_0(\mathcal{S}, \mathcal{W}) = \mathcal{S},\ j \in \{1, \ldots, N\}. \tag{2.25}$$

All states x_0 belonging to the N-Step robust controllable set $\mathcal{K}_N(\mathcal{S}, \mathcal{W})$ can be robustly driven, through a time-varying control law, to the target set $\mathcal{S}$ in N steps, while satisfying input and state constraints for all possible disturbances.

A set $\mathcal{C} \subset \mathbb{R}^n$ is a *robust control invariant set* for the system (2.21) subject to the constraints (2.12), if $\mathcal{C} \subseteq \mathbb{X}$ and

$$x_k \in \mathcal{C} \ \Rightarrow\ \exists u_k \in \mathbb{U} : f(x_k, u_k, w_k) \in \mathcal{C},\ \forall w_k \in \mathcal{W}. \tag{2.26}$$

The set $\mathcal{C}_\infty \subset \mathbb{R}^n$ is called a *maximal robust control invariant set* if it is not a strict subset of any other robust control invariant set.

2.2 Model Predictive Control (MPC)

2.2.1 Nominal MPC

A receding horizon controller, in which the finite time optimal control law is computed by solving a CFTOCP online, is usually referred to as MPC [3]. The key idea of the receding horizon controller is to shift the optimization horizon forward in time, as illustrated in Fig. 2.1.

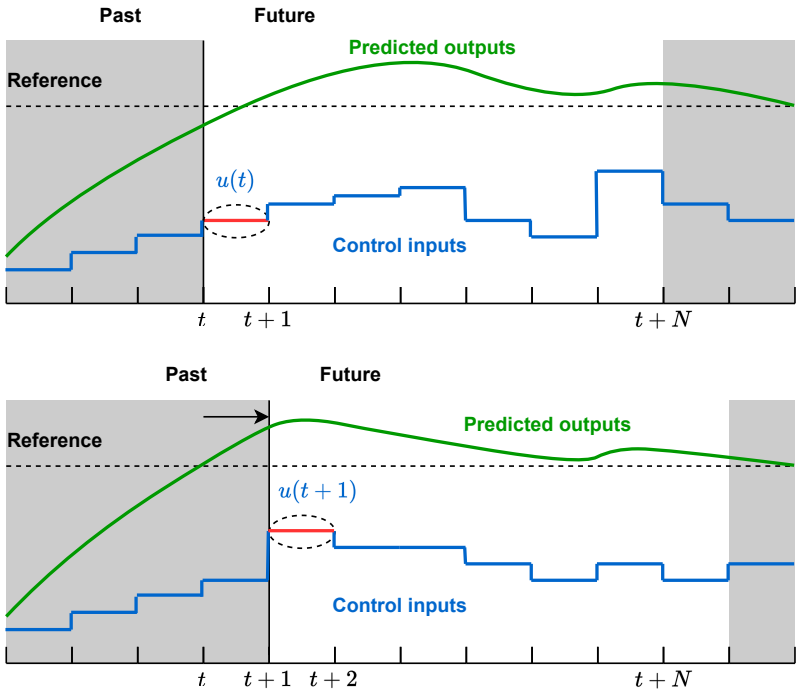


Fig. 2.1 Receding horizon control at current (top) and next (bottom) steps [3]

Considering the regulation of the DLTI system (2.11) to a desired point, based on the current state $x(t)$, an open-loop CFTOCP is solved

$$J^\star(x_t) = \min_{U_t} \; J_f(x_{t+N}) + \sum_{k=0}^{N-1} J_q(x_{t+k}, u_{t+k}) \tag{2.27a}$$

$$\text{s.t.} \; x_{t+k+1|t} = Ax_{t+k|t} + Bu_{t+k|t}, \;\; k = 0, \ldots, N-1, \tag{2.27b}$$

$$x_{t+k|t} \in \mathbb{X}, \;\; k = 1, \ldots, N, \tag{2.27c}$$

$$u_{t+k|t} \in \mathbb{U}, \;\; k = 0, \ldots, N-1, \tag{2.27d}$$

$$x_{t+N|t} \in \mathbb{X}_f, \tag{2.27e}$$

$$x_{t|t} = x(t). \tag{2.27f}$$

Here, $U_t = \begin{bmatrix} u_{t|t}^\top & u_{t+1|t}^\top & \cdots & u_{t+N-1|t}^\top \end{bmatrix}^\top$ represents the stacked vector of predicted control inputs over the prediction horizon N. $x_{t+k|t}$ represents the k-th predicted state vector

at time t, which is obtained by starting from the current time step t and applying control sequences to the system model (2.27b). Similarly, $u_{t+k|t}$ represents the k-th computed input vector at time t. Let the optimal solution to CFTOCP (2.27) at time t be $U^{\star}_{t\to t+N|t} = \begin{bmatrix} {u^{\star}_{t|t}}^{\top} & {u^{\star}_{t+1|t}}^{\top} & \dots {u^{\star}_{t+N-1|t}}^{\top} \end{bmatrix}^{\top}$.

The first element is applied to the system (2.11):

$$u(t) = u^{\star}_{t|t}. \tag{2.28}$$

CFTOCP (2.27) is then repeatedly solved with the moving horizon, based on newly collected measurements.

Additionally, since the objective function, the system dynamics, and the constraints in (2.27) are time-invariant, the solution to this problem is a time-invariant function of the initial state $x(t)$. Therefore, to simplify the notation, we replace t with 0 in (2.27), and omit $|t$, rewriting it as:

$$\begin{aligned}
J^{\star}(x_0) = \min_{U_0} \;\; & J_f(x_N) + \sum_{k=0}^{N-1} J_q(x_k, u_k) && \text{(2.29a)}\\
\text{s.t.} \;\; & x_{k+1} = Ax_k + Bu_k, \;\; k = 0, \dots, N-1, && \text{(2.29b)}\\
& x_k \in \mathbb{X}, \;\; k = 1, \dots, N, && \text{(2.29c)}\\
& u_k \in \mathbb{U}, \;\; k = 0, \dots, N-1, && \text{(2.29d)}\\
& x_N \in \mathbb{X}_f, && \text{(2.29e)}\\
& x_0 = x(0)\ . && \text{(2.29f)}
\end{aligned}$$

Here, U_0 represents the computed control sequence at the current time step. x_0 represents the initial state at the current time step.

2.2.2 Robust and Stochastic MPC

According to how uncertainties are handled in the control formulation, MPC is typically classified into Robust MPC (RMPC) and SMPC categories [151]. RMPC employs a set-based description of uncertainty, assuming all disturbances lie within a bounded set and designing the controller to satisfy constraints for every admissible realization [152, 153]. RMPC formulations define such robust sets by characterizing the disturbance through compact sets and constructing robust positive invariant sets or robust reachable sets for the error dynamics. These sets enable robust MPC to maintain feasibility and satisfy constraints for every admissible disturbance [151]. In contrast, SMPC represents uncertainties probabilistically in the constraint formulation [153]. A probabilistic model, such as a known distribution, an estimated distribution, or samples-based representation, is used to describe the uncertainty. Constraints are enforced through chance constraints,

which restrict the probability of violation to remain below a specified threshold [133,134], typically resulting in less conservative control than RMPC [154].

2.2.3 Recursive feasibility

Depending on the initial condition, MPC may lead to states where the CFTOCP becomes infeasible due to the finite horizon considered. To ensure recursive feasibility, solving the MPC problem with an infinite horizon is ideal, but this is computationally intractable. Therefore, it is necessary to include appropriate terminal conditions that compensate for truncation errors introduced by the finite horizon in MPC, thereby guaranteeing recursive feasibility. The concept of recursive feasibility is defined as follows [3].

Definition 2.2.1. ***(Recursive Feasibility).*** *The MPC problem* (2.29) *is said to be* ***recursively feasible*** *if, whenever it is feasible at the initial state x_0, the closed-loop system under MPC feedback remains feasible for all $k \geq 0$.*

The terminal cost $J_f(x_N)$ in (2.29a), along with the terminal set $\mathbb{X}_f$ in (2.29e), serves as terminal conditions that help approximate the effect of the infinite horizon [155, 156]. In particular, the terminal set $\mathbb{X}_f$, usually chosen as a control invariant set or a robust control invariant set, ensures that the system can always stay within feasible regions. The terminal cost approximates the infinite-horizon cost-to-go, promoting closed-loop stability and improving performance. The terminal conditions need to be specifically designed for different applications [3, 151].

Remark 2.2.1. *While stability is another desirable property in MPC design and is often achieved together with recursive feasibility via terminal cost and terminal set design [151], it is not the main focus of this thesis.*

2.3 MPC formulations for autonomous highway driving

2.3.1 Objective function, constraints and uncertainty handling

Considering the nominal formulation of a CFTOCP (2.29) in MPC, the objective function (2.29a) can be designed by incorporating various goals, such as tracking a reference trajectory, reducing energy consumption, or reaching a destination quickly [7,54,83,157]. In parallel with the objective design, constraints play a critical role in ensuring safe and feasible vehicle motion. Traffic rules such as speed limits and lane boundaries are typically treated as hard constraints that the EV must strictly obey [7]. Safety constraints that capture vehicle interactions are also usually modeled as hard constraints [7, 11]. However, some requirements may instead be modeled as soft constraints to retain feasibility, meaning they can be relaxed when necessary through slack variables [83, 84]. Whether a specific objective is implemented as a hard or soft constraint depends on the application

needs [83]. Additionally, the vehicle's kinematic or dynamic model in (2.29b), which may also be nonlinear, imposes further constraints on the system [11, 158]. All of these constraints together define the feasible state set $\mathbb{X}$ and input set $\mathbb{U}$ of (2.29), ensuring that the resulting control inputs remain physically realizable.

The terminal constraint in (2.29e) ensures that the predicted state at the end of the horizon lies in a region where safety is maintained, thereby guaranteeing safe future operation [152, 159]. The recursive feasibility of MPC can then be established even in uncertain environments [83, 115]. For the application of autonomous driving, a common strategy to define a terminal set is using the reachable set derived from vehicle dynamics and safety considerations [160, 161]. For instance, in [160], a terminal set for a LV-EV pair is constructed by first defining a robust positive invariant set enforcing three conditions: (i) the EV's speed must not exceed the LV's speed, (ii) its maximum possible deceleration must be lower than that of the LV, (iii) a required safety distance must be maintained. This set is then iteratively expanded to obtain an inner approximation of the maximal robust control invariant set, from which the terminal set is selected. Another representative approach to guarantee safety in the presence of multiple moving obstacles is presented in [162]. This method defines the terminal set as a robust positive invariant set under a pre-stabilizing feedback law. The EV's occupancy over the horizon is over-approximated using set-based dynamics to account for disturbances, forming a tube that ensures all predicted states and inputs satisfy tightened constraints. Recursive feasibility is preserved by assuming the existence of sets that provide consistent EV occupancy across time steps, so each newly computed tube lies within the previously predicted one, and that each obstacle's actual position remains within its predicted set. Additionally, without relying on a terminal set, an alternative option for recursive feasibility guarantee is to hand-craft a back-up controller, which gets invoked whenever the optimal control problem is infeasible [163].

2.3.2 Scenario-based MPC (SCMPC)

As a variant of SMPC, SCMPC approximates system uncertainties using a finite set of representative scenarios [164]. These scenarios are generated as independent and identically distributed (i.i.d.) samples from the underlying uncertainty distribution, ensuring that they capture the statistical properties of the uncertainty [136].

In the context of automated driving, each scenario represents a specific evolution of the traffic environment over the prediction horizon while capturing interactions between the EV and the TVs [115]. Let the number of surrounding TVs be V. The state information of each TV is denoted with a superscript '(TV)'. A scenario is defined as a tuple that includes the predicted state sequences of all V TVs over the prediction horizon:

$$\{x^{(\mathrm{TV1})}_{0,\dots,N},\ x^{(\mathrm{TV2})}_{0,\dots,N},\ \dots,\ x^{(\mathrm{TV}V)}_{0,\dots,N}\}. \tag{2.30}$$

Scenarios in SCMPC provide the basis for constructing safety constraints that ensure robust behavior of the EV under varying and uncertain traffic conditions. Suppose there are n_s scenarios, the related feasible state set under the safety constraints of scenario s is denoted with $\mathbb{X}^{(\text{safety},s)}$. Then, the feasible set $\mathbb{X}$ satisfies:

$$\mathbb{X} \subseteq \bigcap_{s=1}^{n_s} \mathbb{X}^{(\text{safety},s)}. \tag{2.31}$$

Given that the objective function is in a squared Euclidean form, the controller aims to track a reference trajectory while minimizing control effort. The general framework of SCMPC under this setup is formulated as follows:

$$J^\star(x_0) = \min_{U_0} \quad \sum_{k=0}^{N-1} \|x_{k+1} - x_{\text{ref},k+1}\|_{W_Q}^2 + \|u_k\|_{W_R}^2 \tag{2.32a}$$

$$\text{s.t.} \quad x_{k+1} = f(x_k, u_k), \quad k = 0, \ldots, N-1, \tag{2.32b}$$

$$x_k \in \mathbb{X}, \quad k = 1, \ldots, N, \tag{2.32c}$$

$$u_k \in \mathbb{U}, \quad k = 0, \ldots, N-1, \tag{2.32d}$$

$$x_N \in \mathbb{X}_f, \tag{2.32e}$$

$$x_0 = x(0), \tag{2.32f}$$

where $x_{\text{ref},k+1}$ represents the reference trajectory for the EV. The reference trajectory is designed according to the desired control task, such as tracking a target speed, maintaining a lane, or following a high-level planned path.

Different SCMPC methods employ distinct optimization formulations, especially regarding scenario generation and the formulation of scenario-based safety constraints. Representative approaches for scenario generation include the use of motion modes, such as car-following and lane-change models [116, 119, 165, 166]. In the automated driving application of [164], the motion modes 'lane-keeping', 'lane-change-right', and 'lane-change-left' are considered for each TV. A learning-based method identifies the maneuver each TV is likely to perform from a real-world dataset, from which the probability distributions of motion parameters are also estimated, while mode transitions are modeled using HMMs. Scenario trajectories are then generated by sampling the parameters for each mode. The number of scenarios is required to satisfy a lower bound determined by the prescribed risk level. An alternative approach bypasses predefined motion modes and instead evaluates TV behaviors at each step based on safety-related indices (e.g., time-to-collision) [129]. For each EV tactical decision, a decision-conditioned scenario tree is constructed, where plausible TV reactions are enumerated, and their likelihoods are derived from these indices. An SCMPC problem is then solved over each scenario tree, tracking a reference velocity while enforcing safety constraints. The EV selects the

tactical option with minimal expected cost, and the resulting trajectory is executed by a lower-level tracking MPC. Beyond these methods, scenario trees can also be generated using alternative approaches. Game-theoretic interaction models [126–128] and data-driven behavior models [130,131,167] have also been proposed for interaction-aware traffic prediction, offering additional possibilities for scenario generation.

Based on the predicted motion of TVs, the EV is required to respond safely under all generated scenarios. These scenarios can be assigned probabilities [133, 152, 164, 165] or treated equally [129]. In particular, in the SCMPC proposed in [164], each TV's reachable set over the prediction horizon under a given mode is approximated as a convex hull. Mode-dependent safety distance requirements are then incorporated into the SCMPC formulation via chance constraints, ensuring that the EV maintains a safe separation from these convex hulls with a prescribed confidence level. The convex-hull overapproximation reduces the available safety margin, and the confidence level directly affects the conservativeness of the resulting control inputs. While in [165], the unsafe region associated with a TV is approximated by combining multiple ellipses, each corresponding to the TV's predicted position under a different motion mode. The EV is required to maintain a safe distance from this combined ellipse region with a predefined probability level. To enforce this requirement, the safety-based chance constraints are reformulated into deterministic constraints by generating a sufficient number of sampled scenarios that capture the TV's motion uncertainty under a specified trajectory risk parameter. Approximated combination of ellipses also enlarge the considered unsafe area, potentially increasing the conservativeness of the resulting control inputs. In the approach of [168] specifically, the assigned scenario probabilities are further used in the cost function, where the controller minimizes the expected total scenario cost over the prediction horizon. Alternatively, in [129], a separate SCMPC is solved for each EV tactical decision using a corresponding decision-conditioned scenario tree. Within each SCMPC, all branches of the scenario tree are treated as equally relevant possible evolutions of traffic, and the safety constraints must be satisfied for every branch. This ensures that the computed control actions remain safe under all traffic realizations.

2.3.3 Contingency MPC (CMPC)

CMPC is an extension of MPC that explicitly accounts for potential future events or circumstances, which are referred to as contingencies. A contingency is defined as a possible future event or circumstance that cannot be predicted with certainty [148]. Two key characteristics distinguish contingencies from general uncertainties: one is that their risk of occurrence is situational, and can often be recognized through context or sensors, and another is that their high salience demands anticipation and the discrete focus of a dedicated safety plan [169]. The consideration of contingency scenarios is an important aspect of ensuring safety in automated driving applications [170]. For example, EVs must handle

unexpected scenarios that human drivers routinely encounter. These include safely maneuvering in response to an abrupt vehicle cut-in or a sudden hard braking by the LV. The consideration of a contingency might affect both the constraint and objective formulations of the control problem. For instance, in the case of a potential sudden cut-in by another vehicle, collision avoidance becomes the highest priority. As a result, the original objectives defined for nominal scenarios, such as passenger comfort, energy efficiency, or time optimality, may be overridden. Highly dynamic maneuvers, which would typically be undesirable, may become acceptable or even necessary to ensure safety [169].

CMPC is formulated by augmenting the typical MPC structure with the second type of control inputs $\breve{u}_k$, which account for the contingency scenario and enable the generation of a safe motion trajectory in response to it. The structure of CMPC is illustrated in the Fig. 2.2 below.

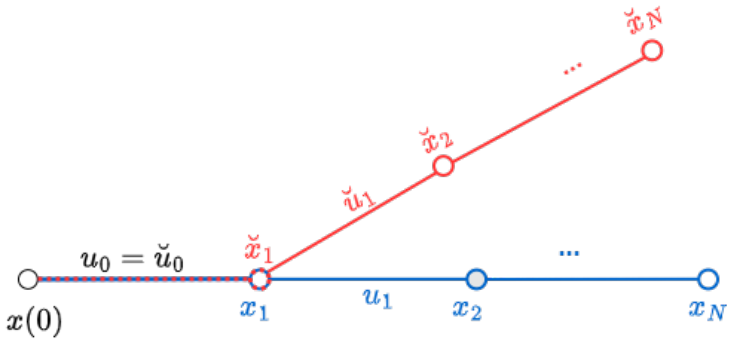


Fig. 2.2 CMPC structure

Both the nominal and contingency trajectories originate from the current measured state $x(0)$, while terminating at x_N and $\breve{x}_N$, respectively. The nominal state trajectory, denoted x_k, $k = 1, 2, \ldots, N$, represents the states the system is expected to follow under normal conditions by applying the nominal inputs u_k. In parallel, $\breve{x}_k$ and $\breve{u}_k$ define the contingency trajectory and input, respectively, which are subject to additional constraints imposed by the contingency scenario. These constraints reflect potential hazards that are not considered in the nominal trajectory. A critical feature of CMPC is that the root inputs u_0 and $\breve{u}_0$ are constrained to be identical. This equality condition ensures that the first control input applied is both performance-optimized under the nominal scenario and robust against the contingency scenario.

The general formulation of a CFTOCP under the CMPC framework is as follows:

$$
\begin{aligned}
J^\star(x_0) = \min_{U_0} \quad & J(x_0, U_0) && \text{(2.33a)}\\
\text{s.t.} \quad & x_{k+1} = f(x_k, u_k), \;\; k = 0, \dots, N-1, && \text{(2.33b)}\\
& \breve{x}_{k+1} = f(\breve{x}_k, \breve{u}_k), \;\; k = 0, \dots, N-1, && \text{(2.33c)}\\
& x_{k+1} \in \mathbb{X}, \;\; u_k \in \mathbb{U}, \;\; k = 0, \dots, N-1, && \text{(2.33d)}\\
& \breve{x}_{k+1} \in \breve{\mathbb{X}}, \;\; u_k \in \breve{\mathbb{U}}, \;\; k = 0, \dots, N-1, && \text{(2.33e)}\\
& x_N \in \mathbb{X}_f, && \text{(2.33f)}\\
& \breve{x}_N \in \breve{\mathbb{X}}_f, && \text{(2.33g)}\\
& u_0 = \breve{u}_0, && \text{(2.33h)}\\
& x_0 = \breve{x}_0 = x(0), && \text{(2.33i)}
\end{aligned}
$$

where the feasible state and input sets under the nominal scenario are denoted by $\mathbb{X}$ and $\mathbb{U}$, while $\breve{\mathbb{X}}$ and $\breve{\mathbb{U}}$ denote the feasible sets under the contingency scenario. The associated terminal sets for the nominal and contingency horizons are $\mathbb{X}_f$ and $\breve{\mathbb{X}}_f$, respectively.

The definition of a contingency scenario is the key aspect that differentiates CMPC-based approaches. In safety-critical automated driving applications, common choices for the contingency scenario include modeling the LV's maximum braking [150], or the sudden appearance of obstacles [148]. Beyond the specific contingency definitions, CMPC methods also differ in several structural aspects. One distinguishing factor is the number and type of contingency branches considered. Some approaches focus on a single hypothesis [150], while others incorporate multiple contingency hypotheses representing different critical events, as seen in multi-branch contingency formulations [171, 172]. CMPC approaches also vary in their planning-horizon structure. While some use identical horizons for both the nominal and contingency branches [150], others employ shortened contingency horizons to reduce computational load [148].

Remark 2.3.2. *In this thesis, the **MPC with interaction-aware predictions-based** controllers integrate approaches from both SCMPC and CMPC. Traffic uncertainties are captured using scenario-based formulations, while a dedicated contingency scenario is incorporated to guarantee the recursive feasibility of the resulting optimization problems.*

2.4 Summary

This chapter reviews the background of the MPC approaches that will be developed. Specifically, the mathematical formulations for nominal MPC, SCMPC, and CMPC are presented, highlighting their structural differences and the types of problems each is designed to solve. These formulations will serve as the foundation for the design of the control algorithms introduced in the upcoming chapters.

CHAPTER 3

Generation of interaction-aware traffic scenarios

3.1 Problem statement

The motion decision of the EV at each time step depends on the maneuvers of the surrounding TVs, and vice versa. Properly modeling their interaction over a certain horizon remains challenging. A common approach to modeling a vehicle's response is to imitate the behavior of rational human drivers: first predicting the motion intentions of the other vehicles of interest based on their current relative distance or velocity, then assuming those vehicles will maneuver as predicted, and finally making decisions accordingly to maintain safety. However, due to uncertainties in real vehicles' motion styles, the predicted motion maneuvers may differ from the actual ones. Nevertheless, because safety is the highest priority for vehicles, drivers tend to make safe decisions despite these uncertainties.

To handle the uncertainty in vehicle driving styles, multi-modal modeling is a promising approach, as many existing models can effectively represent vehicle motion characteristics and interaction awareness [116, 119]. Accordingly, in this thesis, IMM-KF-based algorithms are employed to estimate motion states and their associated probabilities using multiple longitudinal and lateral interaction-aware motion models. These models are referred to as 'policy modes' in the following discussion. Based on the state estimation, predictions under different modes are subsequently generated while ensuring the safety of all vehicles.

3.2 Interaction-aware estimation and prediction

Accurately estimating and predicting the motion intentions of surrounding TVs is essential for the safe control of the EV. The KF serves as an effective tool for this purpose, as it provides optimal and unbiased state estimation for linear time-invariant systems. It also effectively attenuates measurement noise, handles incomplete or intermittent observations, and supports predictive state propagation. Furthermore, its recursive formulation enables efficient real-time implementation with low memory requirements.

The motion decision-making of a vehicle in the real-world can be regarded as a continuous process. However, discrete decision-making sequences often emerge, particularly

when vehicles change their maneuvers. For instance, if an LV initiates a lane change, it may trigger a deceleration or a lane change response from a following vehicle in the target lane. This highlights a causal relationship in vehicle behavior, i.e., some vehicles act first while others react, which is typically analyzed in the time domain. To capture this causality, we introduce a priority-based estimation and prediction framework in a discrete-time setting. In this approach, vehicles are assigned priority levels, and predictions are made sequentially accordingly. The predicted motion states of higher-priority vehicles are then used as inputs for estimating and predicting the behavior of lower-priority vehicles.

3.2.1 A short review of the Kalman Filter (KF)

The KF is a state estimator for a linear system:

$$x_{k+1} = F_k x_k + E_k + w_k, \tag{3.1a}$$
$$y_k = H_k x_k + \nu_k, \tag{3.1b}$$

where $k \in \mathbb{N}$ represents the current time step. $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_y}$ represent the system state and measurement vectors, respectively. F_k is the system matrix, E_k is the system input vector, and H_k is the measurement matrix. The process noise w_k and measurement noise ν_k are assumed to be zero-mean Gaussian with covariances Q and R, respectively.

The KF involves a prediction and an update step. The prediction step yields:

$$\hat{x}_{k-1}^{+} = F_{k-1} x_{k-1}^{-} + E_{k-1}, \tag{3.2a}$$
$$\hat{P}_{k-1}^{+} = F_{k-1} P_{k-1}^{-} F_{k-1}^{\top} + Q_{k-1}, \tag{3.2b}$$

where $\hat{x}_{k-1}^{+}$ is predicted (a priori) state estimate at time k based on the posterior estimate x_{k-1}^{-} from time $k-1$, and $\hat{P}_{k-1}^{+}$ is predicted state error covariance at time k computed from the posterior covariance P_{k-1}^{-} at time $k-1$. The update step is:

$$\Delta y_k = y_k - H_k \hat{x}_{k-1}^{+}, \tag{3.3a}$$
$$S_k = H_k \hat{P}_{k-1}^{+} H_k^{\top} + R_k, \tag{3.3b}$$
$$K_k = \hat{P}_{k-1}^{+} H_k^{\top} {S_k}^{-1}, \tag{3.3c}$$
$$\hat{x}_k^{-} = \hat{x}_{k-1}^{+} + K_k \Delta y_k, \tag{3.3d}$$
$$\hat{P}_k^{-} = (\boldsymbol{I} - K_k H_k) \hat{P}_{k-1}^{+}, \tag{3.3e}$$

with the innovation residual Δy_k and its covariance S_k, the Kalman gain K_k, the posterior state estimate $\hat{x}_k^{-}$, and the estimate covariance $\hat{P}_k^{-}$ at time k.

3.2.2 Interacting Multiple Model Kalman Filter (IMM-KF)

The IMM-KF extends the KF by operating multiple KFs in parallel, each associated with a distinct motion mode. Through probabilistic mode interaction, the IMM-KF fuses the outputs of these filters, dynamically weighting them based on the likelihood of each model given current observations. This mechanism allows the IMM-KF to switch between models or mix them as needed, enabling robust and adaptive state estimation in systems characterized by hybrid or switching dynamics. By effectively managing model uncertainty and leveraging inter-model interactions, the IMM-KF is well suited for EV control while accounting for vehicle interactions [122].

Considering the Markov jump linear systems as follows:

$$x_{k+1}^{(i)} = F_k^{(i)} x_k^{(i)} + E_k^{(i)} + w_k^{(i)}, \tag{3.4a}$$

$$y_k^{(i)} = H_k^{(i)} x_k^{(i)} + \nu_k^{(i)}, \tag{3.4b}$$

where i indicates the current active mode, $i = 1, 2, \ldots, M$, and M denotes the total number of considered modes. The transition from mode $m^{(i)}$ to $m^{(j)}$ is described through a Markov chain.

The state and error covariance estimates for each mode are first initialized based on the mode probabilities $\mu_{k-1}^{(i)}$, state estimation $\hat{x}_{k-1}^{(i)-}$, and error covariance estimation $\hat{P}_{k-1}^{(i)-}$ from the previous time step $k-1$. The transition probability $\pi^{(ij)}$ denotes the likelihood of transitioning from mode i to mode j. The mixing conditional mode probability $\mu_{k-1}^{(j|i)-}$ is then calculated based on the fused initial state estimation $\bar{x}_{k-1}^{(i)-}$, and error covariance estimation $\bar{P}_{k-1}^{(i)-}$ as follows:

$$c^{(i)} = \sum_{j=1}^{M} \pi^{(ji)} \mu_{k-1}^{(j)}, \tag{3.5a}$$

$$\mu_{k-1}^{(j|i)-} = \frac{\pi^{(ji)} \mu_{k-1}^{(j)}}{c^{(i)}}, \tag{3.5b}$$

$$\bar{x}_{k-1}^{(i)-} = \sum_{j=1}^{M} \mu_{k-1}^{(j|i)-} \hat{x}_{k-1}^{(j)-}, \tag{3.5c}$$

$$\bar{P}_{k-1}^{(i)-} = \sum_{j=1}^{M} \mu_{k-1}^{(j|i)-} \left[\hat{P}_{k-1}^{(j)-} + \left(\bar{x}_{k-1}^{(i)-} - \hat{x}_{k-1}^{(j)-} \right) \left(\bar{x}_{k-1}^{(i)-} - \hat{x}_{k-1}^{(j)-} \right)^{\top} \right]. \tag{3.5d}$$

The related state and covariance of each KF is then predicted with the system mode:

$$\hat{x}_{k-1}^{(i)+} = F_{k-1}^{(i)} \bar{x}_{k-1}^{(i)-} + E_{k-1}^{(i)}, \tag{3.6a}$$

$$\hat{P}_{k-1}^{(i)+} = F_{k-1}^{(i)} \bar{P}_{k-1}^{(i)-} F_{k-1}^{(i)\top} + Q_{k-1}^{(i)}. \tag{3.6b}$$

The prior state estimation $\hat{x}_{k-1}^{(i)+}$ and error covariance $\hat{P}_{k-1}^{(i)+}$ are thus obtained. According

to the current step's measurement $y_k^{(i)}$, the innovation residual $\Delta y_k^{(i)}$ and its covariance $S_k^{(i)}$ are then calculated:

$$\Delta y_k^{(i)} = y_k^{(i)} - H_k \hat{x}_{k-1}^{(i)+}, \tag{3.7a}$$

$$S_k^{(i)} = H_k \hat{P}_{k-1}^{(i)+} H_k^\top + R_k^{(i)}. \tag{3.7b}$$

By balancing the difference between the predicted mode and the observed measurement, the Kalman gain $K_k^{(i)}$ is computed to update the posterior predicted state $\hat{x}_k^{(i)-}$ and error covariance $\hat{P}_k^{(i)-}$ estimations:

$$K_k^{(i)} = \hat{P}_{k-1}^{(i)+} H_k^\top {S_k^{(i)}}^{-1}, \tag{3.8a}$$

$$\hat{x}_k^{(i)-} = \hat{x}_{k-1}^{(i)+} + K_k^{(i)} \Delta y_k^{(i)}, \tag{3.8b}$$

$$\hat{P}_k^{(i)-} = (\boldsymbol{I} - K_k^{(i)} H_k) \hat{P}_{k-1}^{(i)+}. \tag{3.8c}$$

Each mode's likelihood $L_k^{(i)}$ is calculated to update the probability $\mu_k^{(i)}$:

$$L_k^{(i)} = \frac{\exp(-\frac{1}{2} \Delta y_k^{(i)\top} {S_k}^{(i)-1} \Delta y_k^{(i)})}{\left| 2\pi S_k^{(i)} \right|^{1/2}}, \tag{3.9a}$$

$$\mu_k^{(i)} = \frac{c^{(i)} L_k^{(i)}}{\sum_{j=1}^{M} c^{(j)} L_k^{(j)}}. \tag{3.9b}$$

The final optimal state and error covariance estimations, $\hat{x}_k^-$ and $\hat{P}_k^-$, are obtained by combining the individual estimation with the updated probability:

$$\hat{x}_k^- = \sum_{i=1}^{M} \mu_k^{(i)} \hat{x}_k^{(i)-}, \tag{3.10a}$$

$$\hat{P}_k^- = \sum_{i=1}^{M} \mu_k^{(i)} [P_k^{(i)-} + (\hat{x}_k^- - \hat{x}_k^{(i)-})(\hat{x}_k^- - \hat{x}_k^{(i)-})^\top]. \tag{3.10b}$$

3.2.3 Intention-based policy mode

In contrast to many driver assistance systems, like ACC, autonomous driving on highways requires the EV to consider multiple TVs simultaneously. The TVs of interest are numbered sequentially as TV1, TV2, and so on. In the longitudinal direction, two policy modes are used: 'Velocity Tracking (VT)' and 'Distance Keeping (DK)'. The vehicle tracks a reference velocity in VT while keeping a safe distance from its LV in DK [122]. In the lateral direction, three policy modes corresponding to 'lane 1', 'lane 2', and 'lane 3' represent the target lanes of the vehicles, as illustrated in Fig. 3.1. Therefore, the total number of modes M is 6.

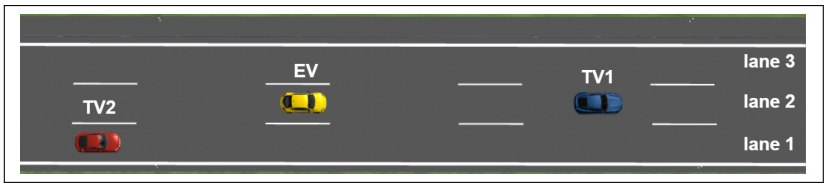


Fig. 3.1 Illustration of lanes and vehicles

The common state vector x_k in all modes at time k is

$$x_k \triangleq \left[\underbrace{p_{\text{lon},k} \quad v_{\text{lon},k} \quad a_{\text{lon},k}}_{x_{\text{lon},k}} \quad \underbrace{p_{\text{lat},k} \quad v_{\text{lat},k} \quad a_{\text{lat},k}}_{x_{\text{lat},k}}\right]^\top, \tag{3.11}$$

where $p_{*,k}, v_{*,k}, a_{*,k}$ are respectively the position, velocity, and acceleration in the corresponding direction, $* \in \{\text{lon}, \text{lat}\}$. The unknown reference variable $r_{\text{ref},k}$, i.e., reference velocity $v_{\text{ref},k}$ or the reference time gap $t_{\text{gap},k}$, is also included in the longitudinal policy mode to be estimated. Therefore, the full state vector z_k in each policy mode at time k is

$$z_k \triangleq \left[\underbrace{x_{\text{lon},k} \quad r_{\text{ref},k}}_{z_{\text{lon},k}} \quad \underbrace{x_{\text{lat},k}}_{z_{\text{lat},k}}\right]^\top. \tag{3.12}$$

Note that we use $x_k^{(\circledast)}$ or $z_k^{(\circledast)}$ to clarify the common or full state of a specific vehicle, where $\circledast \in \{\text{EV, TV, LV}\}$. To make the policies closer to the driver's real intention, two linear quadratic regulator LQR-based feedback controllers are included in longitudinal and lateral policy modes. The associated control gains are defined as follows:

$$K_{\text{lon}}^{(\diamond)} = \left[K_{\text{lon},1}^{(\diamond)} \quad K_{\text{lon},2}^{(\diamond)} \quad K_{\text{lon},3}^{(\diamond)}\right]^\top, \tag{3.13a}$$

$$K_{\text{lat}} = \left[K_{\text{lat},1} \quad K_{\text{lat},2} \quad K_{\text{lat},3}\right]^\top, \tag{3.13b}$$

where $\diamond \in \{\text{VT}, \text{DK}\}$. Denoting the sampling time as T, the discrete form of each policy mode at time k is

$$\underbrace{\begin{bmatrix} z_{\text{lon},k+1} \\ z_{\text{lat},k+1} \end{bmatrix}}_{z_{k+1}} = \underbrace{\begin{bmatrix} F_{\text{lon},k}^{(\diamond)} & \mathbf{0}_{4\times3} \\ \mathbf{0}_{3\times4} & F_{\text{lat},k}^{(\lambda)} \end{bmatrix}}_{F_k} \underbrace{\begin{bmatrix} z_{\text{lon},k} \\ z_{\text{lat},k} \end{bmatrix}}_{z_k} + \underbrace{\begin{bmatrix} E_{\text{lon},k}^{(\diamond)} \\ E_{\text{lat},k}^{(\lambda)} \end{bmatrix}}_{E_k} + w_k, \tag{3.14a}$$

$$y_k = \boldsymbol{I}_{7\times7} z_k + \nu_k, \tag{3.14b}$$

where $y_k \in \mathbb{R}^7$ and $\boldsymbol{I}_{7\times7}$ are the measurement vector and observation matrix. w_k and ν_k are assumed to be normally distributed process noise and measurement noise, with the

covariances Q and R. The longitudinal system matrices are

$$F_{\text{lon},k}^{(\text{VT})} = \begin{bmatrix} 1 & T & \frac{T^2}{2} & 0 \\ 0 & 1 - \frac{K_{\text{lon},2}^{(\text{VT})} T^2}{2} & T - \frac{K_{\text{lon},3}^{(\text{VT})} T^2}{2} & \frac{K_{\text{lon},2}^{(\text{VT})} T^2}{2} \\ 0 & -K_{\text{lon},2}^{(\text{VT})} T & 1 - K_{\text{lon},3}^{(\text{VT})} T & K_{\text{lon},2}^{(\text{VT})} T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.15a}$$

$$F_{\text{lon},k}^{(\text{DK})} = \begin{bmatrix} 1 - \frac{K_{\text{lon},1}^{(\text{DK})} T^3}{6} & -\frac{K_{\text{lon},1}^{(\text{DK})} T^2}{2} & -K_{\text{lon},1}^{(\text{DK})} T & 0 \\ T - \frac{K_{\text{lon},2}^{(\text{DK})} T^3}{6} & 1 - \frac{K_{\text{lon},2}^{(\text{DK})} T^2}{2} & -K_{\text{lon},2}^{(\text{DK})} T & 0 \\ \frac{T^2}{2} - \frac{K_{\text{lon},3}^{(\text{DK})} T^3}{6} & T - \frac{K_{\text{lon},3}^{(\text{DK})} T^2}{2} & 1 - K_{\text{lon},3}^{(\text{DK})} T & 0 \\ -\frac{K_{\text{lon},1}^{(\text{DK})} v_{\text{lead},k} T^3}{6} & -\frac{K_{\text{lon},1}^{(\text{DK})} v_{\text{lead},k} T^2}{2} & -K_{\text{lon},2}^{(\text{DK})} v_{\text{lead},k} T & 1 \end{bmatrix}^\top, \tag{3.15b}$$

and the LQR-related input matrices are

$$E_{\text{lon},k}^{(\text{VT})} = \mathbf{0}_{4\times 1}, \tag{3.16a}$$

$$E_{\text{lon},k}^{(\text{DK})} = \tag{3.16b}$$

$$\begin{bmatrix} \frac{K_{\text{lon},1}^{(\text{DK})} T^3}{6} - 1 & \frac{K_{\text{lon},2}^{(\text{DK})} T^3}{6} - T & \frac{K_{\text{lon},3}^{(\text{DK})} T^3}{6} - \frac{T^2}{2} & 0 \\ \frac{K_{\text{lon},1}^{(\text{DK})} T^2}{2} & \frac{K_{\text{lon},2}^{(\text{DK})} T^2}{2} - 1 & \frac{K_{\text{lon},3}^{(\text{DK})} T^2}{2} - T & 0 \\ K_{\text{lon},1}^{(\text{DK})} T & K_{\text{lon},2}^{(\text{DK})} T & K_{\text{lon},3}^{(\text{DK})} T - 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_{\text{lon},k}^{(\text{LV})} \\ v_{\text{lon},k}^{(\text{LV})} \\ a_{\text{lon},k}^{(\text{LV})} \\ 0 \end{bmatrix}. \tag{3.16c}$$

Consider the position $p^{(\lambda)}$ of the center line of the target lane λ, $\lambda \in \{1, 2, 3\}$, the lateral system and input matrices are

$$F_{\text{lat},k}^{(\lambda)} = \begin{bmatrix} -\frac{K_{\text{lat},1} T^3}{6} & -\frac{K_{\text{lat},2} T^3}{6} & -\frac{K_{\text{lat},3} T^3}{6} \\ -\frac{K_{\text{lat},1} T^2}{2} & -\frac{K_{\text{lat},2} T^2}{2} & -\frac{K_{\text{lat},3} T^2}{2} \\ -K_{\text{lat},1} T & -K_{\text{lat},2} T & -K_{\text{lat},3} T - 1 \end{bmatrix}, \tag{3.17a}$$

$$E_{\text{lat},k}^{(\lambda)} = \begin{bmatrix} \frac{K_{\text{lat},1} T^3}{6} p^{(\lambda)} \\ \frac{K_{\text{lat},1} T^2}{2} p^{(\lambda)} \\ K_{\text{lat},1} T p^{(\lambda)} \end{bmatrix}. \tag{3.17b}$$

3.2.4 IMM-KF-based interaction-aware estimation and prediction

Considering the interaction between vehicles, their motion states are estimated and predicted using the IMM-KF algorithm based on priority rules. Specifically, for vehicles in the same lane, the front vehicle is assigned higher priority, while for vehicles in different lanes, priority is given to the one with the higher speed. Moreover, the safety of the studied vehicle is ensured by maintaining a safe distance from the vehicles ahead while driving in the same lane. If the safety distance cannot be maintained within the prediction horizon based on the considered mode, the related mode probability $\mu_k^{(i)}$ in (3.9b) needs

to be modified to meet the required safety distance. To this end, an MIQP problem is introduced. After using IMM-KF and obtaining the posterior predicted state $\hat{x}_k^{(i)-}$ and error covariance $\hat{P}_k^{(i)-}$ estimations in (3.8b) and (3.8c), the following MIQP problem is solved:

$$\bar{y}_k^{(i)} = \min_{\breve{z}_k^{(i)-}} \quad \left\| \breve{z}_k^{(i)-} - \hat{z}_k^{(i)-} \right\|_{W_z}^2, \tag{3.18a}$$

$$\text{s.t. } \breve{z}_{t|k}^{(i)+} = \phi(t,1)\breve{z}_k^{(i)-} + \sum_{\delta=k+1}^{t} \phi(t,\delta) E_{\delta-1}, \tag{3.18b}$$

$$\phi(t,\delta) = \begin{cases} (\Pi_{\eta=\delta}^{t-1} F_\eta^\top)^\top & \text{if } t > \delta \\ \boldsymbol{I}_{7\times 7}, & \text{if } t = \delta \end{cases}, \tag{3.18c}$$

$$\bigwedge_{t\in\{k,k+1,\ldots,k+N\}} |\breve{p}_{\text{lon},t|k}^{(i)+} - p_{\text{lon},t|k}^{(\circ)}| \geq \frac{l + l^{(\circ)}}{2},$$

$$\text{if } |\breve{p}_{\text{lat},t|k}^{(i)+} - p_{\text{lat},t|k}^{(\circ)}| \leq \frac{b + b^{(\circ)}}{2}, \tag{3.18d}$$

where $\hat{z}_k^{(i)-}$ and $\breve{z}_k^{(i)-}$ represent the current posterior full state estimation and the corresponding modification. Their difference is represented by $\bar{y}_k^{(i)}$. $\breve{z}_{t|k}^{(i)+}$ represents state predictions over time steps $t = k+1, \ldots, k+1+N$, where N is the prediction horizon. $\circ$ indicates each vehicle that is prioritized over the studied vehicle, which may be a TV or an EV. l and b denote the vehicle's length and width, respectively.

Remark 3.2.3. *The objective of* (3.18) *is to obtain a modified state estimation that prevents collision. Specifically, without solving* (3.18)*, there is no feedback indicating whether a collision occurs based on the current reference state estimation* $\hat{r}_{ref,k}^{(i)-}$*. Therefore, to achieve collision-free prediction,* $\hat{r}_{ref,k}^{(i)-}$ *is primarily adjusted among all estimated states* $\hat{z}_k^{(i)-}$ *by appropriately selecting weight matrix* $W_z \succ 0$*.*

Denote the covariance of the cost in problem (3.18) as $\bar{S}_k^{(i)}$. The innovation residual $\Delta y_k^{(i)}$ and its covariance $S_k^{(i)}$ are then augmented as follows to yield the modified likelihood $\tilde{L}_k^{(i)}$:

$$y_{\text{aug},k}^{(i)} = \begin{bmatrix} \Delta y_k^{(i)} \\ \bar{y}_k^{(i)} \end{bmatrix}, \quad S_{\text{aug},k}^{(i)} = \begin{bmatrix} S_k^{(i)} & \mathbf{0} \\ \mathbf{0} & \bar{S}_k^{(i)} \end{bmatrix}, \tag{3.19a}$$

$$\tilde{L}_k^{(i)} = \frac{\exp(-\frac{1}{2} y_{\text{aug},k}^{(i)\top} {S_{\text{aug},k}}^{(i)-1} y_{\text{aug},k}^{(i)})}{\left| 2\pi S_{\text{aug},k}^{(i)} \right|^{1/2}}. \tag{3.19b}$$

Then, the policy mode probability is updated based on the modified likelihood $\tilde{L}_k^{(i)}$:

$$\tilde{\mu}_k^{(i)} = \frac{c^{(i)} \tilde{L}_k^{(i)}}{\sum_{j=1}^{M} c^{(j)} \tilde{L}_k^{(j)}}. \tag{3.20}$$

The state estimation and its covariance are then obtained according to the updated probability $\tilde{\mu}_k^{(i)}$ of the individual mode

$$\hat{x}_k^- = \sum_{i=1}^{M} \tilde{\mu}_k^{(i)} \hat{x}_k^{(i)-}, \tag{3.21a}$$

$$\hat{P}_k^- = \sum_{i=1}^{M} \tilde{\mu}_k^{(i)} [\hat{P}_k^{(i)-} + (\hat{x}_k^- - \hat{x}_k^{(i)-})(\hat{x}_k^- - \hat{x}_k^{(i)-})^\top]. \tag{3.21b}$$

Note that the state estimation of each policy mode remains as $\hat{z}_k^{(i)-}$. Only the state prediction is modified in terms of $\check{z}_k^{(i)-}$ and (3.18b).

3.2.5 Improved IMM-KF-based interaction-aware estimation and prediction

Conventionally, the transition probabilities in the IMM-KF are often fixed based on the prior information or heuristic assumptions. However, such conservative settings may result in suboptimal or inaccurate state estimations [173], particularly in dynamic traffic environments with rapidly changing behaviors. To address this limitation, the transition probability $\pi_k^{(ji)}$ is adaptively updated based on the difference in mode probabilities before and after solving the MIQP problem. Specifically, this update is informed by the modified mode probability $\bar{\mu}_k^{(i)}$, calculated from (3.20), and the original mode probability $\mu_k^{(i)}$, computed from (3.9b):

$$\Delta\mu_k^{(i)} = \bar{\mu}_k^{(i)} - \mu_k^{(i)}. \tag{3.22}$$

The difference between these two values is used to guide the adjustment of the transition probabilities, allowing the IMM-KF to better reflect the evolving motion intentions and interactions of surrounding vehicles. By identifying the change direction for each mode based on the mode probability difference $\Delta\mu_k^{(i)}$, the transition probability matrix is modified and normalized as follows:

$$\bar{\pi}_k^{(ji)} = \frac{1}{1 - \Delta\mu_k^{(i)}} \pi_k^{(ji)}, \tag{3.23a}$$

$$\tilde{\pi}_k^{(ji)} = \frac{1}{\sum_{i=1}^{M} \bar{\pi}_k^{(ji)}} \bar{\pi}_k^{(ji)}. \tag{3.23b}$$

Using the adaptively updated transition probability $\tilde{\pi}_k^{(j|i)}$, the modified fused mode

probability $\tilde{c}^{(i)}$ is computed:

$$\tilde{c}^{(i)} = \sum_{j=1}^{M} \tilde{\pi}^{(ji)} \mu_{k-1}^{(j)}. \tag{3.24}$$

The final modified mode probability $\tilde{\mu}_k^{(i)}$ is accordingly calculated by

$$\tilde{\mu}_k^{(i)} = \frac{\tilde{c}^{(i)} \tilde{L}_k^{(i)}}{\sum_{j=1}^{M} \tilde{c}^{(j)} \tilde{L}_k^{(j)}}. \tag{3.25}$$

In summary, to obtain more accurate and safe interaction-aware traffic predictions, the IMM-KF is enhanced by modifying both the mode likelihood distribution and the associated transition probabilities. These adjustments are based on an evaluation of the potential risk introduced by the estimated reference values. As a result, the probabilities of higher-risk modes and their associated transitions are reduced, guiding the prediction process toward safer behaviors.

3.3 Scenario generation

Based on these interaction-aware predictions, scenarios that represent traffic uncertainties can be generated. These scenarios capture the possible future evolutions of surrounding vehicles' states, incorporating their motion intentions and interactions. A scenario is defined as a tuple of motion maneuvers for all TVs. Assuming that the number of investigated TVs is V, then a total of M^V possible scenarios can be generated. $\mu_i^{(n)}$ is the probability of TV n with the policy mode i, $i \in \{1, 2, \ldots, 6\}$. Assuming statistical independence of each vehicle's no-collision prediction over the prediction horizon, then the probability of the scenario s is calculated by

$$\Pr(s) = \prod_{n=1}^{V} \mu_i^{(n)}, \quad s = 1, \ldots, M^V, \tag{3.26}$$

where $\sum_{s=1}^{M^V} \Pr(s) = 1$.

To ensure computational tractability, a finite set of representative scenarios is selected through filtering based on a predefined threshold $\underline{P}$. This results in a scenario set that balances accuracy with efficiency, providing sufficient diversity to cover the main behavioral uncertainties without overwhelming the optimization process. Specifically, scenarios with a probability less than $\underline{P}$ are not considered. The probability of the remaining scenarios is normalized by

$$\overline{\Pr}(s) = \frac{\Pr(s)}{1 - \sum_{\zeta=1}^{n_{\bar{s}}} \Pr(\zeta)}, \quad s = 1, \ldots, M^V - n_{\bar{s}}, \tag{3.27}$$

where $n_{\bar{s}}$ is the total number of scenarios with probability less than $\underline{P}$.

3.4 Simulation and discussion

To compare the proposed improved IMM-KF-based algorithm (Section 3.2.5) with the nominal IMM-KF-based algorithm (Section 3.2.4), the following traffic scenarios are designed to evaluate their ability to capture critical motion mode changes. The initial conditions of the base scenario are: $x^{(\mathrm{TV1})} = \begin{bmatrix} 440 & 28 & 0 & 26.94 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV2})} = \begin{bmatrix} 395 & 32 & 0 & 26.94 & 0 & 0 \end{bmatrix}^\top$, and $x^{(\mathrm{EV})} = \begin{bmatrix} 350 & 35 & 0 & 22.98 & 0 & 0 \end{bmatrix}^\top$. The initial scene is illustrated in Fig. 3.2. Over a simulation horizon of 35 s, TV1 maintains a constant velocity for the first 6 s, accelerates to 30 m/s over the next 2 s, and then maintains the velocity thereafter. TV2 switches to the lane where the EV stays from 6 s to 12 s. The EV drives behind the two TVs and react to their motion. Based on this scenario setup, we consider other scenarios by varying the TV1's longitudinal position within $[445\ \mathrm{m}, 450\ \mathrm{m}]$, and the EV's initial velocity within $[32\ \mathrm{m/s}, 28\ \mathrm{m/s}]$. A total of nine scenarios are obtained.

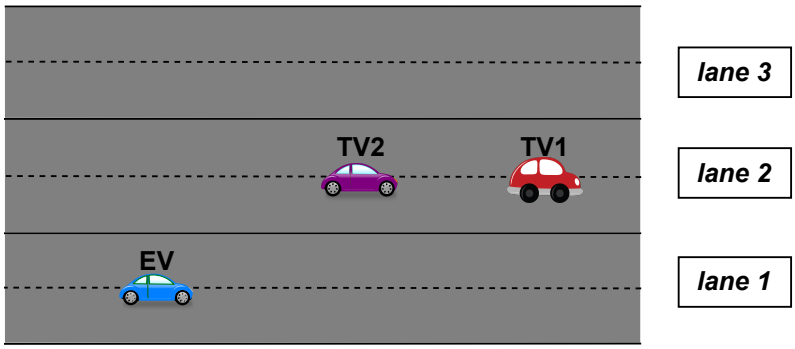


Fig. 3.2 Initial traffic scene used in IMM-KF-based algorithms comparison

The simulation results are analyzed based on the base scenario, with trajectories, longitudinal velocities, and lateral positions of the three vehicles shown in Fig. 3.3. The EV is controlled by a controller, but since the focus here is on motion estimation and prediction using IMM-KF, the controller itself is not discussed. From Fig. 3.3, we observe that after TV2 moves into lane 2 at 9 s, the EV decelerates and initiates a lane change around 10 s and 12 s, respectively. As TV1 moves slightly farther away, completing the maneuver within 3 s. The EV then remains in lane 1 for another 10s until the distance to TV1 decreases, prompting a return to lane 2 between $25 - 28$ s.

We evaluate the probability distribution results for the EV to demonstrate the improvement of the newly proposed IMM-KF-based algorithm compared to the nominal one. A representative example is shown in Fig. 3.4. To keep the main text concise, the remaining eight scenarios, which exhibit similar behaviors and reinforce the conclusions, are provided in Appendix A. From Fig. 3.4, we observe that the probability distribution aligns well with the actual motion of the EV. A notable difference between the two algorithms occurs between $10 - 16$s. During this interval, with the nominal IMM-KF-

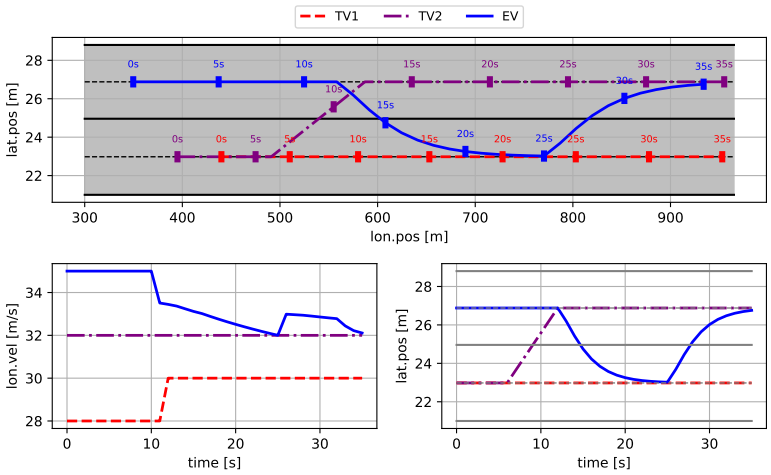


Fig. 3.3 Trajectories, longitudinal velocities and lateral positions of three vehicles

based algorithm, the most probable modes switch among DK-lane 2, VT-lane2, VT-lane 1 at $10.5, 11.5, 12.5$ s, respectively. In contrast, the improved IMM-KF captures the lane change earlier, identifying VT-lane 1 as the most probable mode at 11.5 s. This improvement arises because the proposed IMM-KF-based algorithm adaptively updates the transition probability matrix, which only changes when the motion mode changes. As a result, the advantage of the proposed IMM-KF is most apparent during policy mode switches, where accurate and timely prediction of vehicle motion is critical.

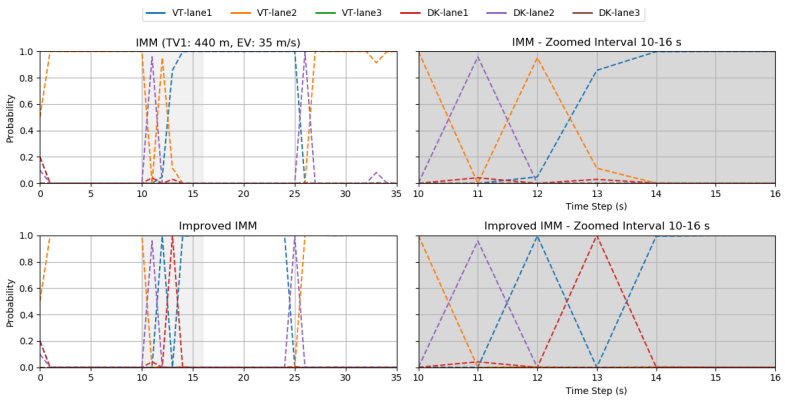


Fig. 3.4 Probability distribution comparison of two IMM-KFs in the base scenario

3.5 Summary

This section presents IMM-KF-based frameworks for estimating and predicting the motion maneuvers of traffic participants. Six distinct longitudinal and lateral policy modes are utilized to capture vehicles' possible motion patterns. By modeling multiple possible motion behaviors, fusing their outputs based on probabilities, and adjusting the probability distribution based on risk assessment, the IMM-KF provides reliable state estimation

in uncertain traffic environments. To further account for inter-vehicle dependencies, a priority-based estimation and prediction mechanism is introduced, where the estimated states of higher-priority vehicles influence the predictions of others. This interaction-aware prediction approach enhances the realism of the traffic representation. An improved IMM-KF is also proposed by adaptively updating both the mode likelihood and the transition probability matrix using the information from solving an MIQP problem. The resulting predictions are used to generate scenarios that represent probabilistic future traffic evolutions. These scenarios are critical for anticipating potential risks and are subsequently integrated into the controller design to enhance safety and robustness.

CHAPTER 4

SCMPC for autonomous highway driving

4.1 Problem statement

The interaction-aware traffic prediction module provides a finite number of scenarios that represent possible traffic evolutions. SCMPC offers a powerful framework to process these scenarios in a structured optimization problem. In the presence of the uncertainty of the traffic environment, the primary objective of the controller design is to ensure the safety of the EV, which can be achieved by incorporating collision avoidance based constraints. Meanwhile, SCMPC also accommodates other important driving considerations to ensure that the control actions are physically realizable, such as the vehicle's physical maneuvering limitations and applicable traffic regulations.

The goal of this thesis is to integrate planning and control tasks within a unified framework to improve control efficiency and avoid the time delay typically introduced by multi-layered architectures. To achieve this, we introduce a simplified reference trajectory generation module for the EV, rather than incorporate a sophisticated high-level path planner. This module is designed based on two key principles: maintaining the current velocity and following the center line of the target lane. To further account for the lateral uncertainty in the driving environment, multiple control modes are defined, each corresponding to a distinct target lane. Decision-making for the EV is embedded within the control process by evaluating and comparing the costs associated with each control mode. The mode that yields the minimum cost is selected, allowing the system to determine the optimal control action in a cost-driven manner. Through this approach, the tasks of trajectory planning, motion control, and decision-making are efficiently integrated and jointly optimized within the MPC framework.

4.2 SCMPC design and validation

By employing the IMM-KF-based algorithm introduced in Section 3.2.4, interaction-aware traffic scenarios are generated. Based on these predicted scenarios, a feasible trajectory for the EV is computed by solving a CFTOCP in a moving horizon fashion. The objective of the optimization problem is to track the reference trajectory with minimal

control effort while satisfying safety constraints, adhering to traffic rules, and maintaining driving comfort.

4.2.1 Triple-integrator vehicle model

A triple integrator is adopted to represent the motion dynamics of the EV in both longitudinal and lateral directions. This model captures the evolution of position, velocity, and acceleration over time, offering a balance between modeling accuracy and computational simplicity suitable for real-time control. The discrete-time representation for each direction is given by:

$$\underbrace{\begin{bmatrix} p^{(\mathrm{EV})}_{*,k+1} \\ v^{(\mathrm{EV})}_{*,k+1} \\ a^{(\mathrm{EV})}_{*,k+1} \end{bmatrix}}_{x^{(\mathrm{EV})}_{*,k+1}} = \underbrace{\begin{bmatrix} 1 & T & \frac{1}{2}T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} p^{(\mathrm{EV})}_{*,k} \\ v^{(\mathrm{EV})}_{*,k} \\ a^{(\mathrm{EV})}_{*,k} \end{bmatrix}}_{x^{(\mathrm{EV})}_{*,k}} + \underbrace{\begin{bmatrix} \frac{1}{6}T^3 \\ \frac{1}{2}T^2 \\ T \end{bmatrix}}_{B} \underbrace{j^{(\mathrm{EV})}_{*,k}}_{u^{(\mathrm{EV})}_{*,k}}, \tag{4.1}$$

where T is the prediction time step, and $* \in \{\mathrm{lon}, \mathrm{lat}\}$ indicates the longitudinal and lateral components, respectively. Here, $x^{(\mathrm{EV})}_{*,k+1}$ denotes the state vector at time step k, consisting of position, velocity, and acceleration, while $u^{(\mathrm{EV})}_{*,k}$ represents the jerk input $j^{(\mathrm{EV})}_{*,k}$.

To model the combined longitudinal and lateral dynamics in a compact form, (4.1) is rewritten as a block-structured system:

$$x^{(\mathrm{EV})}_{k+1} = \underbrace{\begin{bmatrix} A & 0_{3\times3} \\ 0_{3\times3} & A \end{bmatrix}}_{\bar{A}} x^{(\mathrm{EV})}_{k} + \underbrace{\begin{bmatrix} B \\ B \end{bmatrix}}_{\bar{B}} u^{(\mathrm{EV})}_{k}, \tag{4.2}$$

where $x^{(\mathrm{EV})}_{k+1} \in \mathbb{R}^6$ contains both longitudinal and lateral states, and $u^{(\mathrm{EV})}_{k} \in \mathbb{R}^2$ includes the corresponding jerk inputs. $\bar{A}$ and $\bar{B}$ are compact system and input matrices, respectively. This discrete-time double-axis model is used within the SCMPC framework for trajectory planning and control.

4.2.2 SCMPC framework

The proposed SCMPC scheme has two control modes: (i) the EV remains in the current lane and keeps its velocity (ii) the EV switches to the target lane and keeps its velocity. For each control mode, a CFTOCP is solved for a corresponding reference trajectory. The decision-making module then compares the resulting costs and selects the control mode with the lower cost. In addition to generated interaction-aware predicted scenarios, a so-called "worst-case" scenario is introduced to ensure the recursive feasibility of the CFTOCP. In this scenario, the LV is assumed to decelerate with its minimum allowable

acceleration throughout the prediction horizon. We introduce two sets of control inputs $U_0 = \{u_0^{(\mathrm{EV})}, ..., u_{N-1}^{(\mathrm{EV})}\}$ and $\breve{U}_0 = \{\breve{u}_0^{(\mathrm{EV})}, ..., \breve{u}_{N-1}^{(\mathrm{EV})}\}$. The first input set of sequences is calculated to avoid collision between the EV and the LV/TVs under the generated scenarios, and to minimize the cost function with associated states $x_k^{(\mathrm{EV})}$. The second set of sequences is obtained by considering the safety constraints under the "worst-case" scenario, with associated states $\breve{x}_k^{(\mathrm{EV})}$, and the terminal set of the states $\breve{\mathbb{X}}_f^{(\mathrm{EV})}$, which is detailed in Section 4.2.4. To guarantee the recursive feasibility, it is required that the first computed inputs $u_0^{(\mathrm{EV})}$ and $\breve{u}_0^{(\mathrm{EV})}$ must be equal, as stated in (4.3f). The CFTOCP is formulated as

$$J^\star(x_0) = \min_{U_0^{(\mathrm{EV})}, \breve{U}_0} \sum_{k=0}^{N-1} \left\| x_{k+1}^{(\mathrm{EV})} - x_{\mathrm{ref},k+1}^{(\mathrm{EV})} \right\|_{W_Q}^2 + \left\| u_k^{(\mathrm{EV})} \right\|_{W_R}^2, \tag{4.3a}$$

$$\text{s.t. } x_{k+1}^{(\mathrm{EV})} = \bar{A} x_k^{(\mathrm{EV})} + \bar{B} u_k^{(\mathrm{EV})}, \ \ k = 0, 1, ..., N-1, \tag{4.3b}$$

$$\breve{x}_{k+1}^{(\mathrm{EV})} = \bar{A} \breve{x}_k^{(\mathrm{EV})} + \bar{B} \breve{u}_k^{(\mathrm{EV})}, \ \ k = 0, 1, ..., N-1, \tag{4.3c}$$

$$x_k^{(\mathrm{EV})} \in \mathbb{X}^{(\mathrm{EV})}, \quad \breve{x}_k^{(\mathrm{EV})} \in \breve{\mathbb{X}}^{(\mathrm{EV})}, \ \ k = 0, 1, ..., N-1, \tag{4.3d}$$

$$u_k^{(\mathrm{EV})} \in \mathbb{U}^{(\mathrm{EV})}, \quad \breve{u}_k^{(\mathrm{EV})} \in \breve{\mathbb{U}}^{(\mathrm{EV})}, \ \ k = 0, 1, ..., N-1, \tag{4.3e}$$

$$u_0^{(\mathrm{EV})} = \breve{u}_0^{(\mathrm{EV})}, \tag{4.3f}$$

$$\breve{x}_N^{(\mathrm{EV})} \in \breve{\mathbb{X}}_f^{(\mathrm{EV})}, \tag{4.3g}$$

$$x_0^{(\mathrm{EV})} = \breve{x}_0^{(\mathrm{EV})} = x^{(\mathrm{EV})}(0). \tag{4.3h}$$

Here $x_{\mathrm{ref},k+1}^{(\mathrm{EV})}$ represents the reference state based on the relevant control mode. In the lane-change mode, it is assumed that the EV is permitted to change only one lane at a time, depending on the current traffic context. Consequently, the number of reference trajectories depends on the current lane of the EV. The weighting matrices $W_Q \in \mathbb{R}^{6\times6}$ and $W_R \in \mathbb{R}^{2\times2}$ are tunable and positive definite, used to penalize state deviation and control effort in the cost function, respectively. The feasible state sets $\mathbb{X}^{(\mathrm{EV})}$ and $\breve{\mathbb{X}}^{(\mathrm{EV})}$, as well as the input sets $\mathbb{U}^{(\mathrm{EV})}$ and $\breve{\mathbb{U}}^{(\mathrm{EV})}$ are defined by appropriate safety, physical, and traffic-rule-related constraints. These sets are described in detail in the following subsections.

Remark 4.2.4. *If there is no LV in reality, it is assumed that there is an LV far away from the EV.*

Remark 4.2.5. *Once the lane-change mode is activated, it stays active until the lane-change maneuvers are completed. For SCMPC in lane-change mode, except for considering the "worst-case" scenario of the LV in the target lane, the "worst-case" scenario of the LV in the previous lane is also considered during the EV's lane change.*

4.2.3 Constraint formulation

The EV's motion state and action are limited by traffic rules and driving comfort. The lateral position is constrained by the lane bounds $[l_{\text{lb}}, l_{\text{ub}}]$:

$$0 < v_{\text{lon},k}^{(\text{EV})}, \quad l_{\text{lb}} \leq p_{\text{lat},k}^{(\text{EV})} \leq l_{\text{ub}}, \tag{4.4a}$$

$$\underline{a}_{\text{lon}} \leq a_{\text{lon},k}^{(\text{EV})} \leq \overline{a}_{\text{lon}}, \quad \underline{a}_{\text{lat}} \leq a_{\text{lat},k}^{(\text{EV})} \leq \overline{a}_{\text{lat}}, \tag{4.4b}$$

$$\underline{j}_{\text{lon}} \leq j_{\text{lon},k}^{(\text{EV})} \leq \overline{j}_{\text{lon}}, \quad \underline{j}_{\text{lat}} \leq j_{\text{lat},k}^{(\text{EV})} \leq \overline{j}_{\text{lat}}, \tag{4.4c}$$

where $\underline{\bullet}$ and $\overline{\bullet}$ denote the minimum and maximum values of the associated variables.

A safe distance between the EV and the LV in the same lane is required:

$$d_k \geq \underline{d}, \tag{4.5}$$

and the safety distance $\underline{d}$ is computed by

$$\underline{d} = \tau v_{\text{lon},k}^{(\text{EV})} + \triangle d, \tag{4.6}$$

with the design parameters τ and $\triangle d$ [174]. If the reference point of all vehicles is in their respective center, for example, choose $\triangle d \geq \frac{l^{(\text{EV})}+l^{(\text{LV})}}{2}$, where $l^{(\text{EV})}$ and $l^{(\text{LV})}$ are the length of EV and LV. During the lane-change period, in addition to keeping a safe distance from the LVs in both the current and target lane, the EV also maintains a safe distance with the TV behind it in the target lane. The required safety distance also satisfies (4.6). The safety constraint under the generated scenario is based on (4.6), while the safety distance for considering the "worst-case" scenario collapses to $\triangle d$.

4.2.4 Recursive feasibility analysis for SCMPC

Given the uncertainty of the traffic environment, multiple interaction-aware scenarios are generated and considered for EV control. However, the safety of the EV cannot be fully guaranteed under such considerations if an unexpected traffic scenario occurs due to an emergency. To address this, we incorporate a "worst-case" scenario alongside the generated scenarios in the controller design, which provides a safety guarantee for the computed control strategy. The corresponding recursive feasibility of the optimization problem is discussed below.

The definition of recursive feasibility under the SCMPC framework is given by:

Definition 4.2.2. *(**Recursive Feasibility**) The SCMPC is **recursively feasible** if* (4.5) *always holds. Namely, in lane-keeping mode, a collision between the EV and the LV is always avoidable. In lane-change mode, no accident occurs between the EV and the other vehicles during the lane-change process, and then the EV remains safe in the target lane.*

If the safety constraints are satisfied in the "worst-case" scenario, it means that the EV can handle any traffic situation under the SCMPC framework. This capability is characterized by a parameter called the minimal stopping horizon, denoted as $\underline{N}$, and defined as follows:

Definition 4.2.3. *(**Minimal Stopping Horizon**) Given the initial velocity $v_{lon,0}^{(EV)}$ and the minimal acceleration $\underline{a}_{lon}^{(EV)}$ of the EV, the minimal stopping horizon $\underline{N} \in \mathbb{N}$ satisfies*

$$\underline{N} = \left\lceil \frac{v_{lon,0}^{(EV)}}{|\underline{a}_{lon}^{(EV)}|T} \right\rceil, \tag{4.7}$$

where $\lceil \bullet \rceil$ is defined as the smallest integer that is not smaller than a real number $\bullet$.

If we choose the prediction horizon $N \geq \underline{N}$, the terminal set $\breve{\mathbb{X}}_{\mathrm{f}}$ at the time t is

$$\breve{\mathbb{X}}_{\mathrm{f}}^{(\mathrm{EV})} \triangleq \{\breve{x}_{N|t}^{(\mathrm{EV})} | \breve{x}_{N|t}^{(\mathrm{EV})} = \begin{bmatrix} p_{\mathrm{lon},\underline{N}|t}^{(\mathrm{EV})} & 0 & 0 & p^{(\lambda)} & 0 & 0 \end{bmatrix}^{\top}\}, \tag{4.8}$$

where the stopping longitudinal position $p_{\mathrm{lon},\underline{N}|0}^{(\mathrm{EV})}$ of the EV is determined by its initial position $p_{\mathrm{lon},0}^{(\mathrm{EV})}$, initial velocity $v_{\mathrm{lon},0}^{(\mathrm{EV})}$ and minimal acceleration $\underline{a}_{\mathrm{lon}}^{(\mathrm{EV})}$. The terminal lateral position of the EV is the position of the center line of the target lane $p^{(\lambda)}$ under the specific control mode. Based on the general traffic situation and rules, we make the following assumptions.

Assumption 1. All vehicles only drive forward, and the EV is only responsible for front collisions.

Assumption 2. $u_k^{(\mathrm{EV})} = \begin{bmatrix} 0 & 0 \end{bmatrix}^{\top}$ is one element of the feasible set $\breve{\mathbb{U}}$.

The recursive feasibility of the SCMPC can then be proved.

Theorem 4.2.1. *If SCMPC is initially feasible, and the prediction horizon $N \geq \underline{N}$, then the controller is recursively feasible based on Assumptions 1 and 2.*

Proof. Let two initial control inputs of the generated normal scenarios and the "worst-case" scenario be $\{u_{0|0}^{(\mathrm{EV})}, u_{1|0}^{(\mathrm{EV})}, ..., u_{\underline{N}|0}^{(\mathrm{EV})}, ..., u_{N|0}^{(\mathrm{EV})}\}$ and $\{\breve{u}_{0|0}^{(\mathrm{EV})}, \breve{u}_{1|0}^{(\mathrm{EV})}, ..., \breve{u}_{\underline{N}|0}^{(\mathrm{EV})}, ..., \breve{u}_{N|0}^{(\mathrm{EV})}\}$. Choose the second control sequence as the initially feasible solution:

$$\{\breve{u}_{0|0}^{(\mathrm{EV})\star}, \breve{u}_{1|0}^{(\mathrm{EV})\star}, ..., \breve{u}_{\underline{N}|0}^{(\mathrm{EV})\star}, ..., \breve{u}_{N|0}^{(\mathrm{EV})\star}\}, \tag{4.9}$$

and its related state sequence is:

$$\{\breve{x}_{0|0}^{(\mathrm{EV})\star}, \breve{x}_{1|0}^{(\mathrm{EV})\star}, ..., \breve{x}_{\underline{N}|0}^{(\mathrm{EV})\star}, ..., \breve{x}_{N|0}^{(\mathrm{EV})\star}\}. \tag{4.10}$$

The terminal state $\breve{x}_{N|0}^{(\mathrm{EV})\star}$ is determined according to (4.8). We apply $\breve{u}_{0|0}^{(\mathrm{EV})\star}$ to the system (4.2), and obtain

$$x_1^{(\mathrm{EV})} = \bar{A}x_0^{(\mathrm{EV})} + \bar{B}\breve{u}_{0|0}^{(\mathrm{EV})\star} = \breve{x}_{1|0}^{(\mathrm{EV})\star}. \tag{4.11}$$

Then the following is a feasible solution for the MPC problem initialized at $x_1^{(\mathrm{EV})}$:

$$\begin{aligned}\{u_{1|1}^{(\mathrm{EV})}, u_{2|1}^{(\mathrm{EV})}, ..., u_{\underline{N}|1}^{(\mathrm{EV})}, ..., u_{N-1|1}^{(\mathrm{EV})}, u_{N|1}^{(\mathrm{EV})}\} = \\ \{\breve{u}_{1|0}^{(\mathrm{EV})\star}, \breve{u}_{2|0}^{(\mathrm{EV})\star}, ..., \breve{u}_{\underline{N}|0}^{(\mathrm{EV})\star}, ..., \breve{u}_{N-1|0}^{(\mathrm{EV})\star}, \begin{bmatrix}0 & 0\end{bmatrix}^\top\},\end{aligned} \tag{4.12}$$

where $\begin{bmatrix}0 & 0\end{bmatrix}^\top \in \breve{\mathbb{U}}^{(\mathrm{EV})}$. The related state sequence is

$$\begin{aligned}\{x_{2|1}^{(\mathrm{EV})}, x_{3|1}^{(\mathrm{EV})}, ..., x_{\underline{N}+1|1}^{(\mathrm{EV})}, ..., x_{N|1}^{(\mathrm{EV})}, x_{N+1|1}^{(\mathrm{EV})}\} = \\ \{\breve{x}_{2|0}^{(\mathrm{EV})\star}, \breve{x}_{3|0}^{(\mathrm{EV})\star}, ..., \breve{x}_{\underline{N}|0}^{(\mathrm{EV})\star}, ..., \breve{x}_{N|0}^{(\mathrm{EV})\star}, \bar{A}\breve{x}_{N|0}^{(\mathrm{EV})\star} + \bar{B}\cdot\begin{bmatrix}0 & 0\end{bmatrix}^\top\},\end{aligned} \tag{4.13}$$

where $\bar{A}\breve{x}_{N|0}^{(\mathrm{EV})\star} + \bar{B}\cdot\begin{bmatrix}0 & 0\end{bmatrix}^\top = \breve{x}_{N|0}^{(\mathrm{EV})\star}$. Both sequences are feasible for the MPC problem because they satisfy the dynamics and the constraints. □

4.3 Simulation and discussion

In IPG CarMaker, a real vehicle is represented by a simulated vehicle as the EV. To apply the control inputs longitudinal jerk $u_{\mathrm{lon},k}^{(\mathrm{EV})}$ and lateral jerk $u_{\mathrm{lat},k}^{(\mathrm{EV})}$ to the EV, they need to be converted into actions appropriate for the actuator, i.e., the engine torque or brake torque and steering angle. Particularly, the considered longitudinal conversion model is

$$a_{\mathrm{lon},k}^{(\mathrm{EV})} = \frac{i_{g,k}\xi}{m r_\omega} W_k. \tag{4.14}$$

Here $i_{g,k}$ represents the time-varying gear ratio, ξ is the drivetrain efficiency, m is the vehicle's overall mass, and r_w denotes the wheel radius. When $a_{\mathrm{lon},k}^{(\mathrm{EV})} > -0.2\ \mathrm{m/s^2}$, W_k represents the engine torque. Otherwise, it represents the brake torque. In IPG CarMaker, the range of the values for gas and brake are in $[0, 1]$, so the calculated engine or brake torque is divided by the related maximal value before being applied.

Based on the calculated $u_{\mathrm{lat},k}^{(\mathrm{EV})}$, the desired lateral states $\bar{p}_{\mathrm{lat},k}^{(\mathrm{EV})}, \bar{v}_{\mathrm{lat},k}^{(\mathrm{EV})}, \bar{a}_{\mathrm{lat},k}^{(\mathrm{EV})}$ are obtained by a triple integrator model with the sampling time T. The steering angle σ_k is computed from a PD controller plus the feedforward compensation by considering the lateral position error as follows:

$$\sigma_k = K_{\mathrm{P}} e_{\mathrm{lat},k}^{(\mathrm{EV})} + K_{\mathrm{D}} \frac{d e_{\mathrm{lat},k}^{(\mathrm{EV})}}{dt} + \bar{a}_{\mathrm{lat},k}^{(\mathrm{EV})}, \tag{4.15a}$$

$$e_{\mathrm{lat},k}^{(\mathrm{EV})} = p_{\mathrm{lat},k}^{(\mathrm{EV})} - \bar{p}_{\mathrm{lat},k}^{(\mathrm{EV})}, \tag{4.15b}$$

where $e_{\mathrm{lat},k}^{(\mathrm{EV})}$ is the lateral position error, which shows the deviation of the EV's real position from the desired one. $K_{\mathrm{P}}, K_{\mathrm{D}}$ are the PD control gains for tuning.

The presented approach is evaluated in three different traffic cases by controlling the

Table 4.1 Design parameters used in the SCMPC simulation study

$t_{\mathrm{s}}[\mathrm{s}]$	$T[\mathrm{s}]$	N	$\underline{a}_{\mathrm{lon}}^{(\mathrm{EV})}[\mathrm{m/s^2}]$
30.00	0.40	15	-4.00
$\overline{a}_{\mathrm{lon}}^{(\mathrm{EV})}[\mathrm{m/s^2}]$	$\underline{a}_{\mathrm{lat}}^{(\mathrm{EV})}[\mathrm{m/s^2}]$	$\overline{a}_{\mathrm{lat}}^{(\mathrm{EV})}[\mathrm{m/s^2}]$	$\underline{j}_{\mathrm{lon}}^{(\mathrm{EV})}[\mathrm{m/s^3}]$
1.50	-2.00	2.00	-5.50
$\overline{j}_{\mathrm{lon}}^{(\mathrm{EV})}[\mathrm{m/s^3}]$	$\underline{j}_{\mathrm{lat}}^{(\mathrm{EV})}[\mathrm{m/s^3}]$	$\overline{j}_{\mathrm{lat}}^{(\mathrm{EV})}[\mathrm{m/s^3}]$	$\underline{a}_{\mathrm{lon}}^{(\mathrm{LV})}[\mathrm{m/s^2}]$
5.50	-4.00	4.00	-4.00
ξ	$\tau[\mathrm{s}]$	K_{lat}	$p_c^{(1),(2),(3)}[\mathrm{m}]$
1.00	0.40	$[1.15, 3.39, 3.58]$	$\{-12.13, -8.38, -4.63\}$
$m[\mathrm{kg}]$	$\{l_f, l_r\}[\mathrm{m}]$	$l^{(\circledast)}[\mathrm{m}]$	$l_{\mathrm{b}}[\mathrm{m}]$
1463.00	$\{0.99, 1.54\}$	$\{4.47, 4.12, 4.28\}$	$\{-14, -10.25, -6.5, -2.75\}$
$r_\omega[\mathrm{m}]$	$b^{(\circledast)}[\mathrm{m}]$	W_R	W_Q
0.33	$\{1.97, 1.80, 1.80\}$	$\mathrm{diag}([0.1, 0.01])$	$0.01 * \mathrm{diag}([10, 1, 1, 10, 1, 1])$
$\underline{P}$	$\{K_{\mathrm{P}}, K_{\mathrm{D}}\}$		
0.075	$\{0.7, 1.5\}$		

EV under a high-fidelity environment, IPG CarMaker. We use DemoCar as the EV, IPG CompanyCar 2018 Blue, and Audi TT 2015 as the TVs.

To demonstrate the interaction awareness of traffic participants, we consider three different traffic scenarios, where the initial motion states of vehicles are set differently. Then, the TVs and the EV react differently to others. Specifically, when a slower vehicle is in front of a TV/EV, the latter might overtake or brake, depending on their relative distance, velocity, and surrounding vehicles. Therefore, these three cases, as comparisons with each other, are chosen to show the interaction results of vehicles from different perspectives. Table 4.1 shows the parameters used in the case studies. The total simulation time is denoted as t_{s}. The variable l_{b} represents the position of lane boundaries. Note that in the current simulation setup, if the EV is in lane 2, we only consider EV changes to lane 1, not to lane 1 or lane 3 in the lane-change mode of the SCMPC. When the SCMPC is infeasible in a certain control mode, we set the corresponding cost function value to 150. Once the EV starts to change lanes, it will complete the lane change without being able to change the decision. We assume it enters the new lane within 4 s. The cost of the lane-keeping mode during this time is also set to 150.

4.3.0.1 Case 1: the EV changes lanes and brakes

Fig. 4.1 gives the initial traffic scene of case 1. Specifically, the position, velocity, and acceleration of TV1, TV2, and EV are $x^{(\mathrm{TV1})} = \begin{bmatrix} 375 & 20 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV2})} = \begin{bmatrix} 288 & 20 & 0 & -12.13 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{EV})} = \begin{bmatrix} 350 & 18.9 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, respectively. Given the initial traffic situation, TV1 keeps its speed in the first 5 s and then decelerates with $-4\,\mathrm{m/s^2}$ until 7 s and eventually returns to $20\,\mathrm{m/s}$ in the following time. The priority

of the three vehicles is TV1, EV, TV2 in descending order. Therefore, the EV reacts to TV1, and TV2 reacts to TV1 and the EV.

Fig. 4.1 Initial traffic scene of Case 1 used for SCMPC structure validation

Figs. 4.2, 4.3, 4.4, and 4.5 show the simulated information of Case 1. From Fig. 4.5, we can see that the EV stays in lane 2 until 5.6 s due to a lower cost of lane-keeping mode. After that, as the lane-changing mode is less costly given that TV1 decelerates continuously, the EV changes to lane 1 at 8 s (as seen from Fig. 4.2). During the lane change, the EV decreases longitudinal velocity to around 17.5 m/s to avoid colliding with TV1. Then, the EV drives with this constant velocity for the rest of the time. As for TV2, as a vehicle behind the EV in the new lane, it decreases its velocity to the same velocity as the EV to maintain a safe distance from the EV.

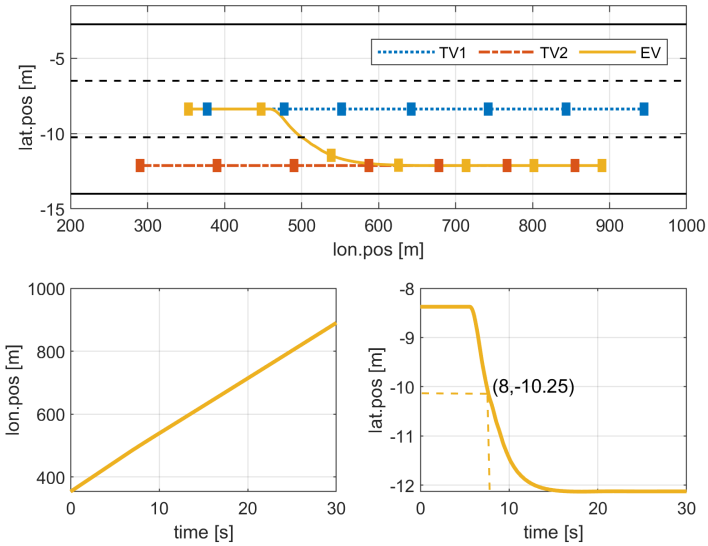


Fig. 4.2 Motion trajectories of vehicles and EV position in Case 1

Furthermore, the policy mode probabilities of TVs, as the basis of scenario generation, are provided and analyzed. Since the EV's policy mode probabilities influence the policy mode probability result of TV, whose priority is lower than the EV, its policy mode probabilities are also studied. The first three subplots of Fig. 4.6 show each vehicle's policy mode probabilities throughout the simulation time. The fourth subplot is the enlarged result of the third one during the time slot 4 s to 11.5 s. From the figure, we can see that TV1 performs VT maneuvers and stays in lane 2 almost all the time with the highest probability. The tracked velocity is time-varying according to TV1's actual motion behavior.

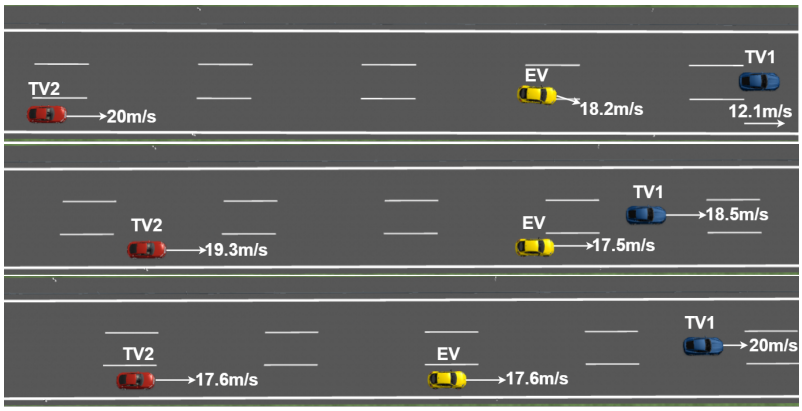


Fig. 4.3 Traffic scenes in Case 1 at 7.2 s, 13 s, and 24 s

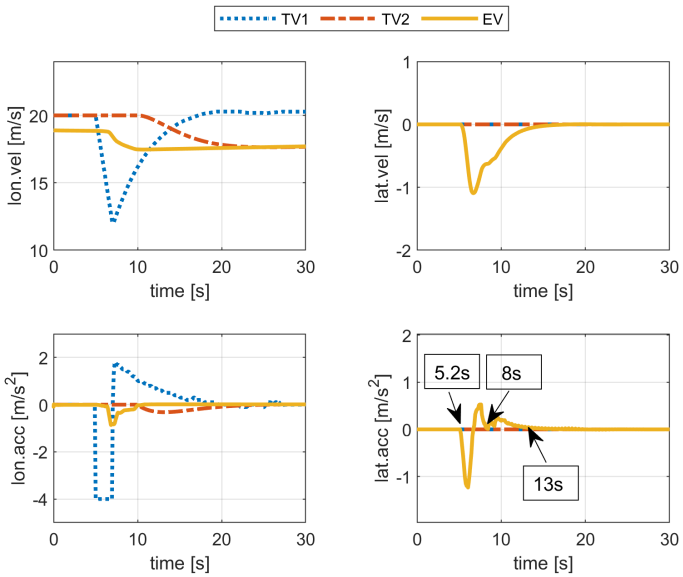


Fig. 4.4 Longitudinal and lateral velocity of all vehicles in Case 1

For the EV, the VT in lane 2 mode probability increases to almost 1 from the initial setting after 0.4 s. Then TV1, as the LV, decelerates between 2 s and 5 s. Their relative distance decreases continuously. Therefore, the probability of the VT in lane 2 mode decreases at 5.2 s, while the probability of the DK in lane 2 mode increases. In addition, based on the cost of the SCMPC in two control modes, the EV initiates a lane change at 5.6 s. After analyzing the relevant innovation residual $\bar{y}_k^{(i)}$ of these two most probable modes, we conclude that their related probabilities are sometimes modified from the MIQP during the simulation. For instance, at 6 s, the calculated 6 policy modes' state estimation errors between $\hat{z}_k^-$ and $\hat{z}_k^{(\text{proj})-}$ from solving the MIQP problem are $\begin{bmatrix} 0.001 & 0.76 & 0.001 & 0.001 & 0.12 & 0.001 \end{bmatrix}^\top$. 0.001 is a designed value representing the related state estimation error 0. The augmented estimation errors of driving in lane 2 in VT mode and DK mode, 0.76 and 0.12, decrease the corresponding modified probability in the IMM-KF. However, they are still the most likely ones. Solving the MIQP

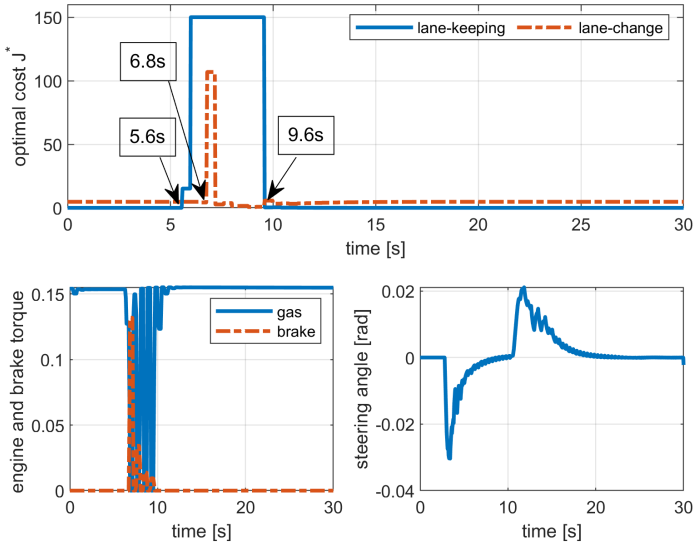


Fig. 4.5 Optimal cost and control actions of SCMPC in Case 1

problem reduces the related probability of the mode that might lead to a collision, making the generated scenarios more reliable. During 6.4 s to 7.2 s, the most likely mode is VT in lane 1, with a probability above 0.7, and the mode of DK in lane 1 has the second-highest probability of about 0.3. After that, the most probable mode fluctuates between staying in lane 1 or lane 2 until 10.4 s. For example, for the situation at 8 s, where the EV has entered lane 1 (as seen in Fig. 4.2), driving in lane 2 has the highest probability. The lateral acceleration value at this time point is around 0. By computing these two modes' innovation residual $\bar{y}_k^i$ in the lateral direction, $\bar{y}_k^{(1)} = \begin{bmatrix} -0.36 & 0.17 & 1.63 \end{bmatrix}^\top$, and $\bar{y}_k^{(2)} = \begin{bmatrix} -0.39 & 0.17 & 0.1 \end{bmatrix}^\top$, we obtain that the lateral acceleration error of staying in lane 1, equal to 1.63, is larger than that of staying in lane 2, equal to 0.1. Consequently, the second mode has a smaller innovation residual and a higher probability. From Fig. 4.4, we see that during lane change from 5.6 s to 9.6 s, and before the EV reaches the center position of the new lane at around 15 s, the EV's real lateral acceleration changes accordingly. It fluctuates, especially after becoming positive based on IPG CarMaker. The fluctuation makes obtaining accurate mode probability estimation difficult. Finally, the most probable mode is VT in lane 1 constantly due to the 'stable' lateral acceleration.

TV2 tracks the velocity and stays in lane 1 from the beginning to 8.4 s with probability 1. After that, since the EV comes to the front of TV2 in lane 1 at 8 s, the probability of this policy mode drops to 0.95, and the probability of keeping the distance in lane 1 increases to 0.05.

Moreover, to investigate the effect of TV characteristics on EV control, the TV2's driving style is changed to aggressive by tuning the interactive parameter in IPG Carmaker, while the remaining vehicle's initial conditions are the same.

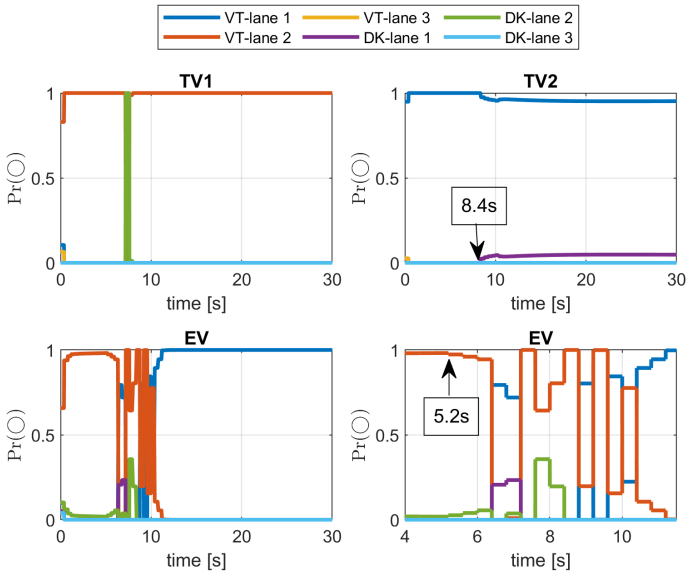


Fig. 4.6 Policy mode probabilities of all vehicles in Case 1. Policy mode $\bigcirc \in$ {VT-lane 1, VT-lane 2, VT-lane 3, DK-lane 1, DK-lane 2, DK-lane 3}

.

Fig. 4.7 shows the related motion trajectories of vehicles. The detailed velocity and acceleration profiles are represented in Fig. 4.8. Combined with Fig. 4.9, we can conclude that the EV still starts to change lanes at 5.6 s, and moves to the new lane at 8 s. Finally, Fig. 4.10 shows the policy mode probabilities of all vehicles during the simulation. The simulation results in this case are almost identical to the simulation results when TV2 aggressiveness is low. In addition, for TV2, after the EV changes lanes, it starts to decelerate at around 16.6 s, which is later than when it is more defensive. The result comparison shows that TV2, which has a lower priority than EV, does not affect the decision-making of the EV.

4.3.0.2 Case 2: the EV keeps the lane and brakes

The initial traffic scene of Case 2 is shown in Fig. 4.11. The motion states of vehicles are $x^{(\mathrm{TV1})} = \begin{bmatrix} 375 & 22 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV2})} = \begin{bmatrix} 330 & 20 & 0 & -12.13 & 0 & 0 \end{bmatrix}^\top$, and $x^{(\mathrm{EV})} = \begin{bmatrix} 350 & 18.9 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, respectively. The corresponding priority of the three vehicles is TV1, EV, TV2 in descending order. After 2 s, TV1 decelerates at $-2\ \mathrm{m/s^2}$ for 4 s and then accelerates back to the initial velocity of $22\ \mathrm{m/s}$.

Figs. 4.12, 4.13, 4.14, and 4.15 are simulation results of Case 2. From Fig. 4.15, we observe that for the EV, the cost of the lane-keeping mode is always lower than that of the lane-change mode, and it increases between 5 s and 8 s. This is because TV1, as the LV of the EV, drives slower than the EV due to the deceleration between 2 s and 6 s. The EV then reduces its speed to maintain the distance from TV1 until TV1 accelerates to a higher velocity than the EV. For the EV, lane change is not possible until around 16 s, as

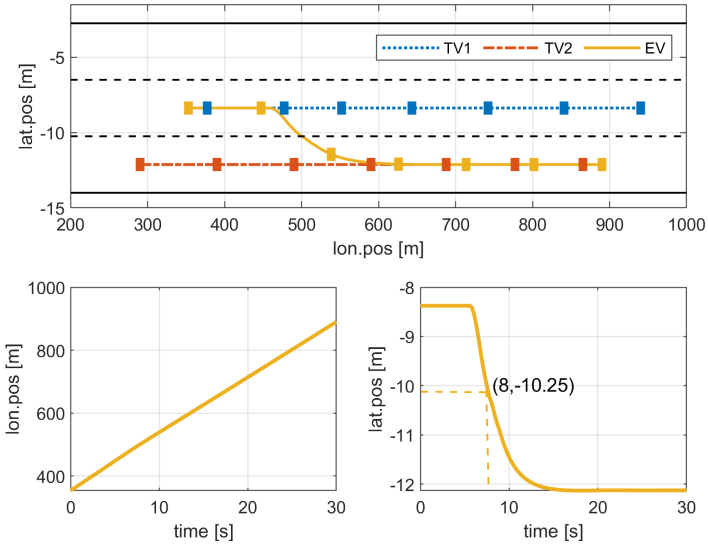


Fig. 4.7 Motion trajectories of vehicles and EV position when TV2 is aggressive

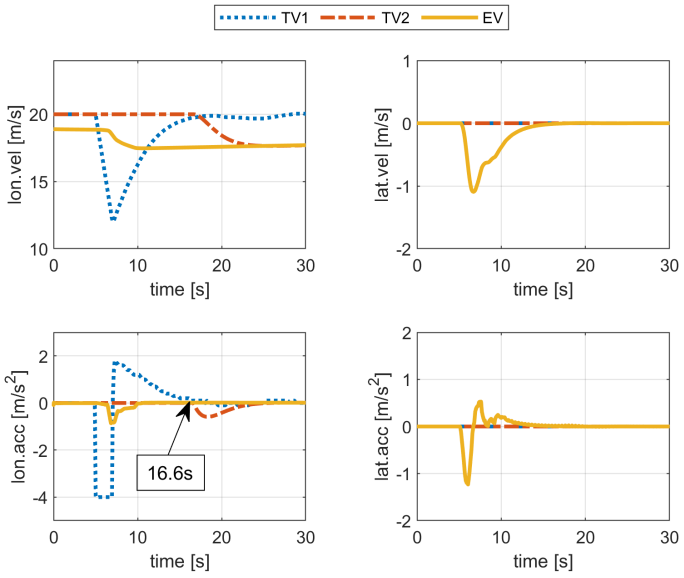


Fig. 4.8 Longitudinal and lateral velocity of all vehicles when TV2 is aggressive

TV2 drives behind the EV with a higher speed in lane 1. Then, lane change is feasible but at a higher cost than keeping the lane. Therefore, the EV maintains its velocity in the following simulation time. TV2 drives at the initial constant speed. The vehicles' associated motion trajectories and detailed velocities are shown in Figs. 4.12, 4.14. Fig. 4.13 provides some snapshots of traffic scenarios of Case 2 during the simulation.

The first three subplots of Fig. 4.16 show the policy mode probabilities of each vehicle throughout the simulation time. The fourth subplot is a zoomed-in view of the third subplot for the time interval 5 s to 7 s. For TV1, tracking the velocity in lane 2 is always the most likely mode with probability 1. Tracking the velocity in lane 1 is the most likely

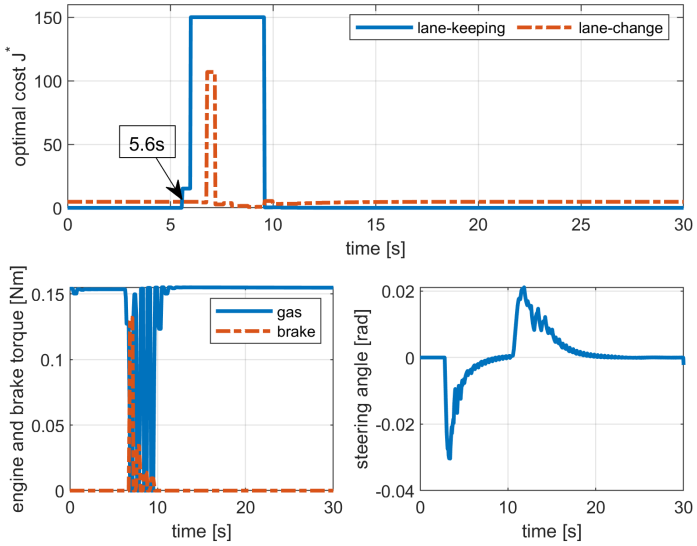


Fig. 4.9 Optimal cost and control actions of SCMPC when TV2 is aggressive

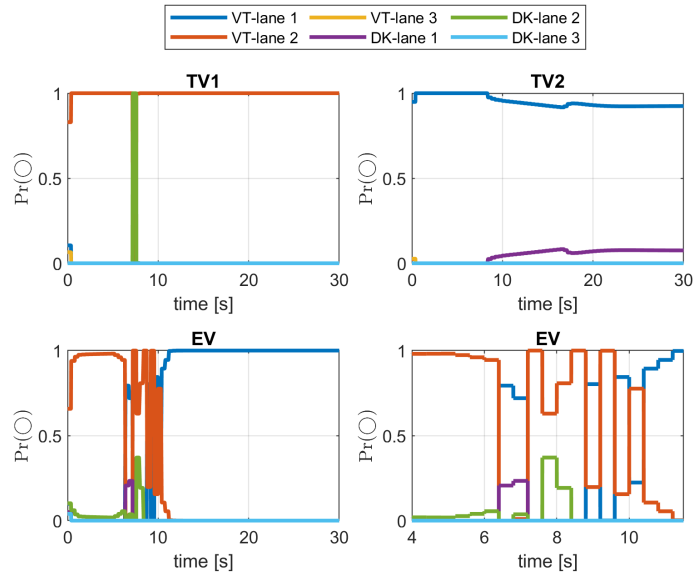


Fig. 4.10 Policy mode probabilities of all vehicles when TV2 is aggressive. Policy mode $\bigcirc \in \{$VT-lane 1, VT-lane 2, VT-lane 3, DK-lane 1, DK-lane 2, DK-lane 3$\}$

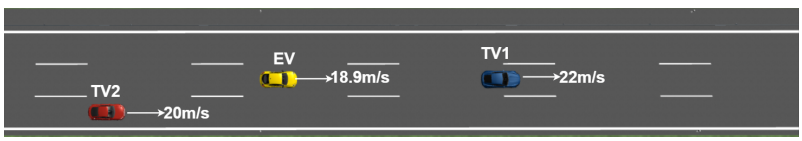


Fig. 4.11 Initial traffic scene of Case 2 used for SCMPC structure validation

mode with probability 1 for TV2. Staying in lane 2 and tracking its velocity is the most likely mode most of the time for the EV, while between 5.6 s and 6.4 s, the probability of keeping the distance mode is higher due to the deceleration of TV1.

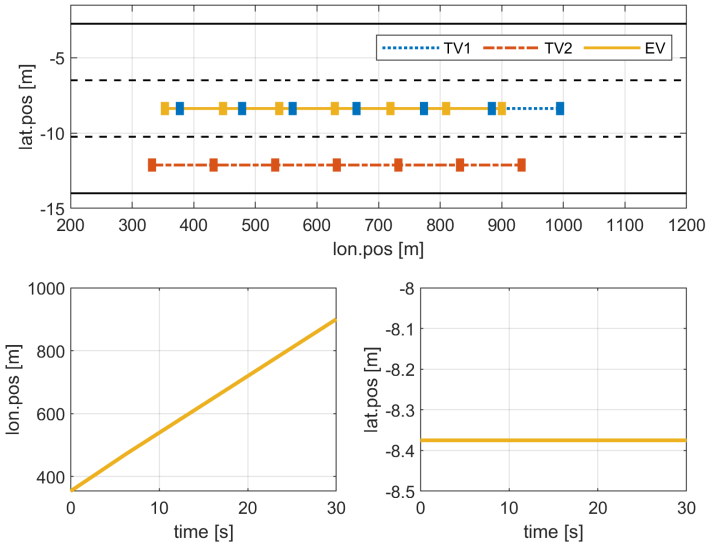


Fig. 4.12 Motion trajectories of vehicles and EV position in Case 2

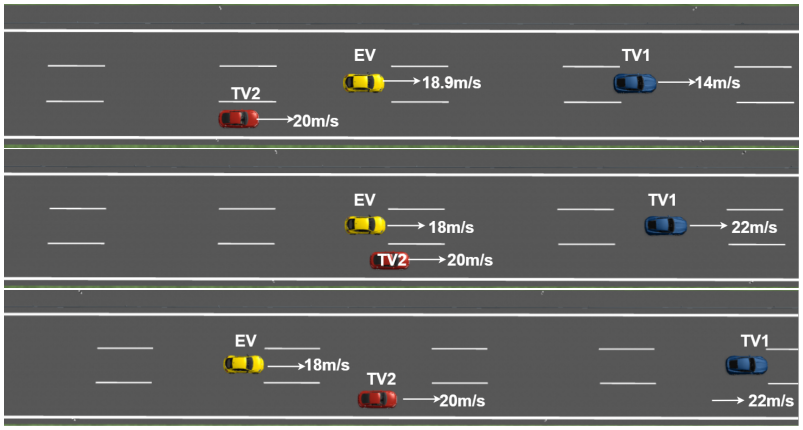


Fig. 4.13 Traffic scenes in Case 2 at 6 s, 14 s, and 20 s

4.3.0.3 Case 3: the EV changes lanes and keeps velocity

Fig. 4.17 shows the initial traffic scene of Case 3. Vehicles' motion states are $x^{(\mathrm{TV1})} = \begin{bmatrix} 375 & 17 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV2})} = \begin{bmatrix} 358 & 20 & 0 & -12.13 & 0 & 0 \end{bmatrix}^\top$, and $x^{(\mathrm{EV})} = \begin{bmatrix} 350 & 18.9 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$. Given the initial traffic situation, TV1 and TV2 move constantly in autonomous driving mode. The priority of the vehicles is TV2, TV1, EV in descending order. The EV responds to the motion maneuvers of TV1 and TV2.

Figs. 4.18, 4.19, 4.20, and 4.21 are simulation results of Case 3. From Fig. 4.21, we see that even though TV1 is slower than the EV, before 3.2 s, the cost of the SCMPC in lane-keeping mode is lower than that of SCMPC in lane-change mode. After that, as the relative distance between EV and TV1 decreases, the less costly mode switches to lane-change mode until 7.2 s. The EV changes to lane 1 at around 5.5 s (as seen from Fig. 4.18). Then, since TV1 is laterally close to the EV, the EV can only stay in the new

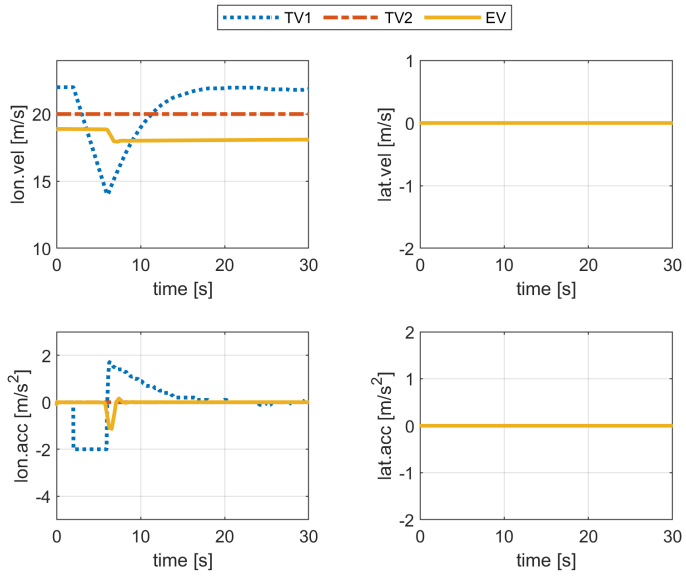


Fig. 4.14 Longitudinal and lateral velocity of all vehicles in Case 2

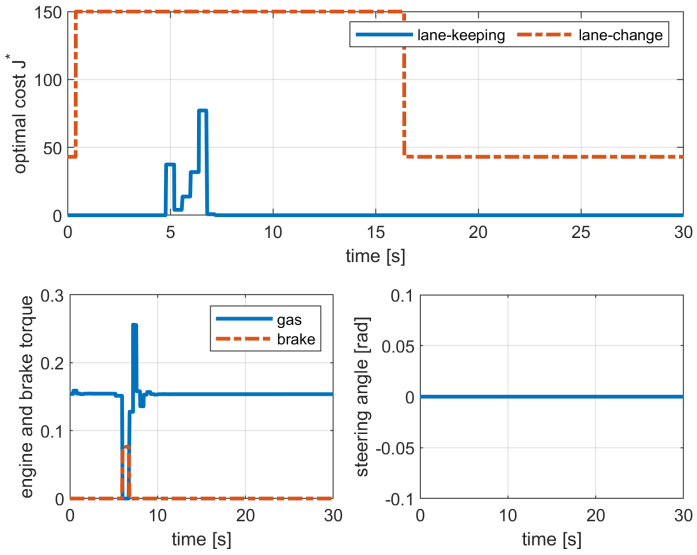


Fig. 4.15 Optimal cost and control actions of SCMPC in Case 2

lane at a lower cost. The EV's lane-keeping and lane-change motion behavior can be seen in Fig. 4.18. The EV maintains its speed throughout the simulation time.

Fig. 4.22 shows three vehicles' policy mode probabilities. Specifically, TV1 continually tracks the speed at $17\ \mathrm{m/s}$ and stays in lane 2 with a probability of around 1. TV2 maintains the initial speed and stays in lane 1 with probability 1. As for the EV, before $4\ \mathrm{s}$, it might perform VT or DK maneuvers in lane 2 with probability 0.93 and 0.07. After that, the most probable policy mode is time-varying between staying in lane 1 and lane 2 until $8\ \mathrm{s}$, as seen from the fourth subplot. Based on these two policy modes and the real measurements from IPG CarMaker, it is difficult for the IMM-KF to get an accurate mode

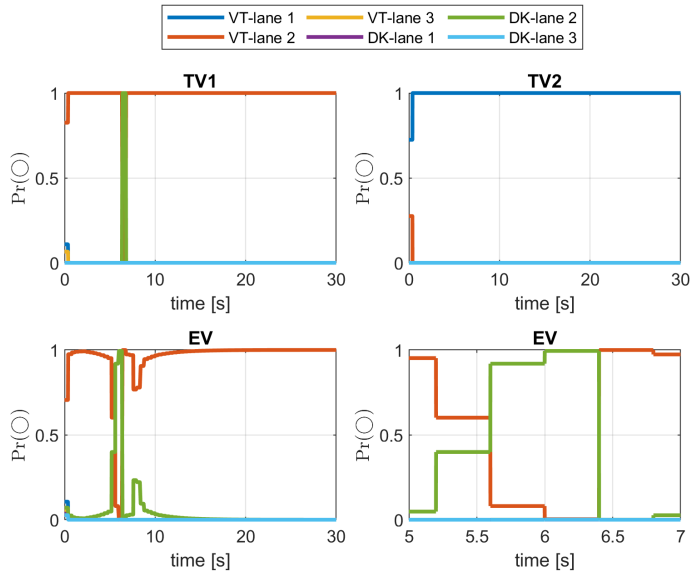


Fig. 4.16 Policy mode probabilities of all vehicles in Case 2. Policy mode $\bigcirc \in$ {VT-lane 1, VT-lane 2, VT-lane 3, DK-lane 1, DK-lane 2, DK-lane 3}

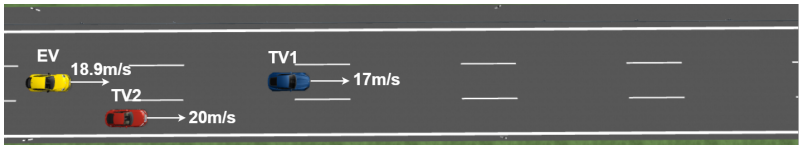


Fig. 4.17 Initial traffic scene of Case 3 used for SCMPC structure validation

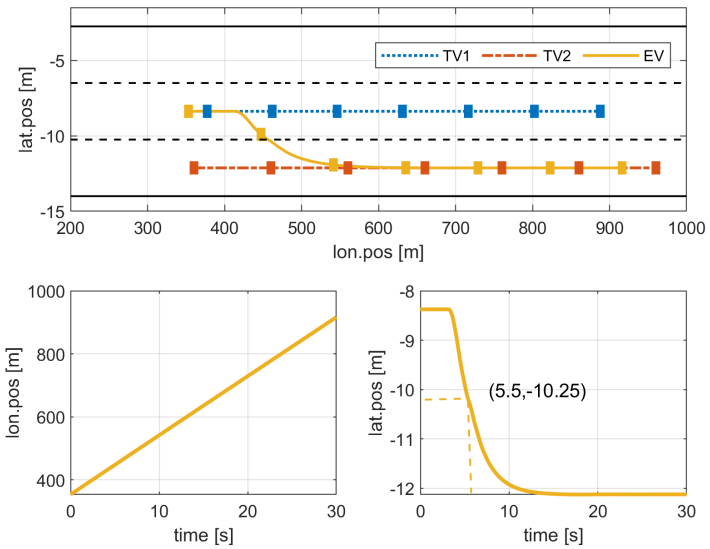


Fig. 4.18 Motion trajectories of all vehicles and EV position in Case 3

probability estimation of the EV's lane change maneuver. Finally, it is most probable that the EV keeps the velocity and stays in lane 1.

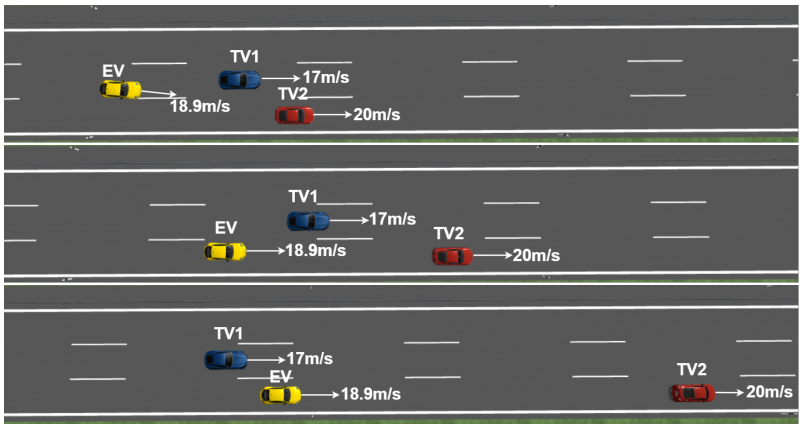


Fig. 4.19 Traffic scenes in Case 3 at 5 s, 11.5 s, and 26 s

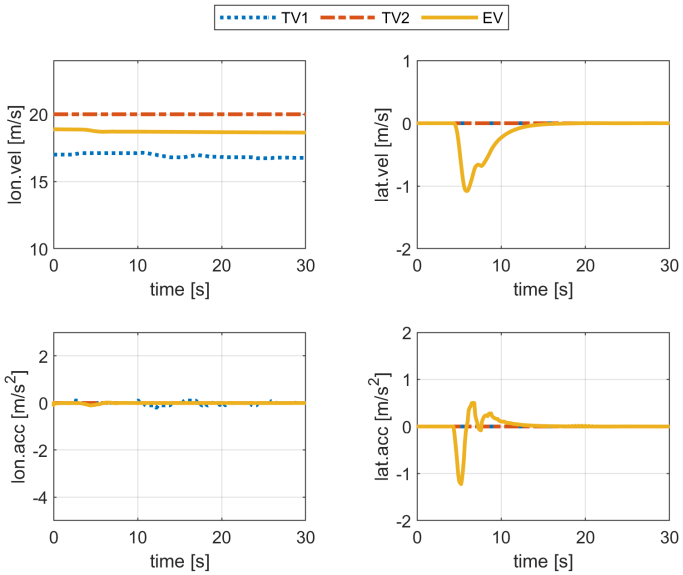


Fig. 4.20 Longitudinal and lateral velocity of all vehicles in Case 3

4.4 Summary

An SCMPC framework is presented for EVs on highways in this Chapter. After applying an IMM-KF to predict possible traffic evolutions, multiple interaction-aware scenarios are generated. In addition to the generated scenarios, a "worst-case" scenario is incorporated into the formulation of the safety constraints to ensure the safety of the EV. To address the uncertain lateral motion decisions of the EV, lane-keeping and lane-change control modes are designed within the SCMPC, where the control input with a lower cost function value is implemented. Moreover, the recursive feasibility of the optimization problem under the SCMPC framework is guaranteed by ensuring no collision between the EV and its LV over the minimal stopping horizon of the EV. The proposed algorithm is validated in three highway scenarios in IPG CarMaker. Simulation results demonstrate that the proposed SCMPC enables the EV to perform safe maneuvers.

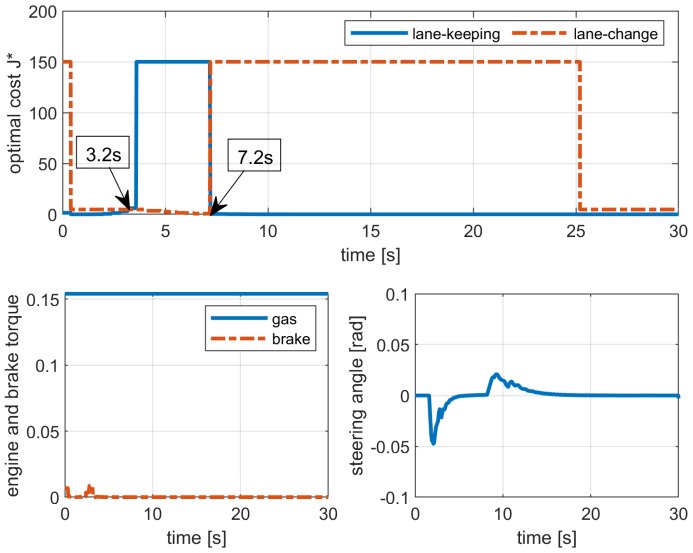


Fig. 4.21 Optimal cost and control actions of SCMPC in Case 3

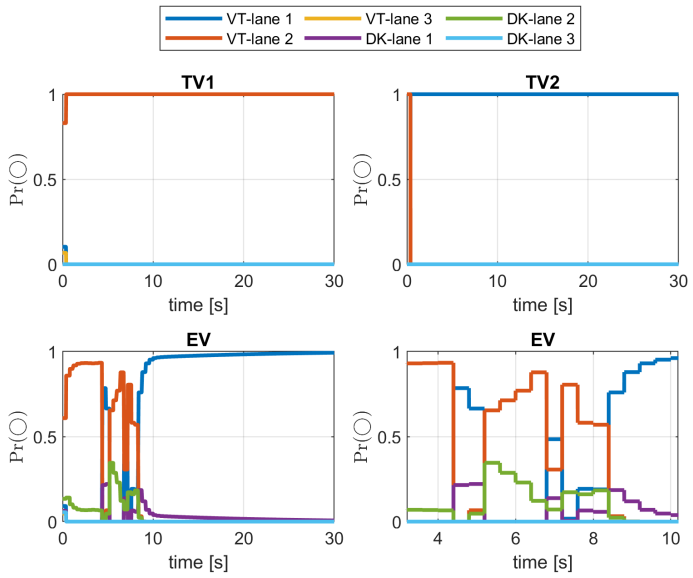


Fig. 4.22 Policy mode probabilities of all vehicles in Case 3. Policy mode $\bigcirc \in$ {VT-lane 1, VT-lane 2, VT-lane 3, DK-lane 1, DK-lane 2, DK-lane 3}

CHAPTER 5

CMPC for autonomous highway driving

5.1 Problem statement

Safety control for the EV is achieved by considering nominal scenarios together with a "worst-case" scenario in the previous chapter. In this setup, the EV is required to maintain different safe distance values from the LV for the two categories of scenarios, and the first-step control actions computed from both categories of scenarios are enforced to be identical. However, this design might lead to more conservative EV behavior compared with a controller that does not include a "worst-case" scenario, implying that other performance aspects may be compromised to guarantee safety.

Although a "worst-case" scenario is incorporated to ensure safety, we can nevertheless seek ways to mitigate any conservatism that introduces. One way to achieve this is that the EV could maintain its current motion state, such as the velocity, throughout the prediction horizon, and adjust it only when necessary. Because MPC applies only the first-step control input at each iteration, the resulting feedback mechanism provides the EV with additional flexibility to adjust its motion according to the actual scenario observed in the next step, thus reducing the potential conservatism. To this end, we propose a CMPC framework that introduces new terminal constraints based on this idea.

5.2 CMPC design and validation

We utilize the improved IMM-KF-based algorithm introduced in Section 3.2.5 to estimate and predict the motion behaviors of vehicles and generate interaction-aware scenarios. Based on the predicted scenarios of the TVs, the EV is controlled under a CMPC framework.

Remark 5.2.6. *In the previous chapter, the superscript for the EV was retained for clarity. In this chapter, to simplify the notation, the superscript for the EV is omitted unless otherwise specified.*

5.2.1 Differential flatness based kinematic unicycle vehicle model

We adopt a nonlinear kinematic unicycle model to represent vehicle dynamics. Its simplicity enables easier implementation of control algorithms [175]. Although it does not capture lateral slip dynamics, the model offers a good balance between computational efficiency and accuracy for highway scenarios with less abrupt lane changes and turns [176]. The continuous-time form of the nonlinear kinematic unicycle vehicle model is:

$$\underbrace{\begin{bmatrix} \dot{p}_{\text{lon}} \\ \dot{p}_{\text{lat}} \\ \dot{\theta} \end{bmatrix}}_{\dot{\chi}} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} v \\ \omega \end{bmatrix}}_{\xi}, \tag{5.1}$$

where $v_{\text{lon}} = \dot{p}_{\text{lon}}$, $v_{\text{lat}} = \dot{p}_{\text{lat}}$ are the longitudinal velocity and the lateral velocity, and the total velocity is $v = \sqrt{v_{\text{lon}}^2 + v_{\text{lat}}^2}$. θ is the vehicle orientation angle. ω denotes the associated orientation angular velocity here. The kinematic unicycle model of a vehicle front axle is shown in Fig. 5.1.

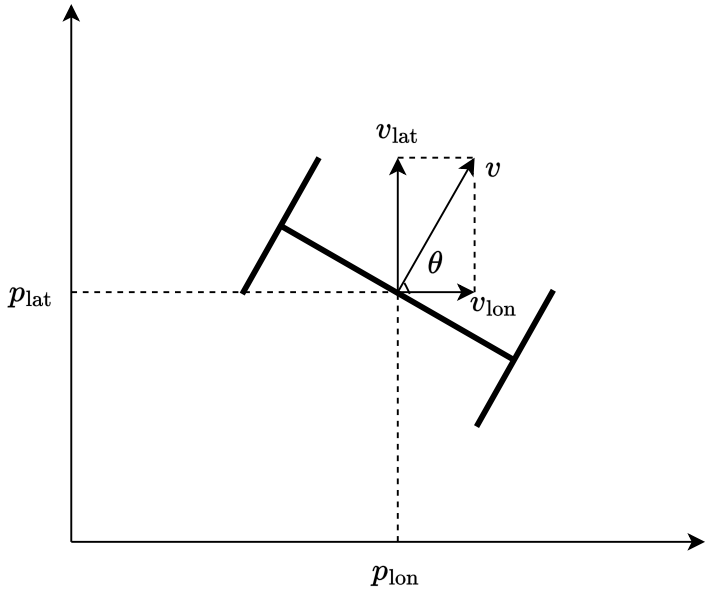


Fig. 5.1 Kinematic unicycle model of a vehicle front axle

Direct linearization of the nonlinear model (5.1) around an arbitrary equilibrium results in a non-controllable linear model, despite the nonlinear model being locally controllable. To address this, we apply a differential flatness method to derive an equivalent flat system, simplifying the problem compared to the original nonlinear model. We choose the flat output $\gamma = \begin{bmatrix} p_{\text{lon}} & p_{\text{lat}} \end{bmatrix}^\top$, the new system state $\chi = \begin{bmatrix} p_{\text{lon}} & p_{\text{lat}} & \dot{p}_{\text{lat}} \end{bmatrix}^\top$, and the new system input $u = \begin{bmatrix} u_1 & u_2 \end{bmatrix}^\top = \begin{bmatrix} \dot{p}_{\text{lon}} & \ddot{p}_{\text{lat}} \end{bmatrix}^\top$, where $\ddot{p}_{\text{lat}} = a_{\text{lat}}$ is the lateral acceleration.

The corresponding flatness system is written as

$$\dot{\chi} = \begin{bmatrix} \dot{p}_{\text{lon}} \\ \dot{p}_{\text{lat}} \\ \ddot{p}_{\text{lat}} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}}_{A} \begin{bmatrix} p_{\text{lon}} \\ p_{\text{lat}} \\ \dot{p}_{\text{lat}} \end{bmatrix} + \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}}_{B} \begin{bmatrix} \dot{p}_{\text{lon}} \\ \ddot{p}_{\text{lat}} \end{bmatrix}. \tag{5.2}$$

Then, we use the Euler method with the sampling time T to obtain the discrete-time form of the system equation (5.2) at time k.

$$\chi_{k+1} = \underbrace{(AT + \boldsymbol{I}_3)}_{\bar{A}} \chi_k + \underbrace{BT}_{\bar{B}} u_k. \tag{5.3}$$

The related output equation is

$$\gamma_k = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}}_{\bar{C}} \chi_k. \tag{5.4}$$

After obtaining the control input u_k, using the following equations to get the original control input ξ_k:

$$v_k = \frac{u_{1,k}}{\cos\theta_k}, \tag{5.5a}$$

$$\omega_k = \frac{u_{2,k}\cos{\theta_k}^2 - \dot{u}_{1,k}\sin\theta_k\cos\theta_k}{u_{1,k}}, \tag{5.5b}$$

where the original state θ_k and $\dot{u}_{1,k}$ are computed by

$$\theta_k = \arctan(\frac{v_{\text{lat},k}}{v_{\text{lon},k}}), \quad \dot{u}_{1,k} = \frac{u_{1,k} - u_{1,k-1}}{T}. \tag{5.6}$$

5.2.2 CMPC framework

Lane-keeping and lane-change are possible motion strategies for the EV, which correspond to two control modes. The final decision of the EV is made by choosing the mode with the minimal cost. For path planning, the related reference output $\gamma_{\text{ref},k} \in \mathbb{R}^2$ is generated to guide the EV to drive at a constant high longitudinal velocity, and to stay at the center of the target lane. In particular, the EV attempts to switch to the new lane in one step during the lane change. To ensure the safety of the EV, an extra "worst-case" scenario is included in the MPC design, where the LV is assumed to decelerate maximally to a standstill over the prediction horizon. The corresponding control sequences $\breve{u}_k$ act as contingency strategies and are subject to a proposed terminal constraint, as shown in (5.7g). A detailed recursive feasibility proof is discussed in Section 5.2.4.

The objective of the CFTOCP is to minimize the control effort of the EV to follow the designed reference trajectory as closely as possible while avoiding collision with surrounding TVs. $U_0 = \{u_0, ..., u_{N-1}\}$ and $\breve{U}_0 = \{\breve{u}_0, ..., \breve{u}_{N-1}\}$ are two sequences of decision variables. The first control sequences are used to evaluate the cost function with associated output γ_k, and ensure the EV's safety under the generated scenarios. The second control sequences guarantee the EV's safety in the "worst-case" scenario. To handle the possible "worst-case" scenario based on the applied control sequences u_k, the open-loop state χ_N is set to equal to the contingency one $\breve{\chi}_N$ at time step N. The CFTOCP is formulated as follows:

$$J^\star(\chi_0) = \min_{U_0, \breve{U}_0} \sum_{k=0}^{N} \|\gamma_k - \gamma_{\text{ref},k}\|^2_{W_Q} + \|u_k\|^2_{W_R}, \tag{5.7a}$$

$$s.t.\ \ \chi_{k+1} = \bar{A}\chi_k + \bar{B}u_k, \ \ k = 0, 1, ..., N-1, \tag{5.7b}$$

$$\gamma_k = \bar{C}\chi_k, \ \ k = 0, 1, ..., N, \tag{5.7c}$$

$$\breve{\chi}_{k+1} = \bar{A}\breve{\chi}_k + \bar{B}\breve{u}_k, \ \ k = 0, 1, ..., N-1, \tag{5.7d}$$

$$\chi_k \in \mathbb{X}, \quad \breve{\chi}_k \in \breve{\mathbb{X}}, \ \ k = 0, 1, ..., N, \tag{5.7e}$$

$$u_k \in \mathbb{U}, \quad \breve{u}_k \in \mathbb{U}, \ \ k = 0, 1, ..., N, \tag{5.7f}$$

$$\chi_N = \breve{\chi}_N, \ \ \breve{\chi}_N \in \breve{\mathbb{X}}_f, \tag{5.7g}$$

$$\chi_0 = \breve{\chi}_0 = \chi(0). \tag{5.7h}$$

Here $W_Q \in \mathbb{R}^{2\times2}$ and $W_R \in \mathbb{R}^{2\times2}$ are positive definite weighting matrices for tuning. The admissible state sets $\mathbb{X}$, $\breve{\mathbb{X}}$, and $\breve{\mathbb{X}}_f$, and input set $\mathbb{U}$ are restrained by appropriate constraints, as detailed below. Note that when a lane change is possible, two CFTOCPs are solved corresponding to lane-keeping and lane-change control modes. The EV executes according to the command from the less costly mode.

5.2.3 Constraint formulation

The states and inputs of the EV are constrained by lane bounds, traffic rules, and driving comfort as follows:

$$0 \le v_{\text{lon},k}, \quad l_{\text{lb}} \le p_{\text{lat},k} \le l_{\text{ub}}, \tag{5.8a}$$

$$\underline{a}_{\text{lon}} \le a_{\text{lon},k} \le \overline{a}_{\text{lon}}, \quad \underline{a}_{\text{lat}} \le a_{\text{lat},k} \le \overline{a}_{\text{lat}}, \tag{5.8b}$$

where $[l_{\text{ub}}, l_{\text{lb}}]$ are the upper and lower bounds of the lane. $\underline{\bullet}$ and $\overline{\bullet}$ denote the minimum and maximum values of the associated variables.

A safety distance $\underline{d}$ between the EV and LV needs to be maintained:

$$d_k \ge \underline{d}, \quad k = 0, 1, ..., N, \tag{5.9}$$

where $\underline{d}$ is computed by

$$\underline{d} = \tau v_{\text{lon},k} + \triangle d, \tag{5.10}$$

with the design parameters τ and $\triangle d$. If the reference point of all vehicles is in their respective center, for example, choose $\triangle d \geq \frac{l+l^{(\text{LV})}}{2}$, where l and $l^{(\text{LV})}$ are the length of the EV and LV. During the lane change, the EV needs to maintain a safety distance from the LVs in the current and the target lane, and also from the successor TV in the target lane. The safety constraint under the generated scenario is based on (5.10), while for the "worst-case" scenario, the safety distance needs to satisfy $\underline{d} = \triangle d$.

5.2.4 Recursive feasibility analysis for CMPC

The recursive feasibility of the proposed control method is guaranteed by introducing a terminal constraint. The definition of recursive feasibility and the related assumptions are provided below.

Definition 5.2.4. *(**Recursive Feasibility**). The CMPC is **recursively feasible** if and only if for all initially feasible states, and for all optimal sequences of control inputs the MPC optimization problem remains feasible for all time [177]. Namely, the safety constraint* (5.9) *always holds.*

Assumption 3. All vehicles are assumed to drive forward, and the EV is only responsible for avoiding front collisions.

Assumption 4. The EV and its LV have identical deceleration capabilities. In other words, the minimum acceleration values of the EV and the LV are the same.

The minimal stopping horizon of the LV determines the prediction horizon at each time step, and is given by

$$N = \lceil \frac{v_{\text{lon},0}^{(\text{LV})}}{|\underline{a}_{\text{lon}}^{(\text{LV})}|T} \rceil, \tag{5.11}$$

where $\lceil \bullet \rceil$ represents the smallest integer greater than or equal to $\bullet$. $v_{\text{lon},0}^{(\text{LV})}$ represents the initial velocity of the LV at the current time step. For notation simplicity, it is abbreviated from $v_{\text{lon},k+0|k}^{(\text{LV})}$when the current time step is k. $|\underline{a}_{\text{lon}}^{(\text{LV})}|$ represents the absolute value of the minimum acceleration of the LV. Denote the initial position of the LV at the current time is $p_{\text{lon},0}^{(\text{LV})}$. The LV's stopping position at time N, $p_{\text{lon},N}^{(\text{LV})}$, is then computed by:

$$p_{\text{lon},N}^{(\text{LV})} = p_{\text{lon},0}^{(\text{LV})} + v_{\text{lon},0}^{(\text{LV})} NT + \frac{1}{2}\underline{a}_{\text{lon}}(NT)^2. \tag{5.12}$$

To prevent the EV's reaction from being overly conservative in the "worst-case" scenario, the EV is considered to stop later than the LV. The relative distance d_N between the

EV and LV at the final prediction step must satisfy the following condition to ensure the EV's safety when the LV stops:

$$d_N \geq d_s, \;\; d_s = \breve{v}_{\text{lon},N}\Delta NT + \frac{1}{2}\underline{a}_{\text{lon}}(\Delta NT)^2 + \Delta d, \tag{5.13}$$

where d_s represents the minimal distance that the EV travels from time N to a standstill. $\breve{v}_{\text{lon},N}$ represents the EV's contingency velocity at time N, and ΔN is the minimal extra time for the EV to stop from time N, which is calculated as:

$$\Delta N = \frac{\breve{v}_{\text{lon},N}}{|\underline{a}_{\text{lon}}|T}. \tag{5.14}$$

Consequently, the terminal set can be expressed as:

$$\breve{\mathbb{X}}_f = \{\breve{\chi}_N |\; \breve{p}_{\text{lon},N} \leq p^{(\text{LV})}_{\text{lon},N} - d_s\}. \tag{5.15}$$

Theorem 5.2.2. *(**Recursive Feasibility**). The optimization problem* (5.7) *is **recursively feasible** based on Assumptions 3, 4, and the prediction horizon N in* (5.11)*.*

Denote the control sequences and state sequences by considering the generated normal scenarios at time k as $\{u_{k|k}, u_{k+1|k}, ..., u_{N_k-1|k}\}$ and $\{\chi_{k+1|k}, \chi_{k+2|k}, ..., \chi_{N_k|k}\}$, the control sequences and state sequences by considering the "worst-case" scenario as $\{\breve{u}_{k|k}$, $\breve{u}_{k+1|k}, ..., \breve{u}_{N_k-1|k}\}$ and $\{\breve{\chi}_{k+1|k}, \breve{\chi}_{k+2|k}, ..., \breve{\chi}_{N_k|k}\}$, and the optimal control sequences and state sequences as $\{\breve{u}^\star_{k|k}, \breve{u}^\star_{k+1|k}, ..., \breve{u}^\star_{N_k-1|k}\}$ and $\{\breve{\chi}^\star_{k+1|k}, \breve{\chi}^\star_{k+2|k}, ..., \breve{\chi}^\star_{N_k|k}\}$.

Proof. Let two control sequences by considering the generated normal scenarios and the "worst-case" scenario at time 0 be $\{u_{0|0}, u_{1|0}, ..., u_{N_0-1|0}\}$, and $\{\breve{u}_{0|0}, \breve{u}_{1|0}, ..., \breve{u}_{N_0-1|0}\}$, respectively. Choose the second control sequence as the initially feasible solution $\{\breve{u}^\star_{0|0}$, $\breve{u}^\star_{1|0}, ..., \breve{u}^\star_{N_0-1|0}\}$. Its related state sequence is $\{\breve{\chi}^\star_{1|0}, \breve{\chi}^\star_{2|0}, ..., \breve{\chi}^\star_{N_0|0}\}$. The terminal state $\breve{\chi}^\star_{N_0|0}$ is determined according to (5.15). We apply $\breve{u}^\star_{0|0}$ to the system (5.3), and obtain

$$\chi_1 = \bar{A}\chi_0 + \bar{B}\breve{u}^\star_{0|0} = \breve{\chi}^\star_{1|0}.$$

In the next time step, the prediction horizon N_1 is determined by the current real velocity of the LV $v^{(\text{LV})}_{\text{lon},1+0|1}$according to (5.11). The worst case is that the LV does decelerate maximally at time 0. Then $N_1 < N_0$ because the LV's velocity is decreased. Due to $N_1 < N_0$, the optimal input sequence at time 1 is shorter than that of time 0. The following is a feasible solution for the MPC initialized at χ_1:

$$\{u_{1|1}, u_{2|1}, ..., u_{N_1-1|1}\} = \{\breve{u}^\star_{1|0}, \breve{u}^\star_{2|0}, ..., \breve{u}^\star_{N_1-1|0}\},$$

The related state sequence is

$$\{\chi_{2|1}, \chi_{3|1}, ..., \chi_{N_1|1}\} = \{\breve{\chi}^{\star}_{2|0}, \breve{\chi}^{\star}_{3|0}, ..., \breve{\chi}^{\star}_{N_1|0}\},$$

where $\breve{\chi}^{\star}_{N_1|0} \in \breve{\mathbb{X}}_f$. Both sequences are feasible for the MPC problem because they satisfy the dynamics and the constraints. □

Remark 5.2.7. *The prediction horizon N and the additional time ΔN required for the EV to stop are time-varying due to the time-dependent variables $v^{(LV)}_{lon,0}$ and $\breve{v}_{lon,N}$.*

The CFTOCP is a second-order cone programming problem based on the terminal constraint, which is still a convex problem yielding a unique and global solution [178].

5.2.5 Discussion

The main advantages of the proposed CMPC over the SCMPC in Chapter 4 are summarized as follows:

i) The proposed terminal set in (5.15) is larger, resulting in an expanded feasible set. In SCMPC, the terminal set is defined based on the EV employing maximum deceleration as a contingency strategy, with the corresponding terminal position being the EV's stopping position.

ii) The proposed CMPC approach reduces the number of decision variables. Based on the proposed CMPC, the prediction horizon N is computed based on the minimal stopping horizon of the LV. Consequently, the size of the decision variable for the CFTOCP is $2N$. However, in SCMPC, the prediction horizon is determined by the minimal stopping horizon of the EV, which is $N + \Delta N$. As a result, the decision variable size becomes $2(N + \Delta N)$ when identical prediction horizons are used to compute two control sequences.

iii) The proposed contingency scheme enables the EV to operate safely and less conservatively under the "worst-case" scenario compared to the method in SCMPC, which is further elaborated through various scenarios in the subsequent section.

5.3 Simulation and discussion

The effectiveness of the proposed control structure is validated in a high-fidelity IPG CarMaker and Simulink co-simulation environment. To properly implement the calculated control action in IPG CarMaker, the longitudinal control input is converted to gas or brake torque, while the lateral control inputs are converted into steering angle, as described in Section 4.3. Moreover, the conservatism of the proposed control approach is evaluated

and compared with other methods in three different cases. Table 5.1 presents the parameters used in the studied cases. The total simulation time is represented by t_{s}. The length and width of TV1, TV2, and the EV are specified in sets l and b, respectively. It is noted that, in the simulation, changing to lane 3 is not considered in the lane-change mode. The problem is solved using Gurobi [179].

Table 5.1 Design parameters used in the CMPC simulation study

$t_{\mathrm{s}}(\mathrm{s})$	$T(\mathrm{s})$	$\overline{a}_{\mathrm{lon}}^{(\mathrm{LV})}(\mathrm{m/s^2})$	$\underline{a}_{\mathrm{lon}}^{(\mathrm{LV})}(\mathrm{m/s^2})$
30.00	0.7	3.00	-3.00
$\tau(\mathrm{s})$	$\underline{P}$	$\overline{a}_{\mathrm{lon}}^{(\mathrm{EV})}(\mathrm{m/s^2})$	$\underline{a}_{\mathrm{lon}}^{(\mathrm{EV})}(\mathrm{m/s^2})$
0.40	0.075	3.00	-3.00
$l_{\mathrm{ub}}(\mathrm{m})$	$l_{\mathrm{lb}}(\mathrm{m})$	$\overline{a}_{\mathrm{lat}}^{(\mathrm{EV})}(\mathrm{m/s^2})$	$\underline{a}_{\mathrm{lat}}^{(\mathrm{EV})}(\mathrm{m/s^2})$
33.80	21.00	3.00	-3.00
Q	R	$l(\mathrm{m})$	$b(\mathrm{m})$
$\boldsymbol{I}_7$	$\boldsymbol{I}_7$	$\{5.96, 4.35, 4.85\}$	$\{2.32, 2.50, 2.02\}$
W_Q	W_R	K_{lat}	$p_c^{(1),(2),(3)}(\mathrm{m})$
$10\boldsymbol{I}_2$	$10\boldsymbol{I}_2$	$[1.15, 3.39, 3.58]$	$\{22.98, 26.88, 31.3\}$

5.3.1 IPG CarMaker evaluation

The initial traffic scenario under IPG CarMaker evaluation is illustrated in Fig. 5.2. The corresponding states of the three vehicles are: $x^{(\mathrm{TV1})} = \begin{bmatrix} 445 & 28 & 0 & 22.98 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV2})} = \begin{bmatrix} 395 & 32 & 0 & 22.98 & 0 & 0 \end{bmatrix}^\top$, and $x^{(\mathrm{EV})} = \begin{bmatrix} 350 & 35 & 0 & 26.94 & 0 & 0 \end{bmatrix}^\top$, respectively. A longer simulation time of 50 s is considered for IPG CarMaker implementation to observe more diverse control performances of the EV. Based on the initial traffic setup, the three vehicles interact with one another, and the EV is controlled by the designed control structure. According to the initial traffic configuration, the priority of the vehicles is $[\mathrm{TV1}, \mathrm{TV2}, \mathrm{EV}]$ in descending order. Consequently, TV2 reacts specifically to TV1, and the EV reacts to both TV1 and TV2.

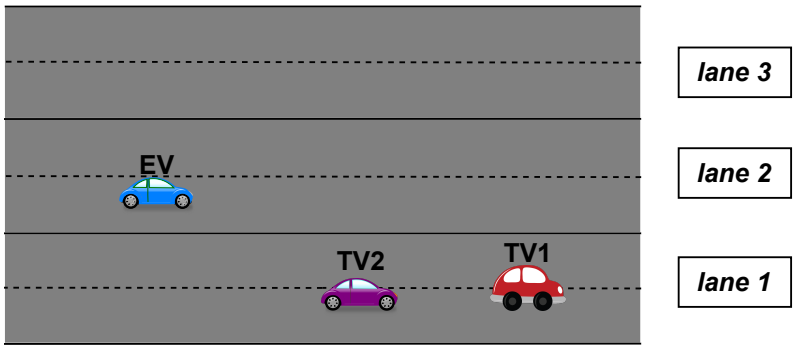


Fig. 5.2 Initial traffic scene used in IPG CarMaker evaluation

The simulation results are illustrated in Figs. 5.3 and 5.4, which show the trajectories, and the position and velocity profiles of the three vehicles, respectively. As observed in subplot (1) of Fig. 5.4, due to interaction with TV2 and TV1, the EV slightly decelerates starting from 2.5 s, as it predicts TV2's intention to lane changes, and decides to cooperate and leave enough space for TV2. At 3 s, TV2 initiates a left lane change, as shown in subplot (3) of Fig. 5.4. This happens because TV2 has a higher velocity than TV1. By interacting with both TV1 and the EV, TV2 changes lanes ahead of the EV instead of decelerating to maintain a safe distance from TV1. In response, the EV continues to decelerate. After TV2 moves to lane 1 at 4.4 s, the gap between TV2 and the EV decreases, prompting the EV to switch to lane 1 for a higher speed at 5 s, reaching the new lane at 7.8 s. TV1 accelerates from 28 m/s to 30 m/s at 8 s. However, about 5 s after the EV's lane change, based on the interaction with TV1 and TV2, given that TV2 has moved farther ahead of the EV than TV1 due to its higher velocity, the EV decides to return to lane 2 at 13 s (as observed in subplot (3) of Fig. 5.4), and reaches the goal at 15.8 s (as shown in subplot (2) of Fig. 5.4). Afterward, the EV continues behind TV2, maintaining nearly the same velocity. As observed in subplot (3) of Fig. 5.4, around 38 s, the EV initiates a right lane change again to pursue a higher velocity. At this time, both TV2 and the EV run ahead of TV1, and there is sufficient space for the EV to safely move in front of TV1. At 41.8 s (as shown in subplot (2) of Fig. 5.4), the EV reaches lane 1 and accelerates to 35 m/s. These simulation results demonstrate the effectiveness of the proposed control architecture in enabling the EV to perform safe maneuvers.

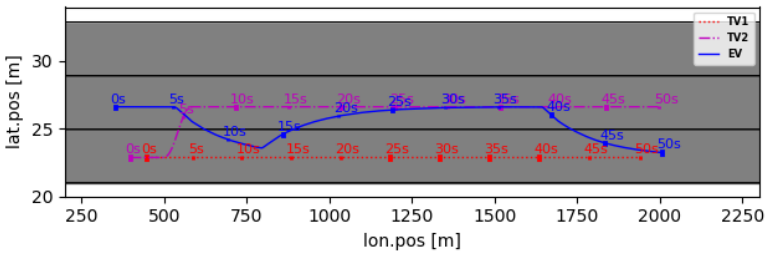


Fig. 5.3 Motion trajectories of vehicles (shown at each 5 s)

5.3.2 Evaluation of conservatism

The inclusion of the "worst-case" scenario may raise concerns about the conservatism of the control actions. In this section, we first investigate the EV's performance when the "worst-case" scenario does occur in three different traffic scenes. The objective is to demonstrate that even under these challenging cases, the EV controlled by the proposed control approach operates with minimal conservatism. Undoubtedly, in normal situations where the "worst-case" scenario does not occur, the EV is expected to behave in a less conservative manner as well. This conclusion will be further validated in the

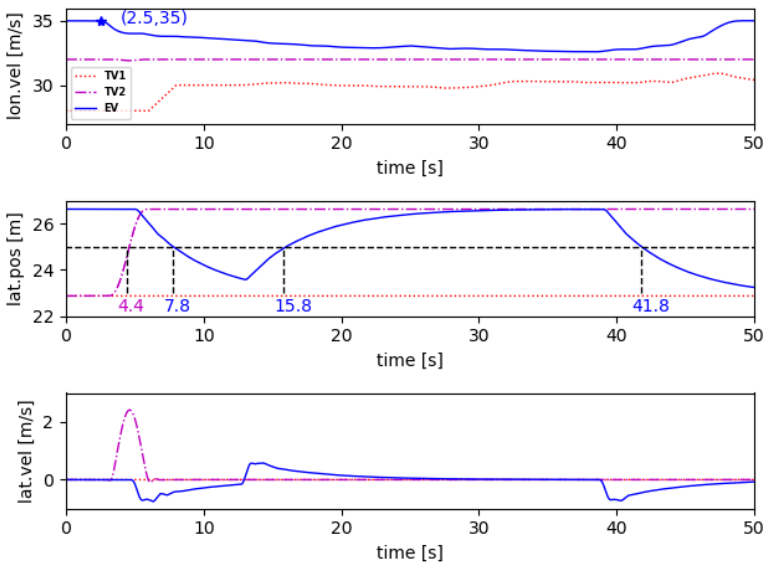


Fig. 5.4 Lateral position and velocity profiles of vehicles

cases studied. Furthermore, the conservatism of the proposed method is compared to the approach presented in Chapter 4, as well as another method that does not account for the "worst-case" scenario. These methods are referred to as CMPC, SCMPC, and NORMAL-MPC, respectively, in the simulation study. The initial traffic scenes of three studied cases are illustrated in Fig. 5.5, which are constructed by progressively worsening the initial condition of the EV. The analysis is based on open-loop simulation results of the initial simulation time.

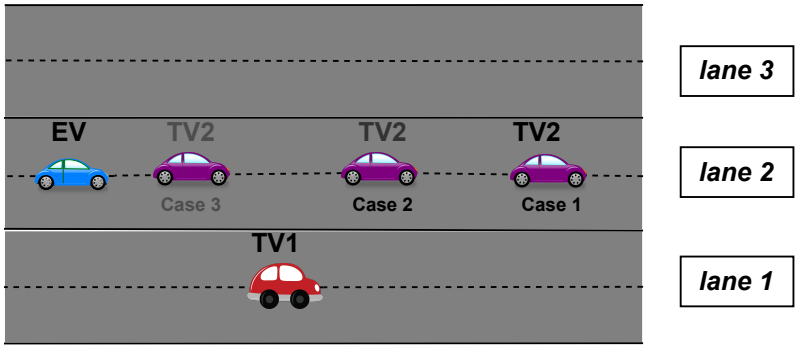


Fig. 5.5 Initial traffic scenes of studied cases showing TV2's initial condition changes

5.3.2.1 Case 1

The initial conditions for three vehicles are $x^{(\mathrm{TV1})} = \begin{bmatrix} 127.87 & 32.48 & 0 & 22.52 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV2})} = \begin{bmatrix} 188.19 & 24 & -3 & 26.42 & 0 & 0 \end{bmatrix}^\top$, and $x^{(\mathrm{EV})} = \begin{bmatrix} 3 & 35 & 0 & 25.65 & 0 & 0 \end{bmatrix}^\top$, respectively. TV2 is ahead of the EV and decelerates at $-3\ \mathrm{m/s^2}$ until it comes to a stop. The "worst-case" scenario does occur for the EV.

To evaluate whether incorporating the "worst-case" scenario leads to conservative control actions, we analyze the safe control sequences derived from CMPC and SCMPC. At the initial time, the corresponding open-loop safe sequences based on the CMPC (plotted in blue solid lines), the open-loop safe sequences based on the SCMPC (plotted in yellow dashed lines with markers), are depicted in Fig. 5.6, as well as the open-loop normal sequences based on the NORMAL-MPC (plotted in green dotted lines). The prediction horizons used in CMPC, SCMPC, and NORMAL-MPC are 8 s, 12 s, and 12 s, respectively. The first two prediction horizons are determined based on the minimal stopping time of the TV2 and EV. The prediction horizon for NORMAL-MPC is chosen to match that of SCMPC. Two subplots present the longitudinal position profile and the velocity step profile of the EV, as well as the longitudinal position profile of TV2 (plotted in a purple dot-dashed line). In response to the "worst-case" scenario, the EV behaves differently under the control of two safety-guaranteed MPCs. Based on the CMPC, the EV maintains the velocity for 2 s before gradually decelerating. In contrast, based on the SCMPC, the EV begins to decelerate at 0.5 s. Under the NORMAL-MPC, however, the EV maintains the velocity for 8 s, but due to no safety guarantee, it then collides with the TV2 at 8 s, as highlighted by a red marker in the figure. From the analysis of the control actions calculated from the three MPCs, we observe that if the "worst-case" scenario happens, the EV operates less conservatively under the proposed CMPC compared to SCMPC, particularly in terms of the braking start time. The way the EV behaves based on the CMPC can be regarded as the least conservative possible manner that still ensures safety.

For the first step of the initial time, the results for CMPC, SCMPC and NORMAL-MPC are almost identical. This indicates that incorporating the "worst-case" scenario does not make the control action conservative at this step. In the subsequent steps, the behavior of TV2 affects the response of the EV. If lane changes are not considered for the EV, the worst scenario assumes that TV2 continues to decelerate at its maximum rate. This situation has been accounted for in the design of the CMPC and SCMPC. Consequently, the EV will operate as predicted by the MPCs under the "worst-case" scenario from the first step onward. Then it is concluded that the EV performs better under CMPC compared to the other two MPCs while maintaining a safety guarantee. The proposed control structure, however, goes beyond lane-keeping by including a lane-change control mode. This additional mode provides the EV with more options when a contingent event occurs in the current lane. To demonstrate this, the closed-loop simulation results based on three MPCs are presented in Figs. 5.7 and 5.8 for the "worst-case" scenario. The results show that, due to the lane change option, the EV changes lanes instead of decelerating under all three MPCs. Moreover, the EV remains in its lane longer under CMPC, primarily because the shorter prediction horizon used in the lane-keeping mode results in a lower associated cost compared to the lane-change mode. The motion distances covered by the EV under three MPCs are nearly identical.

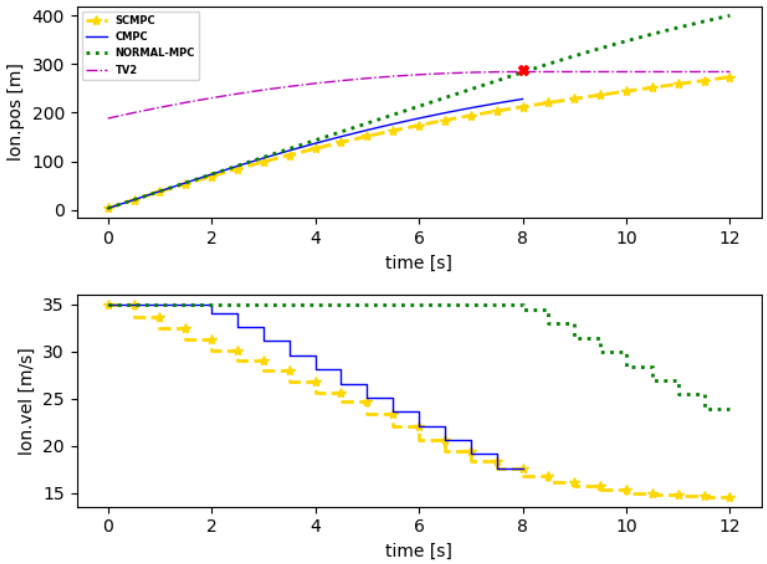


Fig. 5.6 Open-loop simulation results of the initial simulation time under the "worst-case" scenario in Case 1

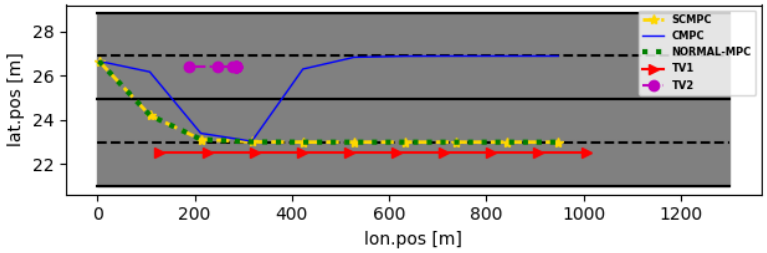


Fig. 5.7 Motion trajectories of vehicles (shown at each 3 s) under the "worst-case" scenario in Case 1

Next, we evaluate the EV's performance under different control methods when TV2 maintains the initial velocity. Figs. 5.9 and 5.10 illustrate the motion trajectories of vehicles, and the longitudinal position, velocity, and lateral position of the EV over the simulation time for the three MPCs. Specifically, under the CMPC, the EV maintains its velocity and lane for the first 5 s, then decelerates for another 4 s due to the reduced relative distance with TV2. Subsequently, the EV switches lanes to minimize cost. After remaining in the new lane for 9 s, the EV moves longitudinally farther ahead of TV2 while achieving a higher velocity. It then drives back to the original lane due to the lower cost of lane-change mode. Under the SCMPC, the EV travels at a constant velocity in the current lane for 7 s before switching lanes. It remains in the new lane for the rest of the simulation time. Similarly, under the NORMAL-MPC, the EV behaves almost identically to the behavior under the SCMPC. By analyzing the EV motion states, it is observed that the EV covers almost the same longitudinal distance under all MPCs.

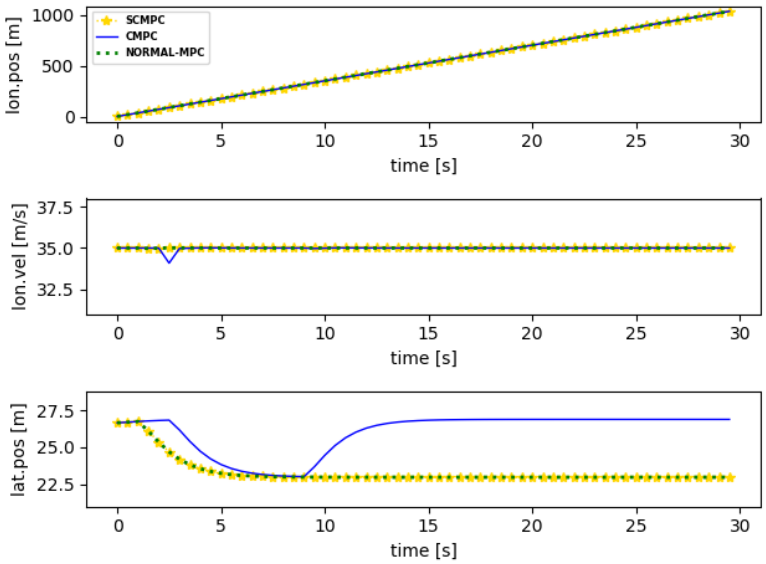


Fig. 5.8 EV's longitudinal position and velocity, and lateral position under the "worst-case" scenario in Case 1

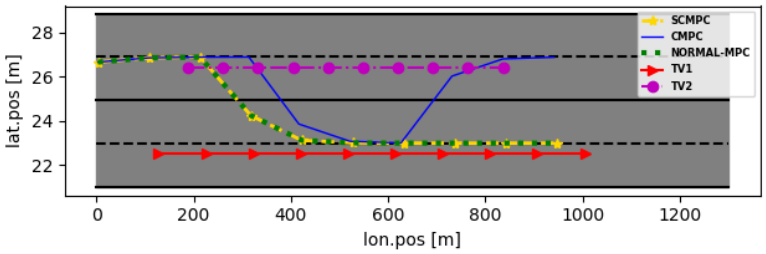


Fig. 5.9 Motion trajectories of vehicles (shown at each 3 s) when TV2 keeps its velocity

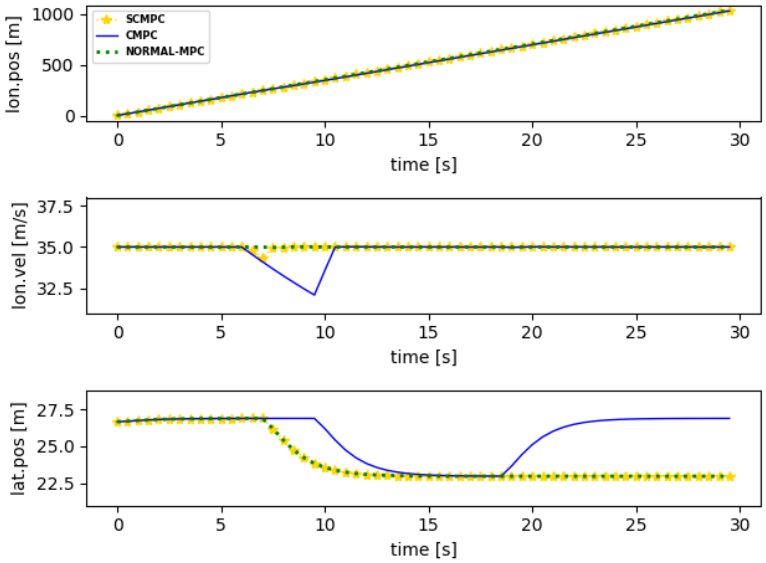


Fig. 5.10 EV's longitudinal position and velocity, and lateral position when TV2 keeps its velocity

5.3.2.2 Case 2

In this case, the initial relative distance between the EV and TV2 is shorter than the Case 1. The initial conditions for TV2 is $x^{(\mathrm{TV2})} = \begin{bmatrix} 158.19 & 24 & -3 & 26.42 & 0 & 0 \end{bmatrix}^\top$. The remaining initial conditions are the same as in Case 1. The open-loop safe sequences for the initial simulation time based on the CMPC and SCMPC are illustrated in Fig. 5.11, along with the open-loop normal sequences based on the NORMAL-MPC. The second subplot shows that the first-step longitudinal velocities of the EV are identical across all three approaches at this time. In response to the "worst-case" scenario, based on the CMPC, EV maintains its velocity for a longer time compared to SCMPC. In contrast, under NORMAL-MPC, the EV collides with the TV2 at 7 s, as highlighted by a red marker in the first subplot. From an overall control perspective, the closed-loop simulation results for the three MPCs are presented in Figs. 5.12 and 5.13. Specifically, when the "worst-case" scenario occurs, under CMPC, the EV stays in its lane for 1.5 s before gradually decreasing speed. It then switches lanes and recovers the original velocity. At 9 s, with TV2 far behind, the EV returns to the original lane as this becomes the less costly option. Under SCMPC and NORMAL-MPC, the EV behaves similarly. It changes lane after 0.5 s and remains in the new lane for the remainder of the simulation. Despite these differences, the EV covers almost the same longitudinal distance under all three MPCs.

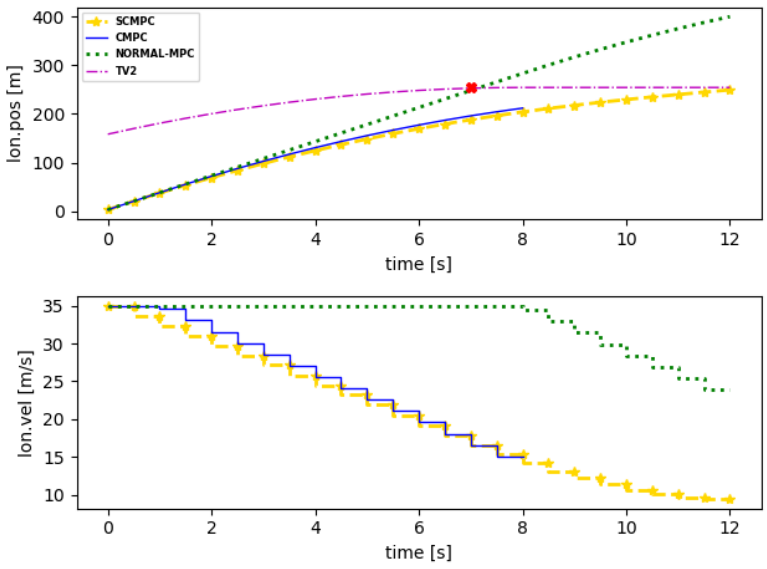


Fig. 5.11 Open-loop simulation results of the initial simulation time under the "worst-case" scenario in Case 2

Then, we investigate the EV's performance under different control methods when TV2 accelerates. TV2 accelerates at $3\ \mathrm{m/s^2}$ until reaching the maximum admissible velocity of $35\ \mathrm{m/s}$. Figs. 5.14 and 5.15 illustrate the motion trajectories of vehicles, as well as the longitudinal position, velocity, and lateral position profiles of the EV over the simulation for three MPCs. In this case, all three MPCs direct the EV to maintain the velocity and

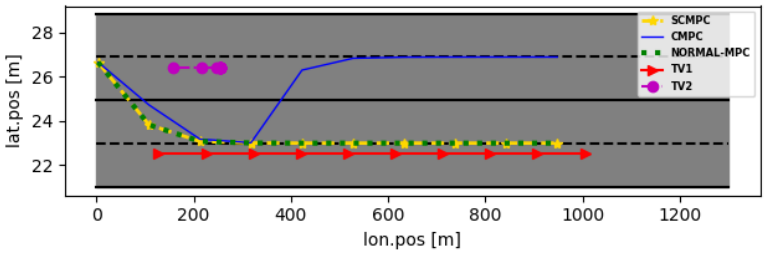


Fig. 5.12 Motion trajectories of vehicles (shown at each 3 s) under the "worst-case" scenario in Case 2

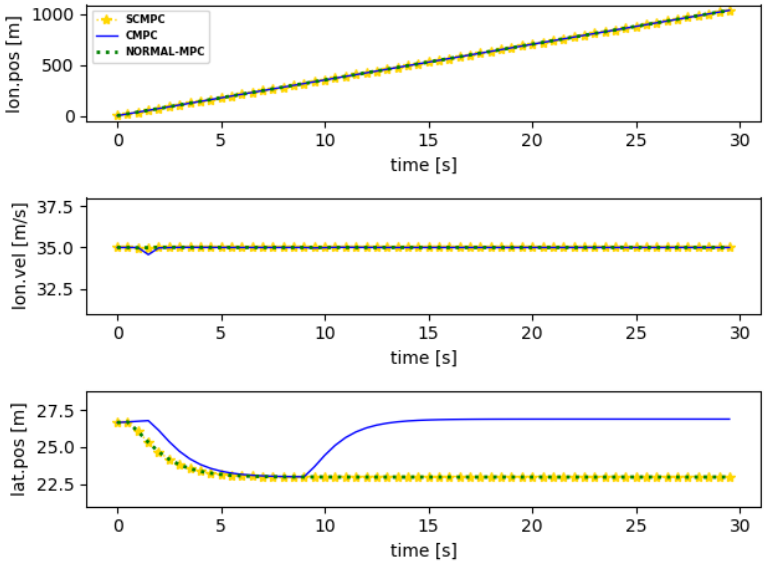


Fig. 5.13 EV's longitudinal position and velocity, and lateral position under the "worst-case" scenario in Case 2

the lane throughout the simulation. The EV performs almost identically under the three MPCs.

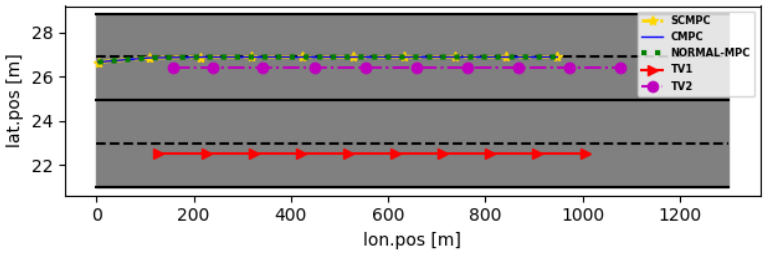


Fig. 5.14 Motion trajectories of vehicles (shown at each 3 s) when TV2 accelerates

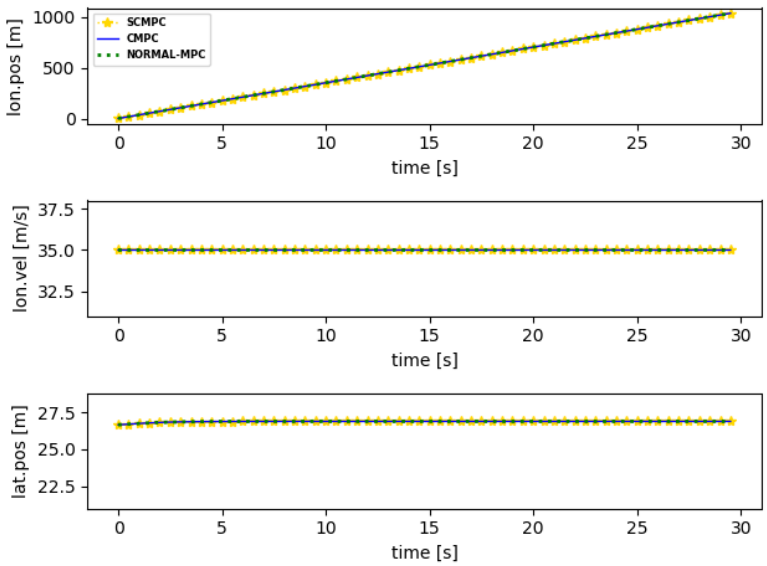


Fig. 5.15 EV's longitudinal position and velocity, and lateral position when TV2 accelerates

5.3.2.3 Case 3

In Case 3, the initial condition for TV2 is $x^{(\mathrm{TV2})} = \begin{bmatrix} 118.19 & 24 & -3 & 26.42 & 0 & 0 \end{bmatrix}^\top$, where the relative distance between the EV and TV2 is shorter than Case 2. The remaining initial conditions are identical to those in the first two cases. The corresponding open-loop safe sequences for the initial simulation time, based on the CMPC and SCMPC, are illustrated in Fig. 5.16, along with the open-loop normal sequences based on the NORMAL-MPC. At this time, the first-step longitudinal velocities obtained from safety-guaranteed MPCs, i.e., CMPC and SCMPC, are nearly identical, but different from the longitudinal velocity computed by the NORMAL-MPC. This difference arises because, given the initial condition, to respond to the potential "worst-case" scenario, the EV is forced to decrease velocity correspondingly under safety-guaranteed MPCs. However, based on the NORMAL-MPC, since the "worst-case" scenario is not considered, the EV just maintains its velocity, but it will collide with TV2 at 6 s, as indicated by a red marker in the first subplot. The results show that when the traffic situation is particularly challenging for the EV, the CMPC directs the EV to decelerate more aggressively to ensure safety. This required deceleration is greater than that calculated by the SCMPC. When considering the possibility of the lane change under the "worst-case" scenario, the closed-loop simulation results for the three MPCs are shown in Figs. 5.17 and 5.18. Under CMPC, the EV decelerates immediately, switches lanes after 2 s, accelerates back to 35 m/s, and returns to the original lane at 9 s due to lower lane-keeping costs. In contrast, the other two methods keep the EV at a constant velocity and in the new lane for the entire simulation.

In conclusion, from analyzing these cases, it is obtained that incorporating the "worst-

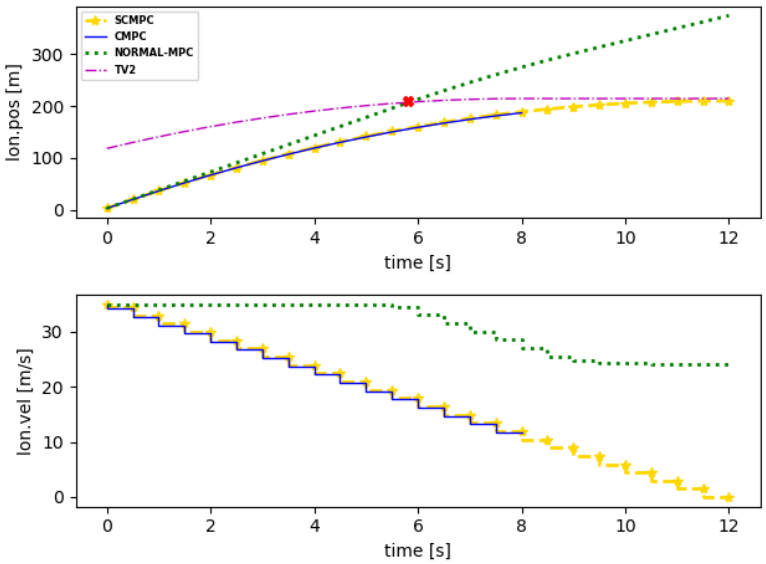


Fig. 5.16 Open-loop simulation results of the initial simulation time under the "worst-case" scenario in Case 3

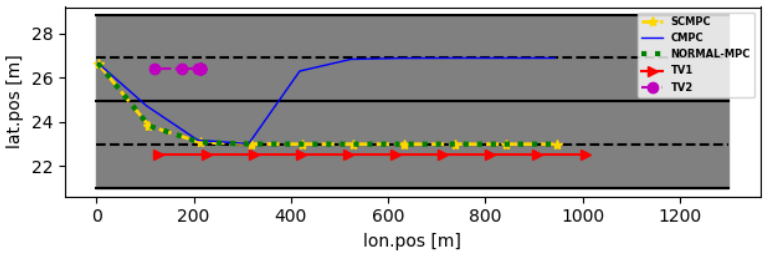


Fig. 5.17 Motion trajectories of vehicles (shown at each 3 s) under the "worst-case" scenario in Case 3

case" scenario into the controller design does not necessarily limit the EV's performance. When the "worst-case" scenario does not occur, the EV's performance across all three MPCs is nearly identical. However, when the "worst-case" scenario happens, integrating the "worst-case" scenario into the controller ensures the EV's safety. Moreover, the proposed CMPC leads the EV to operate less conservatively. Specifically, in response to this situation, the EV maintains the velocity as long as possible, before gradually decelerating to a maximum value. However, if the resulting cost of hard braking becomes too high, the EV will change lanes due to the consideration of the lane-change mode. Therefore, the integration of lane-keeping and lane-change control modes, along with the safety guarantee by accounting for the "worst-case" scenario, ensures that the proposed CMPC is both safe and minimally conservative in its control of the EV.

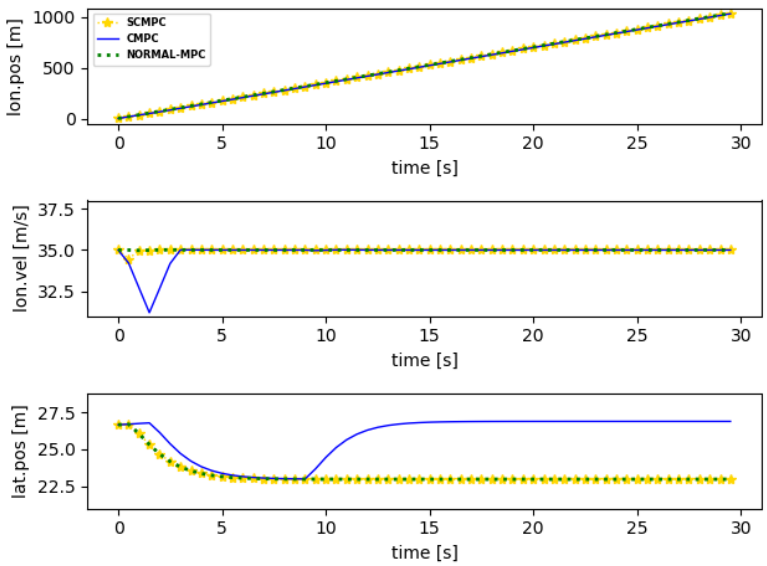


Fig. 5.18 EV's longitudinal position and velocity, and lateral position under the "worst-case" scenario in Case 3

5.4 Summary

In this chapter, a CMPC framework is presented by investigating the EV's reaction behavior under the "worst-case" scenario to reduce the conservatism of control actions. The CMPC has a similar structure to SCMPC introduced in Chapter 4, but with different terminal constraints. Based on the state estimation and prediction using improved IMM-KF, interaction-aware scenarios representing traffic uncertainties with high prediction accuracy are generated, which are integrated into the MPC design along with the proposed "worst-case" scenario. The CMPC is designed to allow the EV maintain its motion velocity and decelerate only when necessary to react to the "worst-case" scenario by aligning the normal state and the contingency state at the final step. This design principle reduces the conservatism of the EV's motion decisions, and the number of decision variables is also reduced by introducing time-varying prediction horizons. Furthermore, the proposed CMPC approach is validated and compared with the SCMPC, demonstrating the effectiveness of the proposed method in safely and less conservatively controlling the EV.

CHAPTER 6

Fully interaction-aware MPC for autonomous highway driving

6.1 Problem statement

In previous chapters, the presented SCMPC and CMPC approaches utilize the information of surrounding TVs, obtained from the scenario-generation component, for EV control. However, since MPC inherently solves an optimization problem in a receding horizon manner, it offers the potential to incorporate the states of both the TVs and the EV within a unified MPC framework while explicitly accounting for their interactions, i.e., a fully interaction-aware MPC. When all vehicle states are treated as decision variables within the MPC, it becomes difficult to characterize how one unknown decision variable influences another, and even more challenging to propagate such uncertainties over the prediction horizon. This significantly complicates interaction modeling within the MPC framework. Although several MPC-based methods have been developed following these principles, they typically suffer from high computational complexity and limited scalability, making real-time implementation difficult [140, 180].

To address these challenges, this chapter proposes an MPC-based scheme that simplifies interaction modeling within the MPC framework while reducing computational burden. Specifically, we aim to decouple the vehicle's longitudinal and lateral decision-making processes. By handling the lateral and longitudinal decisions separately, the complexity introduced by uncertainties in velocity and lane selection is reduced, resulting in a more tractable optimization problem that can be solved efficiently.

6.2 Interaction-aware lateral motion model and traffic prediction

To determine the lateral motion of vehicles over the prediction horizon while accounting for their interactions, we apply the MOBIL model. This model evaluates lane selection based on the resulting longitudinal acceleration benefit for all vehicles in the environment. Once the lateral positions of vehicles at each prediction step are determined, a portion of the traffic context becomes known. This information can then be utilized for the EV's subsequent path planning according to the possible lanes the EV may occupy, as well as for formulating safety constraints to avoid collisions with other vehicles in the same lane.

6.2.1 MOBIL model

The MOBIL model describes vehicle interactions by evaluating the advantages or disadvantages of a vehicle changing lanes in the context of the overall traffic situation. The utility of a lane and the associated risks of lane changes are determined according to the longitudinal accelerations computed using microscopic traffic models [120]. This compact model couples vehicle lateral and longitudinal motion strategies by introducing two design parameters, the politeness factor p_{f}, which quantifies the willingness of a vehicle to cooperate in traffic, and the acceleration threshold Δa_{th}, which determines the conditions under which a lane change is considered beneficial or risky. Specifically, a lane change is considered feasible if the utility gained by the involved vehicles exceeds a predefined threshold; otherwise, it is deemed infeasible. The utility is calculated by combining the change in longitudinal acceleration of the lane-changing vehicle with a weighted sum of the longitudinal acceleration changes of both its new and old followers. The MOBIL model is formally expressed as:

$$\underbrace{\bar{a}_{\mathrm{c}} - a_{\mathrm{c}}}_{\text{driver}} + p_{\mathrm{f}} \left(\underbrace{\bar{a}_{\mathrm{n}} - a_{\mathrm{n}}}_{\text{new follower}} + \underbrace{\bar{a}_{\mathrm{o}} - a_{\mathrm{o}}}_{\text{old follower}} \right) > \Delta a_{\mathrm{th}}, \tag{6.1}$$

where a_c and $\bar{a}_c$ represent the considered vehicle's longitudinal acceleration before and after lane change. The considered vehicle is named 'driver' for the simplicity of description, which can be either the TV or the EV. a_{n} and $\bar{a}_{\mathrm{n}}$ represent the longitudinal accelerations of the new follower of the driver before and after the driver lane change, respectively. a_{o} and $\bar{a}_{\mathrm{o}}$ represent the longitudinal accelerations of the old follower of the driver before and after the driver lane change. The politeness factor p_{f} ranges between $[0, 1]$, where a higher value indicates a more altruistic driving style. Since the degree of the driver's altruism is uncertain, multiple candidate values are considered to enrich the content of traffic prediction. Specifically, the values $p_{\mathrm{f}} = \{0, 0.5, 1\}$ are evaluated to capture different driving behaviors, ranging from self-interested to highly cooperative. Similarly, the switching threshold Δa_{th}, which serves as a baseline for evaluating the overall acceleration gains versus the acceleration losses of the driver lane change, is varied to examine its effect on the lane change decision. The values considered for Δa_{th} include $\Delta a_{\mathrm{th}} = \{0.1, 10\}$ [120].

To determine the vehicles' longitudinal acceleration values before and after the driver's lane change, the following rules are designed based on typical driver reactions in real traffic situations:

i) The vehicle's longitudinal acceleration after the driver's lane change depends on its new LV. If a new LV exists, the acceleration can be computed using a car-following model. Otherwise, the longitudinal acceleration value is assumed to be zero.

ii) The vehicle's current longitudinal acceleration is used as its acceleration before the

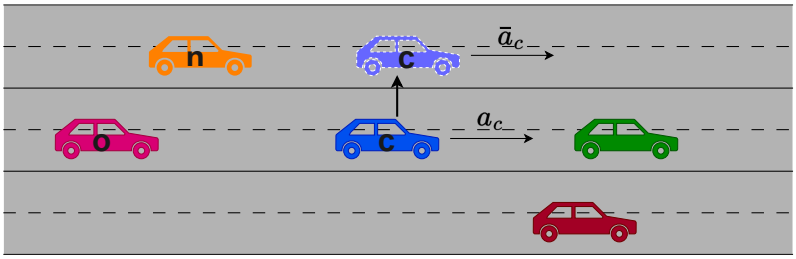


Fig. 6.1 Vehicle c (driver) considers changing lanes to the left. The new and old follower vehicles are denoted as n and o, respectively.

driver's lane change.

Remark 6.2.8. *Note that there is no need to use the MOBIL model to determine lane changes in the following cases: i) the vehicle drives in a lane without an LV; ii) the safety distance between the vehicle and its LV, or with a successor vehicle in the target lane, is not maintained over a certain prediction horizon. In these cases, the vehicle is assumed to remain in its current lane. Additionally, once a lane change is initiated, it is assumed that the vehicle will complete the maneuver.*

6.2.2 Interaction-aware collision-free traffic prediction

By evaluating the benefit of each vehicle's lane change with various design parameters, possible motion behaviors are predicted. For simplicity, it is assumed that lane changes happen instantly, i.e., between two subsequent time steps. For longer prediction horizons, the MOBIL model is applied at each decision-making interval T_{dk} to assess lane change possibilities. To predict vehicle motion over the horizon, a double-integrator motion model with a sampling time T is used to update the longitudinal state:

$$x^{(*)}_{\text{lon},k+1} = Ax^{(*)}_{\text{lon},k} + Bu^{(*)}_{\text{lon},k}, \tag{6.2a}$$

$$A = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} \frac{1}{2}T^2 & T \end{bmatrix}^\top, \tag{6.2b}$$

where the states, $x^{(*)}_{\text{lon},k} = \begin{bmatrix} p^{(*)}_{\text{lon},k}, & v^{(*)}_{\text{lon},k} \end{bmatrix}^\top$, $* \in \{\text{EV}, \text{TV}\}$, denote the vehicle's longitudinal position and velocity. The input, $u^{(*)}_{\text{lon},k} = a^{(*)}_{\text{lon},k}$, represents the longitudinal acceleration, as determined by the MOBIL model. After updating the vehicle's motion state using (6.2), if a collision with its LV is detected, the acceleration is adjusted to ensure collision-free prediction. At each prediction step, the relative distance between the vehicle and its LV is checked using the following condition:

$$p^{(\text{LV})}_{\text{lon},k} - p^{(*)}_{\text{lon},k} < \underline{d}, \tag{6.3}$$

where $* \in \{\text{EV}, \text{TV}\}$, and $p_{\text{lon},k}^{(\text{LV})}$ denotes the longitudinal position of its LV. $\underline{d}$ is a design parameter representing the relative distance threshold between the vehicle and its LV. A relatively large $\underline{d}$ is recommended to account for potential collisions. If the inequality in (6.3) is satisfied, the vehicle's longitudinal acceleration is adjusted to decelerate to maintain the distance:

$$\hat{a}_{\text{lon},k}^{(*)} = \underline{a}_{\text{lon}}, \tag{6.4}$$

where $\hat{a}_{\text{lon},k}^{(*)}$ is the adjusted longitudinal acceleration. $\underline{a}_{\text{lon}}$ is minimum allowable longitudinal acceleration. By combining the possible motion predictions of all vehicles, traffic environment uncertainties are represented in a deterministic manner.

Remark 6.2.9. *Only one vehicle's lane change is considered at each decision-making step. If a vehicle has just changed lanes or is predicted to change, lane changes are not considered in the subsequent steps.*

6.2.3 Traffic prediction categorization

To incorporate lateral interaction awareness into the controller design, the predicted motion behaviors of vehicles are categorized based on the EV's possible lateral motion behaviors. For example, in Fig. 6.1, the vehicle c is referred to as the EV. When the prediction horizon spans one decision step, based on the MOBIL model, the EV is predicted to drive in the current lane, or switch to the left or right lane. Then, all TVs' predicted behaviors are categorized according to these three potential EV motion modes. We take the EV's old follower (noted as 'o') and new follower (noted as 'n') in Fig. 6.1) as an example. In the EV's lane-keeping mode, vehicle o might remain in the middle lane or switch to either the left or right lane, while vehicle n might maintain the lane or switch to the right. Thus, three potential motion actions for vehicle o and two for vehicle n are considered in the EV's 'lane-keeping' motion mode. In contrast, for the EV's left-lane-change mode, assuming only one vehicle changes lanes at each decision step, lane changes by either vehicle o or n are not considered. Similarly, other TVs' motion actions are categorized and considered in the EV's different modes. This lateral traffic prediction process is illustrated in Fig. 6.2. The differences in TVs' motion behaviors across the EV's different modes reflect their mutual influence in the lateral direction.

Remark 6.2.10. *Besides using the MOBIL-based traffic prediction module, machine learning-based methods can also be employed to predict possible lateral motion behaviors of vehicles.*

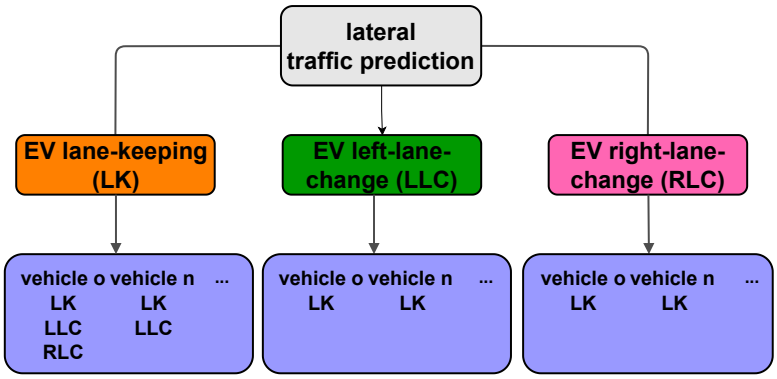


Fig. 6.2 Tree structure of lateral traffic prediction

6.3 Controller design and validation

6.3.1 Vehicle model

Based on the categorized traffic predictions, separate controllers are designed for each control mode. In the ith control mode, assume that n distinct traffic prediction branches are generated, with the jth branch denoted by $j = 1, 2, \ldots, n$. The motion states of the EV and TV are jointly modeled as follows. For a single surrounding TV, the discrete-time system model corresponding to the ith control mode and the jth traffic prediction branch is defined as:

$$X_{k+1}^{(i,j)} = \bar{A} X_k^{(i,j)} + \bar{B} U_k^{(i,j)}. \tag{6.5}$$

The associated full system state and input vectors are:

$$X_k^{(i,j)} = \Bigg[\underbrace{p_{\text{lon},k}^{(\text{EV})} \quad v_{\text{lon},k}^{(\text{EV})}}_{x_{\text{lon},k}^{(\text{EV}),(\text{i,j})}} \quad \underbrace{p_{\text{lat},k}^{(\text{EV})} \quad v_{\text{lat},k}^{(\text{EV})}}_{x_{\text{lat},k}^{(\text{EV}),(\text{i,j})}} \quad \underbrace{p_{\text{lon},k}^{(\text{TV})} \quad v_{\text{lon},k}^{(\text{TV})}}_{x_{\text{lon},k}^{(\text{TV}),(\text{i,j})}} \Bigg]^\top, \tag{6.6a}$$

$$U_k^{(i,j)} = \Bigg[\underbrace{a_{\text{lon},k}^{(\text{EV})} \quad a_{\text{lat},k}^{(\text{EV})}}_{u_k^{(\text{EV}),(i,j)}} \quad a_{\text{lon},k}^{(\text{TV})} \Bigg]^\top, \tag{6.6b}$$

where $p_{\text{lon},k}^{(*)}$, $v_{\text{lon},k}^{(*)}$, and $a_{\text{lon},k}^{(*)}$, with $* \in \{\text{EV}, \text{TV}\}$, denote the longitudinal position, velocity, and acceleration of the respective vehicle under the given mode (i, j). Here the superscript (i, j) is omitted for notational simplicity. Similarly, $p_{\text{lat},k}^{(\text{EV})}$, $v_{\text{lat},k}^{(\text{EV})}$, and $a_{\text{lat},k}^{(\text{EV})}$ represent the lateral position, velocity, and acceleration of the EV. The system dynamics are

governed by the following system and input matrices:

$$\bar{A} = \begin{bmatrix} A & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & A & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{0}_{2\times2} & A \end{bmatrix}, \tag{6.7a}$$

$$\bar{B} = \begin{bmatrix} B & B & B \end{bmatrix}^\top, \tag{6.7b}$$

where A and B are defined in (6.2b). The system state and input vectors, denoted by $X_k^{(i,j)}$ and $U_k^{(i,j)}$, along with the system and input matrices $\bar{A}$, $\bar{B}$, are appropriately augmented by the number of TVs considered in the model.

6.3.2 Fully interaction-aware MPC framework

Based on the EV's categorized motion modes, such as lane-keeping or lane change, multiple MPCs corresponding to different control modes are designed, each with a distinct reference trajectory. The CFTOCP is solved for each mode by minimizing the control efforts of both the EV and surrounding TVs while ensuring constraint satisfaction. Simultaneously, the EV aims to follow its assigned reference trajectory. After solving multiple CFTOCPs for all control modes, the decision-making module selects the first-step control action from the minimal-cost control mode by comparing the respective costs of each mode. Specifically, the CFTOCP for each control model is formulated as follows:

$$J^{(i)\star}(X_0) = \min_{U^{(i)},X^{(i)}} \sum_{k=0}^{N-1}\sum_{j=1}^{n} \left\| x_{k+1}^{(\mathrm{EV}),(i,j)} - x_{\mathrm{ref},k+1}^{(\mathrm{EV}),(i)} \right\|_{W_Q}^2 + \left\| U_k^{(i,j)} \right\|_{W_R}^2, \tag{6.8a}$$

$$\text{s.t. } X_{k+1}^{(i,j)} = \bar{A}X_k^{(i,j)} + \bar{B}U_k^{(i,j)}, \;\; k = 0,1,...,N-1, \; j = 1,2,...,n, \tag{6.8b}$$

$$X_k^{(i,j)} \in \mathbb{X}, \;\; k = 0,1,...,N-1, \; j = 1,2,...,n, \tag{6.8c}$$

$$U_k^{(i,j)} \in \mathbb{U}, \;\; k = 0,1,...,N-1, \; j = 1,2,...,n, \tag{6.8d}$$

$$u_0^{(\mathrm{EV}),(i,1)} = u_0^{(\mathrm{EV}),(i,2)} = ... = u_0^{(\mathrm{EV}),(i,n)}, \tag{6.8e}$$

$$X_0^{(i,j)} = X(0), \;\; j = 1,2,...,n. \tag{6.8f}$$

Here, decision variables $U^{(i)}$ and $X^{(i)}$ represent the feasible inputs and states in the ith control mode across n traffic prediction branches over the prediction horizon N, respectively. Specifically,

$$U^{(i)} = \begin{bmatrix} U_{0,1...,N-1}^{(i,1)} & U_{0,1...,N-1}^{(i,2)} & ... & U_{0,1...,N-1}^{(i,n)} \end{bmatrix}^\top, \tag{6.9a}$$

$$X^{(i)} = \begin{bmatrix} X_{1,2...,N}^{(i,1)} & X_{1,2...,N}^{(i,2)} & ... & X_{1,2...,N}^{(i,n)} \end{bmatrix}^\top. \tag{6.9b}$$

As defined in (6.6b), $U_k^{(i,j)}$ consists of the longitudinal and lateral inputs (acceleration) of the EV under the jth traffic prediction branch, represented by $u_k^{(\mathrm{EV}),(i,j)}$, as well as the

longitudinal input of the TVs. Similarly, $X_k^{(i,j)}$ consists of the corresponding longitudinal and lateral states (position and velocity) of the EV, represented by $x_{\text{lon},k}^{(\text{EV}),(\text{i,j})}$ and $x_{\text{lat},k}^{(\text{EV}),(\text{i,j})}$, along with the longitudinal states of TVs. $x_k^{(\text{EV}),(i,j)}$ represents the full state of the EV, i.e., $x_k^{(\text{EV}),(i,j)} = \left[x_{\text{lon},k}^{(\text{EV}),(i,j)}, \quad x_{\text{lat},k}^{(\text{EV}),(i,j)}\right]^\top$. Additionally, $x_{\text{ref},k+1}^{(\text{EV}),(i)}$ represents the reference state for the EV in the ith control mode. The reference trajectory directs the EV to maintain its velocity and travel in the center of the desired lane. For lane-change mode, the reference trajectory guides the EV to switch to the desired lane after one prediction step. The matrices $W_Q \in \mathbb{R}^{4\times4}$, and $W_R \in \mathbb{R}^{(m+2)\times(m+2)}$ are positive definite weighting matrices used for tuning, where m represents the number of surrounding TVs. The sets $\mathbb{X}$ and $\mathbb{U}$ represent admissible state and input sets based on the constraints from Section 6.3.3. To ensure the EV's safety in an uncertain environment, the first control action from each traffic prediction branch is set identically, as described in (6.8e). The initial state is denoted as $X(0)$. As the prediction horizon N increases, the EV has more lane change options. The number of control modes for the EV increases. However, due to computational cost, the number of control modes investigated is limited to five.

6.3.3 Constraint formulation

The constraints are formulated by considering traffic rules, driving comfort, and safety. The longitudinal velocity and the lateral position of vehicles are restricted by speed limitations and lane bounds. Driving comfort is guaranteed by limiting the accelerations.

$$0 < v_{\text{lon},k}^{(*)} < \overline{v}_{\text{lon}}, \quad l_{\text{lb}} \leq p_{\text{lat},k}^{(\text{EV})} \leq l_{\text{ub}}, \tag{6.10a}$$

$$\underline{a}_{\text{lon}} \leq a_{\text{lon},k}^{(*)} \leq \overline{a}_{\text{lon}}, \quad \underline{a}_{\text{lat}} \leq a_{\text{lat},k}^{(\text{EV})} \leq \overline{a}_{\text{lat}}, \tag{6.10b}$$

where $[l_{\text{ub}}, l_{\text{lb}}]$ represent the lane bounds of the studied lane. $\underline{\bullet}$ and $\overline{\bullet}$ denote the related variable's minimum and maximum value. A relative longitudinal distance between the vehicle and its LV is maintained all the time to ensure safety as follows:

$$p_{\text{lon},k}^{(\text{LV})} - p_{\text{lon},k}^{(*)} > \underline{d}, \tag{6.11}$$

where $p_{\text{lon},k}^{(\text{LV})}$ represents the longitudinal position of the LV of the studied vehicle, and $* \in \{\text{EV}, \text{TV}\}$. The design parameter $\underline{d}$ represents the safety distance between two vehicles, which satisfies

$$\underline{d} \geq \frac{b^{(\text{LV})} + b^{(*)}}{2}, \tag{6.12}$$

where b denotes the width of the corresponding vehicle.

For the most Preceding Vehicle (PV) in each lane, it is assumed that it maintains its

velocity over the prediction horizon. Thus, its longitudinal acceleration is constrained by:

$$a_{\text{lon},k}^{(\text{PV})} = 0. \tag{6.13}$$

Remark 6.3.11. *The longitudinal states of TVs are treated as decision variables to ensure that, at each prediction step, TVs and the EV interact longitudinally to optimize overall traffic flow. The optimal longitudinal interaction-aware mechanism of vehicles is determined directly by solving the problem. Alternatively, to explicitly account for the longitudinal interaction awareness of TVs in MPC, their longitudinal accelerations can be constrained as follows:*

$$a_{lon,k}^{(TV)} = K_v(v_{lon,k}^{(LV)} - v_{lon,k}^{(TV)}) + K_d(d_{lon,k}^{(TV)} - \underline{d}^{(TV)}), \tag{6.14}$$

where K_v, K_d are the 'VT' and 'DK' control gains, respectively. $v_{lon,k}^{(LV)}$ represents the longitudinal velocity of the LV of the TV. $d_{lon,k}^{(TV)}$ is the relative longitudinal distance between the TV and its LV. $\underline{d}^{(TV)}$ denotes the safety distance for the TV. The parameters K_v, K_d, and $\underline{d}^{(TV)}$ are design parameters that characterize each TV, and can be selected, estimated, or learned appropriately.

6.4 Simulation and discussion

6.4.1 Simulation setup

The proposed control structure is validated in a high-fidelity environment, IPG CarMaker. In this environment, a real car is represented as a simulated EV, and TVs interact with surrounding vehicles based on the HDM. The HDM provides a sophisticated representation of longitudinal and lateral maneuvers grounded in real-world driver behavior. The HDM used in IPG CarMaker is built upon the Improved Intelligent Driver Model (IIDM) [181], a widely recognized model that enhances the realism of acceleration and deceleration behaviors compared to conventional IDM [182]. Specifically, the HDM allows for detailed customization of parameters for both individual TVs and overall traffic flow. Each TV's longitudinal and lateral driving style is configured using comprehensive parameters normalized within the range $[0, 1]$. To properly implement the calculated control action in IPG CarMaker, the longitudinal control input is converted to gas or a brake torque, while the lateral control inputs are converted to steering angle, as described in 4.3.

Three representative cases are first studied to demonstrate the effectiveness of the proposed method in accounting for vehicle interactions during EV control. For comparison, the EV is also controlled by a non-interaction-aware MPC in these cases, where vehicles are not modeled to respond to each other within the receding horizon. In this approach, the motion predictions of TVs are generated using a Constant Velocity model. The non-

interaction-aware MPC shares the same control goals and constraints as the interaction-aware controller. Moreover, the proposed control structure is validated with Monte Carlo simulations under different initial traffic conditions and diverse driving styles. The parameters used in the simulations are listed in Table. 6.1. The total simulation time is denoted as t_s. Lane bounds and central positions are represented by sets l_b, and l_c. The height and width of vehicles {EV, TV1, TV2, TV3} are given in the sets l and b.

Table 6.1 Design parameters used in the proposed MPC scheme

$t_\mathrm{s}[\mathrm{s}]$	$T[\mathrm{s}]$	p	$\Delta a_\mathrm{th}[\mathrm{m/s^2}]$
30.00	1.00	$\{0.00, 0.50, 1.00\}$	$\{0.10, 10\}$
N	$T_\mathrm{dk}[\mathrm{s}]$	$\underline{a}_\mathrm{lon}/\underline{a}_\mathrm{lat}[\mathrm{m/s^2}]$	$l_\mathrm{b}[\mathrm{m}]$
6	6	-3.00	$\{-14, -10.25, -6.5, -2.75\}$
$\underline{d}[\mathrm{m}]$	W_Q/W_R	$l[\mathrm{m}]$	$b[\mathrm{m}]$
20.00	$\boldsymbol{I}_4/\boldsymbol{I}_5$	$\{4.58, 4.97, 4.15, 4.12\}$	$\{1.59, 1.97, 1.73, 1.80\}$
$\underline{d}[\mathrm{m}]$	$\overline{v}_\mathrm{lon}[\mathrm{m/s}]$	$\overline{a}_\mathrm{lon}/\overline{a}_\mathrm{lat}[\mathrm{m/s^2}]$	$l_\mathrm{c}[\mathrm{m}]$
6.00	40.00	3.00	$\{-12.13, -8.38, -4.63\}$

6.4.2 Case studies

6.4.2.1 Case 1: TV2 nudges the EV to change lanes

The initial conditions for vehicles are: $x^{(\mathrm{EV})} = \begin{bmatrix}372 & 18 & 0 & -8.38 & 0 & 0\end{bmatrix}^\top$, $x^{(\mathrm{TV1})} = \begin{bmatrix}404 & 13 & 0 & -12.13 & 0 & 0\end{bmatrix}^\top$, $x^{(\mathrm{TV2})} = \begin{bmatrix}325 & 28 & 0 & -8.38 & 0 & 0\end{bmatrix}^\top$, and $x^{(\mathrm{TV3})} = \begin{bmatrix}322 & 20 & 0 & -4.63 & 0 & 0\end{bmatrix}^\top$, respectively. The initial traffic scene of this case is shown in Fig. 6.3.

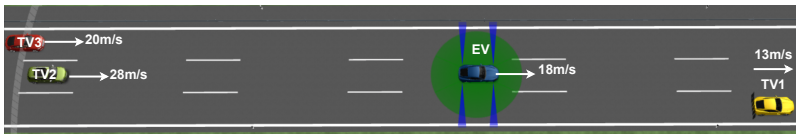


Fig. 6.3 Initial traffic scene of Case 1 used for the proposed MPC structure validation

Fig. 6.4 shows the motion state information of all vehicles during the simulation. Subplot (1) represents the position of each vehicle at every 2 s. The EV starts to change lanes to the left at the first time step, and completes the lane change around after 4 s. Since the EV is initially much slower than its successor, TV2, changing lanes is more beneficial for the overall traffic situation than staying in the current lane. The left lane change is considered safe by the EV under interaction-aware control, as it 'communicates' with TV3, and anticipates that TV3 will decelerate to avoid a collision. However, the right lane change is considered dangerous because TV1, driving ahead of the EV at a relatively slow speed, would make it less beneficial for the traffic environment. From subplots (2) and

(4), we observe that, when the EV enters the left lane at 2 s, TV3 does start to decelerate from 20 m/s, eventually matching the EV's velocity. Meanwhile, TV1 and TV2 continue to drive at constant velocity in their respective lanes.

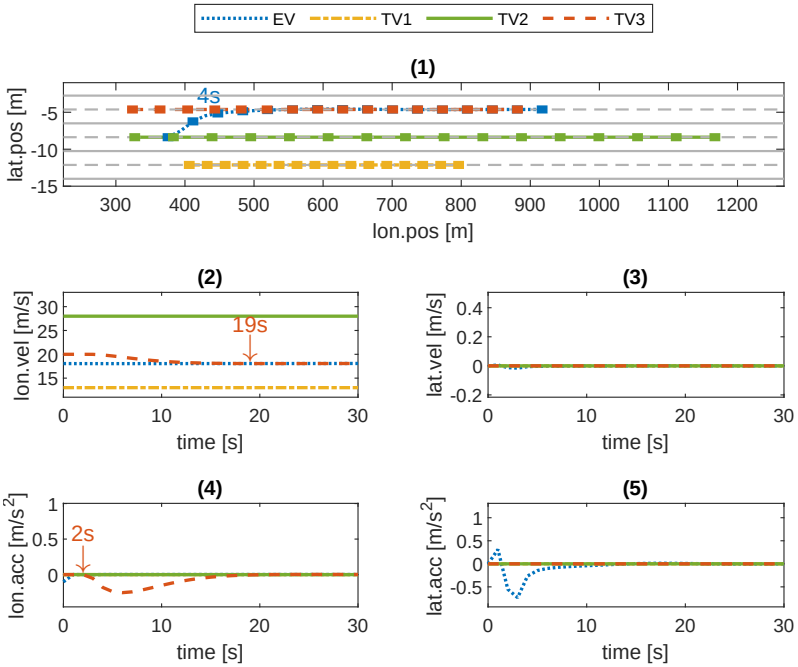


Fig. 6.4 Motion trajectories and velocity/acceleration profiles of vehicles under the interaction-aware MPC in Case 1

Fig. 6.5 shows the simulation results under the non-interaction-aware control in Case 1. Subplot (1) indicates that the EV stays in its lane throughout the simulation. As shown in subplots (2) and (4), TV2 decelerates sharply and converges to the EV's speed from the beginning until 9 s to ensure safety. In contrast to the interaction-aware control pattern, lane changes are not feasible for the EV under the non-interaction-aware control. This is because, without considering the interaction between TV3 and the EV, TV3 is predicted to maintain a constant speed rather than slow down if the EV moves in front of it. The right lane change is also neither possible nor desirable for the EV, as TV1 blocks the lane with a slower velocity. Afterward, TV2 and the EV drive at 18 m/s, making lane-keeping the optimal decision for the EV. For comparison, two traffic scene screenshots from the IPG CarMaker at 8 s under the interaction-aware and non-interaction-aware MPCs are shown in Fig. 6.6.

6.4.2.2 Case 2: EV changes lanes to avoid collision with TV1

The initial states of vehicles in this case are: $x^{(\mathrm{EV})} = \begin{bmatrix} 372 & 25 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV1})} = \begin{bmatrix} 428 & 18 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV2})} = \begin{bmatrix} 434 & 15 & 0 & -12.13 & 0 & 0 \end{bmatrix}^\top$, and $x^{(\mathrm{TV3})} = \begin{bmatrix} 302 & 28 & 0 & -4.63 & 0 & 0 \end{bmatrix}^\top$, respectively. The initial traffic scene of Case 2

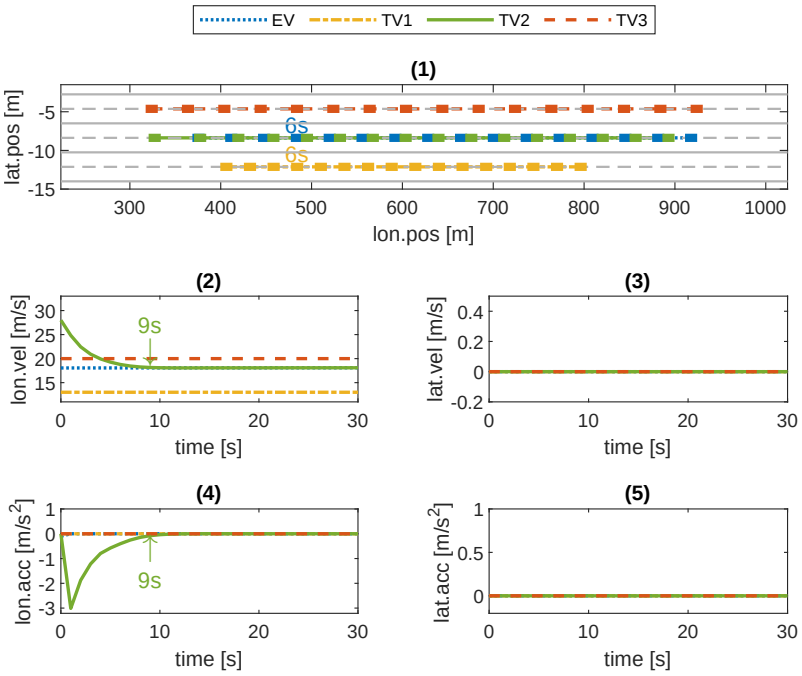


Fig. 6.5 Motion trajectories and velocity/acceleration profiles of vehicles under the non-interaction-aware MPC in Case 1

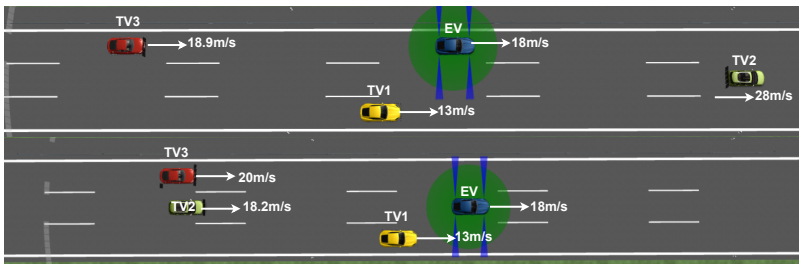


Fig. 6.6 Traffic scenes at 8 s under the interaction-aware (top) and non-interaction-aware (bottom) MPCs in Case 1

is shown in Fig. 6.7.

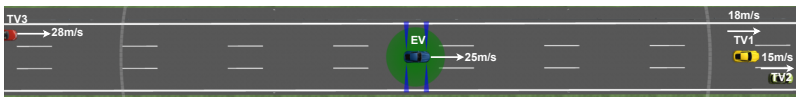


Fig. 6.7 Initial traffic scene of Case 2 used for the proposed MPC structure validation

The simulation results for Case 2 under the interaction-aware control are shown in Fig. 6.8. Subplot (1) illustrates the motion of each vehicle at every 3 s, showing that the EV stays in the middle lane for the first 3 s. During this time, the relative distance between the EV and its LV, TV1, decreases due to the EV's higher speed. At around 6 s, the EV switches to the left lane to avoid decelerating in the current lane and causing traffic congestion. The EV predicts a safe lane change to the left based on the expected deceleration of TV3 under the interaction-aware control. Subplots (2) and (4) show the longitudinal velocity and acceleration profiles of TV3, which decelerates when the EV

enters the left lane at around 4 s, eventually matching the EV's speed. Meanwhile, TV1 and TV2 drive continuously at a constant speed in their lanes.

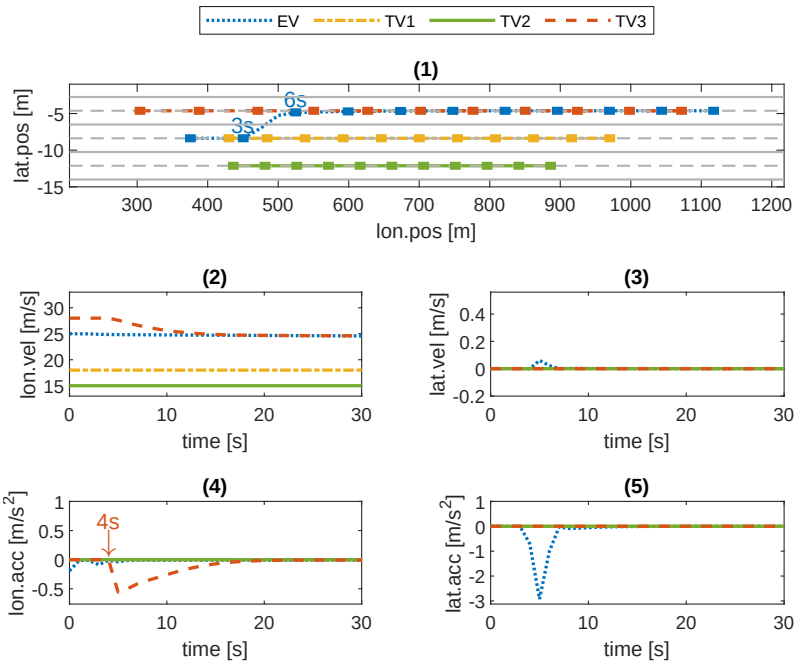


Fig. 6.8 Motion trajectories and velocity/acceleration profiles of vehicles under the interaction-aware MPC in Case 2

The simulation results for Case 2 under the non-interaction-aware control are shown in Fig. 6.9. Subplot (1) presents the position of all vehicles at each 3 s. Combined with vehicles' longitudinal velocity and acceleration profiles in subplots (2) and (4), it is shown that all TVs maintain their initial speed in their respective lanes throughout the simulation. Additionally, the EV begins to slow down with an acceleration above $-1\ \mathrm{m/s^2}$ at 2 s. In contrast to EV's lane change decision under the interaction-aware control, the EV decides to keep the lane and brake. Lane changes are infeasible since TV2 and TV3 are predicted to drive at a constant velocity if the EV moves into their lane. As a result, the EV must stay in its lane and brake to ensure safety. After around 10 s, its speed equals TV1's. Two traffic scene screenshots from the IPG CarMaker at 12 s under the interaction-aware and non-interaction-aware MPCs are shown in Fig. 6.10. Compared to the EV's performance under the interaction-aware control, the inability to predict the TVs' potential reactions under the non-interaction-aware control leads the EV to behave more conservatively by braking, potentially worsening the overall traffic situation.

6.4.2.3 Case 3: TV1 changes lanes to the front of the EV

Vehicles' initial motion states in this case are: $x^{(\mathrm{EV})} = \begin{bmatrix} 299 & 18 & 0 & -12.13 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV1})} = \begin{bmatrix} 355 & 13 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, $x^{(\mathrm{TV2})} = \begin{bmatrix} 265 & 25 & 0 & -4.63 & 0 & 0 \end{bmatrix}^\top$, and

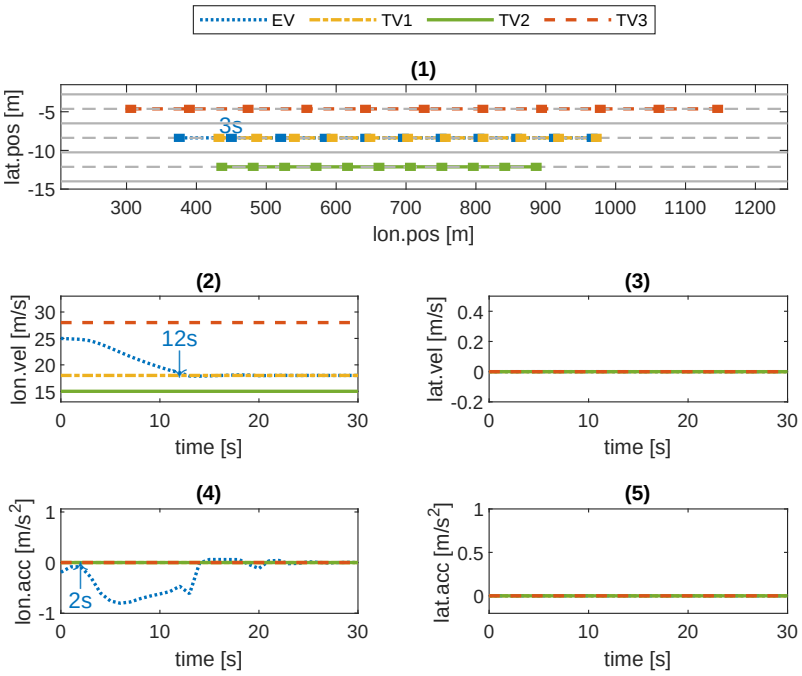


Fig. 6.9 Motion trajectories and velocity/acceleration profiles of vehicles under the non-interaction-aware MPC in Case 2

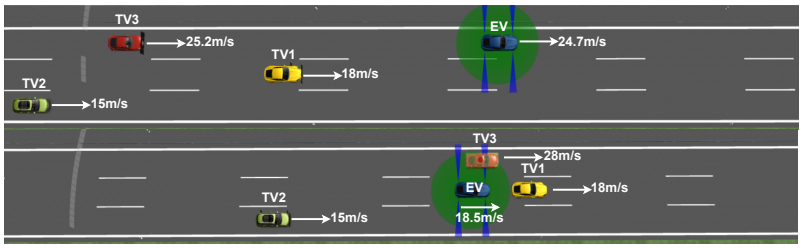


Fig. 6.10 Traffic scenes at 12 s under the interaction-aware (top) and non-interaction-aware (bottom) MPCs in Case 2

$x^{(\mathrm{TV3})} = \begin{bmatrix} 210 & 28 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, respectively. As TV1 drives much slower in front of TV3, and considering the left lane is blocked by TV2 with a higher velocity, TV1 is set to start to switch to the right lane at 7 s, and complete the lane change within the next 4 s. The initial traffic scene of Case 3 is shown in Fig. 6.11.

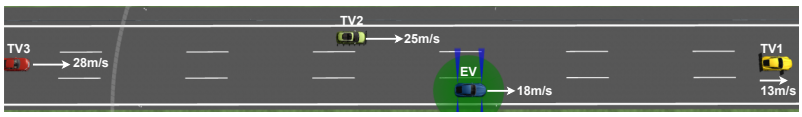


Fig. 6.11 Initial traffic scene of Case 3 used for the proposed MPC structure validation

The simulation results for Case 3 under the designed interaction-aware controller are shown in Figs. 6.12 and 6.13. Subplot (1) of Fig. 6.12 shows the trajectories of all vehicles at each 5 s. Detailed position information is provided in Fig. 6.13, including the enlarged view of the EV and TV1's longitudinal position profile from 8 s to 14 s. Subplot (2) of Fig. 6.12 displays the longitudinal velocity profile of vehicles. TV3 decelerates from 3 s

to 9 s before TV1 moves to the new lane. Afterward, it accelerates back to the initial velocity and drives in the middle lane. After interacting with the surrounding vehicles, the EV begins to slow down at 5 s as it anticipates that the TV1 will move in front of it in a few time steps. When TV1 indeed moves to its front at around 9 s, in response, the EV is constantly braking with the maximum deceleration of -1.8 m/s^2 until 12 s, its velocity eventually converges to the velocity of TV1. Fig. 6.14 illustrates the traffic prediction result at 6 s, where four scenarios are predicted. These scenarios include the EV performing lane-keeping, left lane change, and right lane change maneuvers, leading to the design of three corresponding MPC control modes.

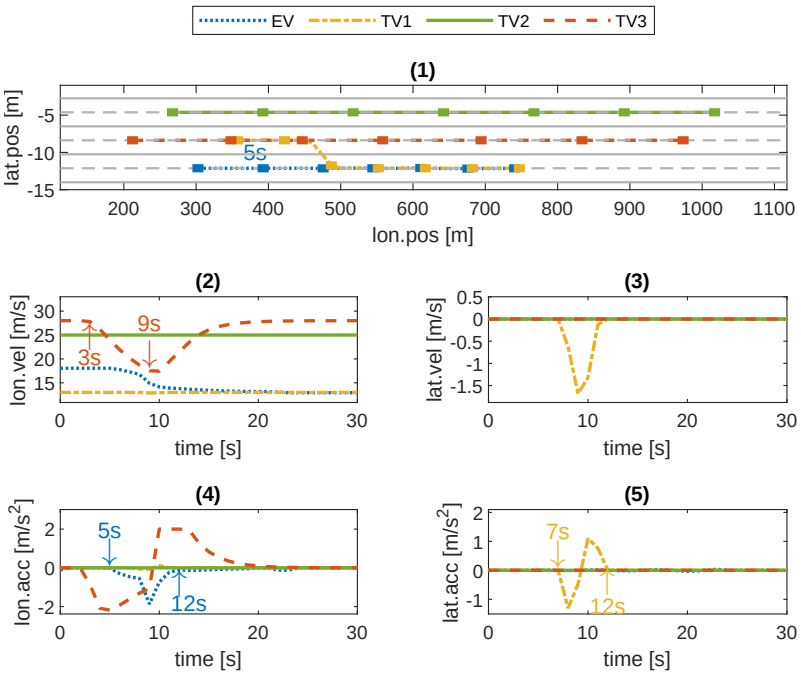


Fig. 6.12 Motion trajectories and velocity/acceleration profiles of vehicles under the interaction-aware MPC in Case 3

The simulation results for Case 3 under the non-interaction-aware control are presented in Fig. 6.15. Similar to the simulation results in the interaction-aware control pattern, before TV1 moves to the right lane, TV3 is continuously slowing down from 3 s to 9 s. Then, TV3 increases the speed to the initial value. For the EV, it decelerates at 9 s with the deceleration maximum of -2.9 m/s^2 when TV1 drives in front of it. Compared to the reaction under the interaction-aware control, the EV brakes later and more sharply as TV1 is not predicted to do the lane change by the EV before it does happen. Then, the EV converges to the velocity of TV1. Two traffic scene screenshots from the IPG CarMaker at 9 s under the interaction-aware and non-interaction-aware MPCs are shown in Fig. 6.16.

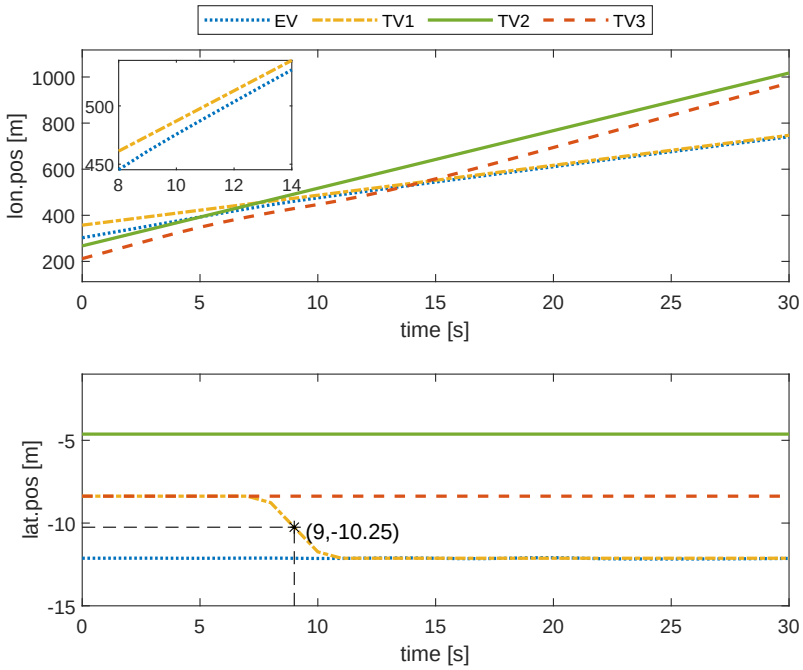


Fig. 6.13 Longitudinal and lateral position profiles of vehicles under the interaction-aware MPC in Case 3

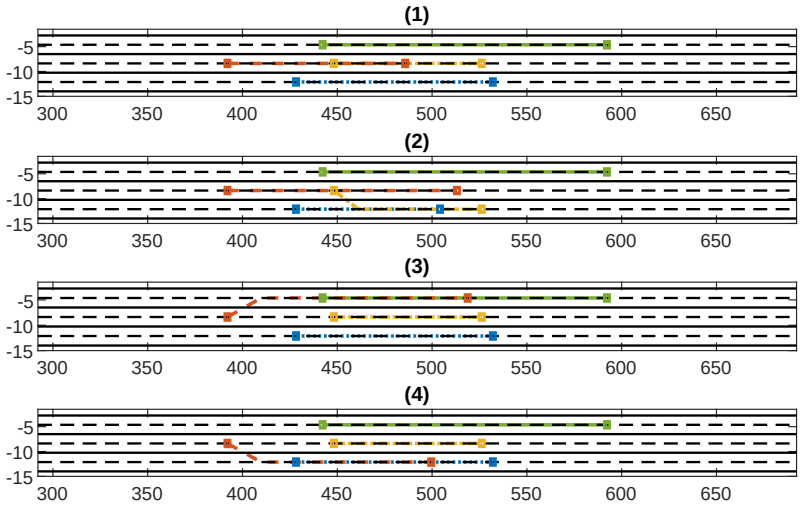


Fig. 6.14 Traffic prediction at 6 s in Case 3

6.4.3 Monte Carlo simulation

To rigorously validate the proposed control structure, Monte Carlo simulations are conducted by varying the TVs' initial conditions and driving styles. Specifically, we use Case 2 from Section 6.4.2 as the base scenario, and vary the initial condition of TVs according to Normal distributions. For example, in the base scenario, TV1's initial longitudinal position and velocity are given by $\begin{bmatrix} p_{\text{lon,base}}^{(\text{TV1})} & v_{\text{lon,base}}^{(\text{TV1})} \end{bmatrix}^\top$. In the Monte Carlo simulations, the initial longitudinal position and velocity of TV1 follow Normal distributions centered

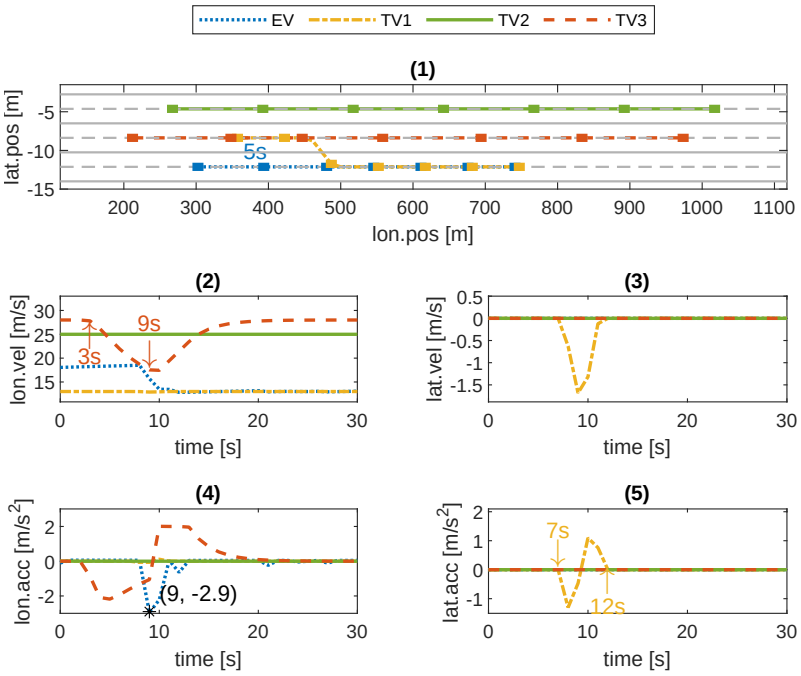


Fig. 6.15 Motion trajectories and velocity/acceleration profiles of vehicles under the non-interaction-aware MPC in Case 3

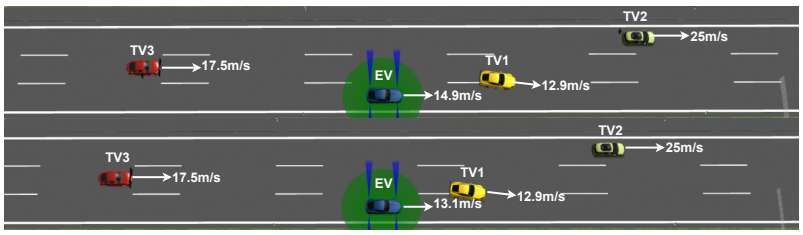


Fig. 6.16 Traffic scenes at 9 s under the interaction-aware (top) and non-interaction-aware (bottom) MPCs in Case 3

around these base values, as follows:

$$p_{\text{lon},0}^{(\text{TV1})} \sim \mathcal{N}(p_{\text{lon,base}}^{(\text{TV1})}, 20),$$
$$v_{\text{lon},0}^{(\text{TV1})} \sim \mathcal{N}(v_{\text{lon,base}}^{(\text{TV1})}, 5).$$

Similarly, the other TVs' initial longitudinal position and velocity in the Monte Carlo simulations are varied based on the base scenario, with a position variance of 20 m, and a velocity variance of 5 m/s. To ensure realistic and diverse behavior of the TVs, key HDM parameters are adjusted to reflect different driving styles. In particular, three different driving styles, are considered for TV1 and TV3: defensive (D), normal (N), and aggressive (A), achieved by varying the key parameters in the HDM [181]. The driving style of TV2 is set to normal and remains unchanged due to its limited influence on the decision-making of the EV. The specific parameter set used in the driving style configuration is provided in B. For the Monte Carlo simulations, we first run 120 simulations with

different initial conditions for TVs based on a specific set of TVs' driving styles. Then, we change TV1 and TV3 driving styles from defensive to aggressive, considering 120 different initial conditions of the TVs. Thus, in total, we run $1080(120 \times 9)$ Monte Carlo simulations using both the proposed interaction-aware controller and the non-interaction-aware controller. The simulation time for each case is 15 s.

Considering the different driving styles of TV1 and TV3 (collectively and sequentially listed in square brackets), the simulation results are analyzed using evaluation metrics, including the number of feasible simulations (denoted as 'feasible'), EV lane-change simulations (denoted as 'lane-change'), and EV lane-keeping simulations (denoted as 'lane-keeping'), along with the corresponding percentages over the entire set of 120 simulations.

Table 6.2 Monte Carlo results: interaction-aware vs non-interaction-aware controllers with different TV driving styles

	feasible	lane-change	lane-keeping
interaction-aware [DD]	120 (100%)	91 (75.8%)	29 (24.2%)
non-interaction-aware [DD]	104 (86.7%)	1 (0.8%)	103 (85.8%)
interaction-aware [DN]	120 (100%)	89 (74.8%)	31 (25.2%)
non-interaction-aware [DN]	103 (85.8%)	1 (0.8%)	102 (85%)
interaction-aware [DA]	120 (100%)	92 (76.7%)	28 (23.3%)
non-interaction-aware [DA]	103 (85.8%)	1 (0.8%)	102 (85%)
interaction-aware [ND]	120 (100%)	91 (75.8%)	29 (24.2%)
non-interaction-aware [ND]	103 (85.8%)	1 (0.8%)	102 (85%)
interaction-aware [NN]	120 (100%)	90 (75%)	30 (25%)
non-interaction-aware [NN]	103 (85.8%)	1 (0.8%)	102 (85%)
interaction-aware [NA]	120 (100%)	91 (75.8%)	29 (24.2%)
non-interaction-aware [NA]	103 (85.8%)	1 (0.8%)	102 (85%)
interaction-aware [AD]	120 (100%)	91 (75.8%)	29 (24.2%)
non-interaction-aware [AD]	104 (86.7%)	1 (0.8%)	103 (85.8%)
interaction-aware [AN]	120 (100%)	92 (76.7%)	28 (23.3%)
non-interaction-aware [AN]	104 (86.7%)	1 (0.8%)	103 (85.8%)
interaction-aware [AA]	120 (100%)	90 (75%)	30 (25%)
non-interaction-aware [AA]	103 (85.8%)	1 (0.8%)	102 (85%)

Table 6.2 presents the Monte Carlo simulation results under different driving styles of the TVs. As an example, we consider the results where both TV1 and TV3 follow the driving style [NN]. In this scenario, the interaction-aware controller results in all simulations being feasible, whereas the non-interaction-aware controller results in 103 feasible simulations (85.8%). In the remaining infeasible cases, the EV executes lane changes under the interaction-aware controller, indicating that it anticipates potential infeasibility and proactively avoids it through lane changes. Moreover, the EV changes lanes more often under the interaction-aware controller, with 90 out of 120 (75%) simulations involving

lane changes, compared to only one lane change observed with the non-interaction-aware controller. These results suggest that the interaction-aware controller enables a more efficient traffic flow, while the non-interaction-aware controller behaves more conservatively, leading to higher infeasibility. Furthermore, under the non-interaction-aware controller, the EV mainly maintains the lane, with 102 out of 120 (85%) simulations, whereas the interaction-aware controller demonstrates a more adaptive balance between lane-change and lane-keeping behaviors. The results also indicate that the proposed interaction-aware controller remains robust across different driving styles. Variations in the behavior of surrounding vehicles do not significantly impact its performance. This adaptability further demonstrates the controller's capability to ensure safe and efficient autonomous driving in diverse traffic environments. However, it is worth noting that the EV's decision-making differs depending on the TVs' driving styles, even though the overall statistical results across different styles appear similar. For example, given the initial conditions $x^{(\text{EV})} = \begin{bmatrix} 372 & 25 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, $x^{(\text{TV1})} = \begin{bmatrix} 417 & 21 & 0 & -8.38 & 0 & 0 \end{bmatrix}^\top$, $x^{(\text{TV2})} = \begin{bmatrix} 433 & 4.73 & 0 & -12.13 & 0 & 0 \end{bmatrix}^\top$, and $x^{(\text{TV3})} = \begin{bmatrix} 285 & 28 & 0 & -4.63 & 0 & 0 \end{bmatrix}^\top$, when both TV1 and TV3 follow the normal driving style [NN], the EV performs a left lane change at 7 s and drives in front of TV3 under the interaction-aware controller. In contrast, when both TV1 and TV3 adopt a defensive style [DD], the EV instead switches to the right lane and drives in front of TV2 at 8 s under the interaction-aware controller.

6.5 Summary

This chapter presents a fully interaction-aware control framework for safe and efficient EV control in uncertain traffic environments. The longitudinal states of both the EV and TVs are treated as MPC decision variables while accounting for their interactions. The MOBIL model is used to describe the lateral interactions between vehicles. The lateral decision uncertainties of vehicles are modeled using distinct traffic predictions, which are categorized based on the EV's potential lateral decisions. These lateral decisions correspond to different MPC control modes. In each control mode, and for each traffic prediction branch, the associated longitudinal states and inputs of the EV are computed, with the first control input being identical across all branches. Among all control modes, the one with the minimum cost is selected for execution. The proposed control approach is validated across three traffic scenarios and through a Monte Carlo simulation study. Simulation results demonstrate that the EV consistently perform safe maneuvers under the proposed control regime. Compared with a non-interaction-aware controller, the EV also behaves less conservatively through possible lane changes.

CHAPTER 7

Findings, conclusions, and future work

The thesis is motivated by the challenges associated with safely and efficiently controlling EVs in uncertain highway environments. The primary objectives are to design MPC control structures capable of capturing uncertainties in vehicle motion behaviors and interactions, to determine appropriate motion strategies for EVs, and to formulate computationally efficient optimization problems. A series of research questions are formulated based on these objectives in Section 1.4 and are addressed throughout Chapters 3–6. This chapter systematically summarizes the corresponding findings.

7.1 Findings and conclusions

Control methodology and algorithm derivation

MPC is an effective approach for handling uncertainties in traffic environments and supporting decision-making for EVs. This research addresses the question of *How can a control framework be developed to enable the implementation of both planning and control for the EV* by proposing a unified MPC formulation in which reference generation and control optimization are performed simultaneously. By tracking a specific velocity (e.g., the current velocity or a designated high-speed value) within the MPC framework, the proposed approaches eliminate the need for a separate high-level motion planner, leading to a compact control structure and avoiding delays associated with multi-layer planning. Both the **MPC with interaction-aware predictions** and the **fully interaction-aware MPC** frameworks adopt this unified design principle, demonstrating that planning and control can be effectively integrated within an MPC-based structure without relying on external planners. However, generating reference trajectories by simply tracking the current velocity may sacrifice maneuvering flexibility, since accelerating back to the original or a higher velocity may become infeasible once the EV decelerates. This limitation motivates the need for more advanced reference generation strategies that preserve feasibility while retaining sufficient maneuvering flexibility. Additionally, the **SCMPC and CMPC** approaches retain an explicit separation between traffic prediction and EV control.

Accurate future information about TVs is essential for the EV to react safely under the MPC framework, making interaction modeling a central research challenge. To address

the question of *How can interactions between the EV and TVs be modeled and predicted under the proposed control framework*, different approaches are employed in the two categories of frameworks. In the **SCMPC and CMPC** frameworks, interactions are captured through interaction-aware traffic prediction that explicitly represents multiple plausible interaction patterns, together with an additional uncertainty adjustment module to ensure collision-free predictions. This design allows the controller to account for uncertainties in surrounding vehicle behavior while maintaining a simple and computationally efficient interaction modeling structure. The **fully interaction-aware MPC** framework captures mode-conditioned interactions through the optimization process, using the predicted lateral configurations of vehicles to determine their coupled longitudinal behavior. This leads to a tighter integration of interaction modeling and control. However, its reliance on lane-change interaction models may limit fidelity in highly complex traffic scenarios or unstructured environment.

With the predicted information of TVs considering their interactions, solving the corresponding optimization problems addresses the questions of *How can decision-making be realized for the EV under the proposed control framework*. Specifically, multiple control modes are designed to represent the EV's possible lateral motion behaviors, such as lane-keeping and lane-changing. The decision-making module operates by evaluating the cost associated with each control mode and selecting the control actions from the least costly one. This decision-making principle simplifies the controller design by evaluating distinct control modes instead of solving a single optimization problem that incorporates lateral uncertainty, thereby facilitating more tractable and efficient lateral decision-making. Moreover, in both the **SCMPC and CMPC** frameworks, the generated scenarios are treated without explicit probability weighting across different control modes, rather than being treated probabilistically in a mode-dependent manner. As a result, the available probability information is not fully exploited, which may lead to conservative behavior compared to stochastic control approaches that explicitly incorporate scenario probabilities, or methods that condition predictions on the selected control mode. In contrast, within the proposed **fully interaction-aware MPC** framework, the predicted scenarios are distributed across different control modes based on vehicle interactions, allowing for decision-making that is conditioned on inter-vehicle interactions. Additionally, both the **MPC with interaction-aware predictions** and the **fully interaction-aware MPC** approaches are scalable with respect to the number of TVs, with a manageable increase in computational cost. In the **SCMPC and CMPC** frameworks, the computational burden increases due to the longer execution time required for vehicle state prediction as the number of TVs grows. In the **fully interaction-aware MPC** framework, the number of decision variables increases with the number of TVs, leading to higher computational demand. However, since the optimization problem remains linear, the resulting computational cost remains tractable.

Recursive feasibility guarantee of optimization problems

Guaranteeing the recursive feasibility of optimization problems under the MPC framework is a strict but essential requirement for ensuring the safety of the EV, particularly in some safety-critical scenarios. In the thesis, except for the **fully interaction-aware MPC** framework, both the **SCMPC and CMPC** frameworks incorporate safety guarantees by involving a "worst-case" scenario and introducing appropriate terminal conditions. This design methodology directly addresses the research question of *How can the feasibility of the associated optimization problem be guaranteed.* In the **SCMPC** control framework, safety is ensured by requiring the first-step control action computed from the nominal scenarios to be identical to that lower-bounded by the "worst-case" scenario. While this strategy guarantees safety, it may introduce conservatism compared to approaches that do not explicitly account for such scenario. In contrast, the **CMPC** framework does not require both first-step control actions to be identical. Instead, safety is enforced through terminal constraints that guarantee convergence to a safe set even under the "worst-case" scenario, thereby relaxing the requirement on the applied first-step input. The terminal conditions employed in recursively feasibility frameworks rely on the assumption of bounded and negligible system modeling errors, which must be carefully considered in practical implementations.

Controller performance for autonomous highway driving

The three proposed frameworks address different trade-offs among interaction modeling fidelity, safety guarantees, and computational complexity, and are therefore suitable for different application scenarios. Evaluation results across a range of representative traffic scenarios demonstrate that the proposed **SCMPC** framework enables the EV to perform safe and efficient maneuvers in complex traffic environments, while the **CMPC** framework is validated for its ability to safely control the EV. Compared to the SCMPC approach, the EV behaves less conservatively under the CMPC approach, primarily due to the larger terminal state set and the greater flexibility in control decisions. Accordingly, the **SCMPC** framework is particularly appropriate for safety-critical situations where conservative behavior is acceptable and recursive feasibility guarantees are required. The **CMPC** framework is preferable when reduced conservatism and improved driving efficiency are desired while still maintaining formal safety guarantees, making it suitable for more dynamic highway scenarios. For the proposed **fully interaction-aware MPC** framework, simulation results demonstrate that the EV consistently performs safe actions with high computational efficiency, and behaves less conservatively through possible lane changes compared to a non-interaction-aware controller. This framework is most advantageous in structured highway environments where interaction patterns can be reasonably approximated, offering the highest level of integration between planning and control with

favorable computational properties. Overall, these findings answer the research question of *What are the key performance characteristics of the proposed control approaches*. Moreover, all frameworks rely on standard onboard sensing and perception data, including positions, velocities, and lane-level information of surrounding vehicles. Such data can be obtained from automotive sensors such as Radar, LiDAR, and cameras in real implementation, or from high-fidelity simulation and traffic datasets for development and validation purposes.

7.2 Future work

Based on our investigation and exploration of traffic prediction and EV control on highways, with a focus on vehicle interactions, we recommend several directions for future research in this section.

The first suggestion for future research is to extend the application of the developed control frameworks to more complex traffic environments, such as urban areas, dense traffic conditions, and lane-merging scenarios. Applying the proposed control strategies in these settings presents promising research opportunities. Modeling interaction in such environments is more challenging due to the increased number and diversity of traffic agents, including pedestrians, various types of vehicles, and traffic signals. Future research should explore more sophisticated interaction-aware models capable of handling these diverse agents and their interactions. Interaction-aware models often incorporate personalization parameters to represent individual driving styles. A promising direction is the use of learning-based techniques to estimate or identify such parameters from collected vehicle data, allowing for more personalized and accurate motion prediction. Another important research direction is the integration of traffic prediction and control frameworks. As proposed in this thesis, integrating TVs' motion states in the MPC framework enables simultaneous decision-making for both the EV and TVs through solving optimization problems, and the interactions between vehicles are dynamically determined accordingly. In this context, modeling interactions efficiently when accounting for both longitudinal and lateral uncertainties within an MPC structure is critical and presents an interesting research direction.

A second suggestion is to establish systematic evaluation metrics for assessing the performance of the proposed controllers. Although a variety of MPC-based control approaches have been developed, most are evaluated using Monte Carlo simulations, real-world traffic data, or standard traffic simulation platforms. Generalized evaluation standards for benchmarking and comparing the developed control algorithms are still lacking. Developing such performance metrics is challenging due to the difficulty of standardizing and quantifying driving objectives across various traffic scenarios. Investigating and defining evaluation standards across different traffic contexts represents a valuable

research direction for objectively comparing controller performance.

The third suggestion is to develop safety-guaranteed control actions for safety-critical scenarios. A "worst-case" scenario is incorporated in the developed SCMPC and CMPC frameworks to ensure the safety of the EV. However, this design approach may lead to conservative control actions. Since different predicted scenarios can be represented with probabilities, an important research direction is to incorporate prediction probability into the MPC formulation, while still guaranteeing the recursive feasibility of the optimization problem. Addressing this challenge could enable the design of robust yet less conservative control strategies.

Last but not least, it is important to evaluate the robustness and effectiveness of the proposed controllers in more practical situations where noise and delays are present in the control loop. In this thesis, we use IPG CarMaker as the simulation environment to test the proposed controllers. However, in this setting, the vehicle dynamics of the TVs are simplified, and motion prediction errors due to modeling inaccuracies are negligible. In real-world applications, data from on-board sensors are subject to noise, and delays are present in vehicle dynamics. Therefore, evaluating the performance of the proposed controllers under these conditions within the MPC frameworks is of particular interest for assessing their robustness and practical effectiveness. Such issues may compromise both the accuracy of traffic prediction and the performance of the controller. Consequently, improving control strategies to better handle inaccuracies and delays in the control loop is a worthwhile direction for future research.

REFERENCES

[1] J. Zhao, W. Zhao, B. Deng, Z. Wang, F. Zhang, W. Zheng, W. Cao, J. Nan, Y. Lian, and A. F. Burke, "Autonomous driving system: A comprehensive survey," *Expert Systems with Applications*, vol. 242, p. 122836, 2024.

[2] L. Hua, J. Antona-Makoshi, and L. Neurauter, "Status and challenges of Level 3 automated driving systems," 2025.

[3] F. Borrelli, A. Bemporad, and M. Morari, *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

[4] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A survey of autonomous driving: Common practices and emerging technologies," *IEEE access*, vol. 8, pp. 58 443–58 469, 2020.

[5] K. Bimbraw, "Autonomous cars: Past, present and future a review of the developments in the last century, the present scenario and the expected future of autonomous vehicle technology," in *2015 IEEE 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, vol. 1. IEEE, 2015, pp. 191–198.

[6] L. Chen, Y. Li, C. Huang, B. Li, Y. Xing, D. Tian, L. Li, Z. Hu, X. Na, Z. Li *et al.*, "Milestones in autonomous driving and intelligent vehicles: Survey of surveys," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 2, pp. 1046–1056, 2022.

[7] L. Claussmann, M. Revilloud, D. Gruyer, and S. Glaser, "A review of motion planning for highway autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 5, pp. 1826–1848, 2019.

[8] W. Payre, J. Cestac, and P. Delhomme, "Intention to use a fully automated car: Attitudes and a priori acceptability," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 27, pp. 252–263, 2014.

[9] F. H. A. U.S. Department of Transportation, "Highway statistics 2020," Federal Highway Administration, Tech. Rep., 2021.

[10] W. Wang, L. Wang, C. Zhang, C. Liu, and L. Sun, "Social interactions for autonomous driving: A review and perspectives," *Foundations and Trends® in Robotics*, vol. 10, no. 3-4, pp. 198–376, 2022.

[11] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH Journal*, vol. 1, no. 1, pp. 1–14, 2014.

[12] P. Karle, M. Geisslinger, J. Betz, and M. Lienkamp, "Scenario understanding and motion prediction for autonomous vehicles—Review and comparison," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 16 962–16 982, 2022.

[13] D. E. Benrachou, S. Glaser, M. Elhenawy, and A. Rakotonirainy, "Use of social interaction and intention to improve motion prediction within automated vehicle framework: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 22 807–22 837, 2022.

[14] T. Brüdigam, M. Olbrich, D. Wollherr, and M. Leibold, "Stochastic Model Predictive Control with a safety guarantee for automated driving," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 22–36, 2021.

[15] M. Shi, H. He, J. Li, M. Han, X. Zhao, and D. Gillet, "Less conservatism, stronger robustness: Iterative robust gain-scheduled path following control of autonomous bus with unstructured and changing dynamics," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 5, pp. 6116–6128, 2024.

[16] Society of Automotive Engineers, "SAE J3016: Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles," SAE International, Tech. Rep., 2014.

[17] B. Lantos and L. Márton, *Nonlinear control of vehicles and robots*. Springer Science and Business Media, 2010.

[18] E. Marti, M. A. De Miguel, F. Garcia, and J. Perez, "A review of sensor technologies for perception in automated driving," *IEEE Intelligent Transportation Systems Magazine*, vol. 11, no. 4, pp. 94–108, 2019.

[19] A. Roland and P. Shiman, *Strategic computing: DARPA and the quest for machine intelligence, 1983-1993*. MIT Press, 2002.

[20] C. Thorpe, M. H. Hebert, T. Kanade, and S. A. Shafer, "Vision and navigation for the Carnegie-Mellon Navlab," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 3, pp. 362–373, 1988.

[21] M. Daily, J. Harris, D. Keirsey, D. Olin, D. Payton, K. Reiser, J. Rosenblatt, D. Tseng, and V. Wong, "Autonomous cross-country navigation with the ALV," in *1988 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1988, pp. 718–726.

[22] V. Patkar and N. Mehendale, "Autonomous ground vehicles: Technological advancements, implementation challenges, and future directions," *International Journal of Intelligent Robotics and Applications*, pp. 1–38, 2025.

[23] P. Ioannou, *Automated highway systems*. Springer Science and Business Media, 1997.

[24] R. Behringer, S. Sundareswaran, B. Gregory, R. Elsley, B. Addison, W. Guthmiller, R. Daily, and D. Bevly, "The DARPA grand challenge-development of an autonomous vehicle," in *IEEE Intelligent Vehicles Symposium, 2004*. IEEE, 2004, pp. 226–231.

[25] U. Ozguner, C. Stiller, and K. Redmill, "Systems for safety and autonomous behavior in cars: The DARPA grand challenge experience," *Proceedings of the IEEE*, vol. 95, no. 2, pp. 397–412, 2007.

[26] M. Buehler, K. Iagnemma, and S. Singh, *The 2005 DARPA grand challenge: The great robot race*. Springer Science and Business Media, 2007, vol. 36.

[27] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA urban challenge: Autonomous vehicles in city traffic*. Springer Science and Business Media, 2009, vol. 56.

[28] M. Lenox and J. McDermott, "Driving Waymo's fully autonomous future," Darden Business Publishing, Tech. Rep. UVA-S-0367, 2021.

[29] A. Broggi, M. Buzzoni, S. Debattisti, P. Grisleri, M. C. Laghi, P. Medici, and P. Versari, "Extensive tests of autonomous driving technologies," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1403–1415, 2013.

[30] A. Broggi, P. Cerri, S. Debattisti, M. C. Laghi, P. Medici, M. Panciroli, and A. Prioletti, "Proud-public road urban driverless test: Architecture and results," in *2014 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2014, pp. 648–654.

[31] J. Ziegler, P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. G. Keller *et al.*, "Making Bertha drive—An autonomous journey on a historic route," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 2, pp. 8–20, 2014.

[32] Y. Freemark, C. Stacy, O. Fiol, J. Morales-Burnett, and S. Weng, "Regulations to respond to the potential benefits and perils of self-driving cars," Urban Institute, Tech. Rep., 2022.

[33] L. Dixon, "Autonowashing: The greenwashing of vehicle automation," *Transportation Research Interdisciplinary Perspectives*, vol. 5, pp. 100–113, 2020.

[34] U. Valente, "Development of autonomous driving in electric vehicles: Analysis of quality engineering and future applicability," Ph.D. dissertation, Politecnico di Torino, 2024.

[35] C. McCarroll and F. Cugurullo, "No city on the horizon: Autonomous cars, artificial intelligence, and the absence of urbanism," *Frontiers in Sustainable Cities*, vol. 4, p. 937933, 2022.

[36] S. Singh and B. S. Saini, "Autonomous cars: Recent developments, challenges, and possible solutions," in *IOP conference series: Materials science and engineering*, vol. 1022, no. 1. IOP Publishing, 2021, p. 012028.

[37] L. Chen, Y. Li, C. Huang, Y. Xing, D. Tian, L. Li, Z. Hu, S. Teng, C. Lv, J. Wang *et al.*, "Milestones in autonomous driving and intelligent vehicles—Part I: Control, computing system design, communication, HD map, testing, and human behaviors," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 9, pp. 5831–5847, 2023.

[38] J. Nidamanuri, C. Nibhanupudi, R. Assfalg, and H. Venkataraman, "A progressive review: Emerging technologies for ADAS driven solutions," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 2, pp. 326–341, 2021.

[39] R. Okuda, Y. Kajiwara, and K. Terashima, "A survey of technical trend of ADAS and autonomous driving," in *2014 IEEE International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*. IEEE, 2014, pp. 1–4.

[40] A. Swief and M. El-Habrouk, "A survey of automotive driving assistance systems technologies," in *2018 IEEE International Conference on Artificial Intelligence and Data Processing (IDAP)*. IEEE, 2018, pp. 1–12.

[41] A. Bucchiarone, S. Battisti, A. Marconi, R. Maldacea, and D. C. Ponce, "Autonomous shuttle-as-a-service (ASaaS): Challenges, opportunities, and social implications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3790–3799, 2020.

[42] C. Iclodean, N. Cordos, and B. O. Varga, "Autonomous shuttle bus for public transportation: A review," *Energies*, vol. 13, no. 11, p. 2917, 2020.

[43] C. Hubmann, M. Becker, D. Althoff, D. Lenz, and C. Stiller, "Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles," in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 1671–1678.

[44] J.-F. Bonnefon, A. Shariff, and I. Rahwan, "The social dilemma of autonomous vehicles," *Science*, vol. 352, no. 6293, pp. 1573–1576, 2016.

[45] A. Chougule, V. Chamola, A. Sam, F. R. Yu, and B. Sikdar, "A comprehensive review on limitations of autonomous driving and its impact on accidents and collisions," *IEEE Open Journal of Vehicular Technology*, vol. 5, pp. 142–161, 2023.

[46] C. Gao, G. Wang, W. Shi, Z. Wang, and Y. Chen, "Autonomous driving security: State of the art and challenges," *IEEE Internet of Things Journal*, vol. 9, no. 10, pp. 7572–7595, 2021.

[47] A. Giannaros, A. Karras, L. Theodorakopoulos, C. Karras, P. Kranias, N. Schizas, G. Kalogeratos, and D. Tsolis, "Autonomous vehicles: Sophisticated attacks, safety issues, challenges, open topics, blockchain, and future directions," *Journal of Cybersecurity and Privacy*, vol. 3, no. 3, pp. 493–543, 2023.

[48] D. J. Yeong, G. Velasco-Hernandez, J. Barry, and J. Walsh, "Sensor and sensor fusion technology in autonomous vehicles: A review," *Sensors*, vol. 21, no. 6, p. 2140, 2021.

[49] L. Liu, S. Lu, R. Zhong, B. Wu, Y. Yao, Q. Zhang, and W. Shi, "Computing systems for autonomous driving: State of the art and challenges," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6469–6486, 2020.

[50] H. A. Ignatious, M. Khan *et al.*, "An overview of sensors in autonomous vehicles," *Procedia Computer Science*, vol. 198, pp. 736–741, 2022.

[51] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1–19, 2015.

[52] J. Van Brummelen, M. O'brien, D. Gruyer, and H. Najjaran, "Autonomous vehicle perception: The technology of today and tomorrow," *Transportation research part C: emerging technologies*, vol. 89, pp. 384–406, 2018.

[53] S. Kuutti, S. Fallah, K. Katsaros, M. Dianati, F. Mccullough, and A. Mouzakitis, "A survey of the state of the art localization techniques and their potentials for autonomous vehicle applications," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 829–846, 2018.

[54] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. 187–210, 2018.

[55] S. Dixit, S. Fallah, U. Montanaro, M. Dianati, A. Stevens, F. Mccullough, and A. Mouzakitis, "Trajectory planning and tracking for autonomous overtaking: State of the art and future prospects," *Annual Reviews in Control*, vol. 45, pp. 76–86, 2018.

[56] T. T. Mac, C. Copot, D. T. Tran, and R. De Keyser, "Heuristic approaches in robot path planning: A survey," *Robotics and Autonomous Systems*, vol. 86, pp. 13–28, 2016.

[57] X. Di and R. Shi, "A survey on autonomous vehicle control in the era of mixed-autonomy: From physics-based to AI-guided driving policy learning," *Transportation Research Part C: Emerging Technologies*, vol. 125, p. 103008, 2021.

[58] S. Kuutti, R. Bowden, Y. Jin, P. Barber, and S. Fallah, "A survey of deep learning applications to autonomous vehicle control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 712–733, 2020.

[59] 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*, vol. 1, no. 1, pp. 33–55, 2016.

[60] J. Fayyad, M. A. Jaradat, D. Gruyer, and H. Najjaran, "Deep learning sensor fusion for autonomous vehicle perception and localization: A review," *Sensors*, vol. 20, no. 15, p. 4220, 2020.

[61] D. Fernandes, A. Silva, R. Névoa, C. Simões, D. Gonzalez, M. Guevara, P. Novais, J. Monteiro, and P. Melo-Pinto, "Point-cloud based 3D object detection and classification methods for self-driving applications: A survey and taxonomy," *Information Fusion*, vol. 68, pp. 161–191, 2021.

[62] A. Gupta and X. Fernando, "Simultaneous localization and mapping (SLAM) and data fusion in unmanned aerial vehicles: Recent advances and challenges," *Drones*, vol. 6, no. 4, p. 85, 2022.

[63] S. A. Bello, S. Yu, C. Wang, J. M. Adam, and J. Li, "Deep learning on 3D point clouds," *Remote Sensing*, vol. 12, no. 11, p. 1729, 2020.

[64] Y. Zhang, A. Carballo, H. Yang, and K. Takeda, "Perception and sensing for autonomous vehicles under adverse weather conditions: A survey," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 196, pp. 146–177, 2023.

[65] J. Vargas, S. Alsweiss, O. Toker, R. Razdan, and J. Santos, "An overview of autonomous vehicles sensors and their vulnerability to weather conditions," *Sensors*, vol. 21, no. 16, p. 5397, 2021.

[66] S. Sun, A. P. Petropulu, and H. V. Poor, "MIMO radar for advanced driver-assistance systems and autonomous driving: Advantages and challenges," *IEEE Signal Processing Magazine*, vol. 37, no. 4, pp. 98–117, 2020.

[67] S. M. Veres, L. Molnar, N. K. Lincoln, and C. P. Morice, "Autonomous vehicle control systems—A review of decision making," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 225, no. 2, pp. 155–195, 2011.

[68] D. Delling, P. Sanders, D. Schultes, and D. Wagner, "Engineering route planning algorithms," in *Algorithmics of Large and Complex Networks: Design, Analysis, and Simulation*. Springer, 2009, pp. 117–139.

[69] L. Claussmann, A. Carvalho, and G. Schildbach, "A path planner for autonomous driving on highways using a human mimicry approach with binary decision diagrams," in *2015 European Control Conference (ECC)*. IEEE, 2015, pp. 2976–2982.

[70] X. Li, Z. Sun, Q. Zhu, and D. Liu, "A unified approach to local trajectory planning and control for autonomous driving along a reference path," in *2014 IEEE International Conference on Mechatronics and Automation (ICMA)*. IEEE, 2014, pp. 1716–1721.

[71] T. Zhang, W. Song, M. Fu, Y. Yang, X. Tian, and M. Wang, "A unified framework integrating decision making and trajectory planning based on spatio-temporal voxels for highway autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 10 365–10 379, 2021.

[72] K. Daniel, A. Nash, S. Koenig, and A. Felner, "Theta*: Any-angle path planning on grids," *Journal of Artificial Intelligence Research*, vol. 39, pp. 533–579, 2010.

[73] D. Connell and H. M. La, "Dynamic path planning and replanning for mobile robots using RRT," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2017, pp. 1429–1434.

[74] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller, "Anytime motion planning using the RRT," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 1478–1483.

[75] M. R. Bachute and J. M. Subhedar, "Autonomous driving architectures: Insights of machine learning and deep learning algorithms," *Machine Learning with Applications*, vol. 6, p. 100164, 2021.

[76] M. Sheckells, T. M. Caldwell, and M. Kobilarov, "Fast approximate path coordinate motion primitives for autonomous driving," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 837–842.

[77] F. Von Hundelshausen, M. Himmelsbach, F. Hecker, A. Mueller, and H.-J. Wuensche, "Driving with tentacles: Integral structures for sensing and motion," *Journal of Field Robotics*, vol. 25, no. 9, pp. 640–673, 2008.

[78] M. Ammour, R. Orjuela, and M. Basset, "A MPC combined decision making and trajectory planning for autonomous vehicle collision avoidance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24 805–24 817, 2022.

[79] S. Zhu and B. Aksun-Guvenc, "Trajectory planning of autonomous vehicles based on parameterized control optimization in dynamic on-road environments," *Journal of Intelligent and Robotic Systems*, vol. 100, no. 3, pp. 1055–1067, 2020.

[80] J. Wit, C. D. Crane III, and D. Armstrong, "Autonomous ground vehicle path tracking," *Journal of Robotic Systems*, vol. 21, no. 8, pp. 439–449, 2004.

[81] O. Amidi and C. E. Thorpe, "Integrated mobile robot control," in *Mobile Robots V*, vol. 1388. SPIE, 1991, pp. 504–523.

[82] Á. Domina and V. Tihanyi, "Comparison of path following controllers for autonomous vehicles," in *2019 IEEE 17th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*. IEEE, 2019, pp. 147–152.

[83] P. Stano, U. Montanaro, D. Tavernini, M. Tufo, G. Fiengo, L. Novella, and A. Sorniotti, "Model predictive path tracking control for automated road vehicles: A review," *Annual Reviews in Control*, vol. 55, pp. 194–236, 2023.

[84] T. M. Vu, R. Moezzi, J. Cyrus, and J. Hlava, "Model Predictive Control for autonomous driving vehicles," *Electronics*, vol. 10, no. 21, p. 2593, 2021.

[85] F. Mohseni, E. Frisk, and L. Nielsen, "Distributed cooperative MPC for autonomous driving in different traffic scenarios," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 2, pp. 299–309, 2020.

[86] R. Zhao, Y. Li, Y. Fan, F. Gao, M. Tsukada, and Z. Gao, "A survey on recent advancements in autonomous driving using deep reinforcement learning: Applications, challenges, and solutions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 12, pp. 19 365–19 398, 2024.

[87] L. Ullrich, M. Buchholz, K. Dietmayer, and K. Graichen, "AI safety assurance for automated vehicles: A survey on research, standardization, regulation," *IEEE Transactions on Intelligent Vehicles*, pp. 1–19, 2024.

[88] A. Kuznietsov, B. Gyevnar, C. Wang, S. Peters, and S. V. Albrecht, "Explainable AI for safe and trustworthy autonomous driving: A systematic review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 12, pp. 19 342–19 364, 2024.

[89] Y. Xing, C. Lv, H. Wang, H. Wang, Y. Ai, D. Cao, E. Velenis, and F.-Y. Wang, "Driver lane change intention inference for intelligent vehicles: Framework, survey, and challenges," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4377–4390, 2019.

[90] K. Li, X. Wang, Y. Xu, and J. Wang, "Lane changing intention recognition based on speech recognition models," *Transportation Research Part C: Emerging Technologies*, vol. 69, pp. 497–514, 2016.

[91] J. Wang, Z. Zhang, and G. Lu, "A Bayesian inference based adaptive lane change prediction model," *Transportation Research Part C: Emerging Technologies*, vol. 132, p. 103363, 2021.

[92] X. Li, W. Wang, and M. Roetting, "Estimating driver's lane-change intent considering driving style and contextual traffic," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3258–3271, 2019.

[93] Y. Xia, Z. Qu, Z. Sun, and Z. Li, "A human-like model to understand surrounding vehicles' lane changing intentions for autonomous driving," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 5, pp. 4178–4189, 2021.

[94] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013, pp. 797–802.

[95] B. Morris, A. Doshi, and M. Trivedi, "Lane change intent prediction for driver assistance: On-road design and evaluation," in *2011 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2011, pp. 895–901.

[96] L. Li, W. Zhao, C. Xu, C. Wang, Q. Chen, and S. Dai, "Lane-change intention inference based on RNN for autonomous driving on highways," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 5499–5510, 2021.

[97] Y. Xing, C. Lv, H. Wang, D. Cao, and E. Velenis, "An ensemble deep learning approach for driver lane change intention inference," *Transportation Research Part C: Emerging Technologies*, vol. 115, p. 102615, 2020.

[98] W. Wang, T. Qie, C. Yang, W. Liu, C. Xiang, and K. Huang, "An intelligent lane-changing behavior prediction and decision-making strategy for an autonomous vehicle," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 3, pp. 2927–2937, 2022.

[99] D. D. Salvucci and A. Liu, "The time course of a lane change: Driver control and eye-movement behavior," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 5, no. 2, pp. 123–132, 2002.

[100] D. D. Salvucci, "Modeling driver behavior in a cognitive architecture," *Human Factors*, vol. 48, no. 2, pp. 362–380, 2006.

[101] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakitis, "Deep learning-based vehicle behavior prediction for autonomous driving applications: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 33–47, 2020.

[102] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTMs," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 1179–1184.

[103] S. Hoermann, M. Bach, and K. Dietmayer, "Dynamic occupancy grid prediction for urban autonomous driving: A deep learning approach with fully automatic labeling," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2056–2063.

[104] A. Zyner, S. Worrall, and E. Nebot, "A recurrent neural network solution for predicting driver intention at unsignalized intersections," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1759–1764, 2018.

[105] L. Xin, P. Wang, C.-Y. Chan, J. Chen, S. E. Li, and B. Cheng, "Intention-aware long horizon trajectory prediction of surrounding vehicles using dual LSTM networks," in *2018 IEEE 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 1441–1446.

[106] W. Luo, B. Yang, and R. Urtasun, "Fast and furious: Real time end-to-end 3D detection, tracking and motion forecasting with a single convolutional net," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 3569–3577.

[107] S. Casas, W. Luo, and R. Urtasun, "Intentnet: Learning to predict intention from raw sensor data," in *Conference on Robot Learning (CoRL)*. PMLR, 2018, pp. 947–956.

[108] Z. Ding and H. Zhao, "Incorporating driving knowledge in deep learning based vehicle trajectory prediction: A survey," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 8, pp. 3996–4015, 2023.

[109] J. Fang, F. Wang, J. Xue, and T.-S. Chua, "Behavioral intention prediction in driving scenes: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 8, pp. 8334–8355, 2024.

[110] W. Lan, D. Li, Q. Hao, D. Zhao, and B. Tian, "Implicit scene context-aware interactive trajectory prediction for autonomous driving," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 9, pp. 5461–5477, 2023.

[111] P. Xu, J.-B. Hayet, and I. Karamouzas, "Context-aware timewise VAEs for real-time vehicle trajectory prediction," *IEEE Robotics and Automation Letters*, vol. 8, no. 9, pp. 5440–5447, 2023.

[112] M. M. Sánchez, D. Pogosov, E. Silvas, D. C. Mocanu, J. Elfring, and R. Van De Molengraft, "Situation-aware drivable space estimation for automated driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9615–9629, 2022.

[113] Y. Chen, S. Veer, P. Karkus, and M. Pavone, "Interactive joint planning for autonomous vehicles," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 987–994, 2023.

[114] T. Lew, R. Bonalli, and M. Pavone, "Risk-averse trajectory optimization via sample average approximation," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1500–1507, 2023.

[115] X. Zhang, S. Zeinali, and G. Schildbach, "Interaction-aware traffic prediction and Scenario-based Model Predictive Control for autonomous vehicles on highways," in *2024 European Control Conference (ECC)*. IEEE, 2024, pp. 3351–3357.

[116] Z. Wang, Y. Shi, W. Tong, Z. Gu, and Q. Cheng, "Car-following models for human-driven vehicles and autonomous vehicles: A systematic review," *Journal of Transportation Engineering, Part A: Systems*, vol. 149, no. 8, p. 04023075, 2023.

[117] S. Albeaik, A. Bayen, M. T. Chiri, X. Gong, A. Hayat, N. Kardous, A. Keimer, S. T. McQuade, B. Piccoli, and Y. You, "Limitations and improvements of the intelligent driver model (IDM)," *SIAM Journal on Applied Dynamical Systems*, vol. 21, no. 3, pp. 1862–1892, 2022.

[118] H. Yu, R. Jiang, Z. He, Z. Zheng, L. Li, R. Liu, and X. Chen, "Automated vehicle-involved traffic flow studies: A survey of assumptions, models, speculations, and perspectives," *Transportation Research Part C: Emerging Technologies*, vol. 127, p. 103101, 2021.

[119] Z. Zheng, "Recent developments and research needs in modeling lane changing," *Transportation Research Part B: Methodological*, vol. 60, pp. 16–32, 2014.

[120] A. Kesting, M. Treiber, and D. Helbing, "General lane-changing model MOBIL for car-following models," *Transportation Research Record*, vol. 1999, no. 1, pp. 86–94, 2007.

[121] Z. Zhu and H. Zhao, "A survey of deep RL and IL for autonomous driving policy learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14 043–14 065, 2021.

[122] V. Lefkopoulos, M. Menner, A. Domahidi, and M. N. Zeilinger, "Interaction-aware motion prediction for autonomous driving: A multiple model Kalman filtering scheme," *IEEE Robotics and Automation Letters*, vol. 6, no. 1, pp. 80–87, 2020.

[123] J. Zhou, B. Olofsson, and E. Frisk, "Interaction-aware moving target Model Predictive Control for autonomous vehicles motion planning," in *2022 European Control Conference (ECC)*. IEEE, 2022, pp. 154–161.

[124] Y. Yoon and K. Yi, "Trajectory prediction using graph-based deep learning for longitudinal control of autonomous vehicles: A proactive approach for autonomous driving in urban dynamic traffic environments," *IEEE Vehicular Technology Magazine*, vol. 17, no. 4, pp. 18–27, 2022.

[125] K. Cho, T. Ha, G. Lee, and S. Oh, "Deep predictive autonomous driving using multi-agent joint trajectory prediction and traffic rules," in *2019 IEEE International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 2076–2081.

[126] P. Hang, C. Lv, Y. Xing, C. Huang, and Z. Hu, "Human-like decision making for autonomous driving: A noncooperative game theoretic approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2076–2087, 2020.

[127] N. Li, D. W. Oyler, M. Zhang, Y. Yildiz, I. Kolmanovsky, and A. R. Girard, "Game theoretic modeling of driver and vehicle interactions for verification and validation of autonomous vehicle control systems," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 5, pp. 1782–1797, 2017.

[128] M. Bahram, A. Lawitzky, J. Friedrichs, M. Aeberhard, and D. Wollherr, "A game-theoretic approach to replanning-aware interactive scene prediction and planning," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3981–3992, 2015.

[129] R. Kensbock, M. Nezami, and G. Schildbach, "Scenario-based decision-making, planning and control for interaction-aware autonomous driving on highways," in *2023 IEEE Intelligent Vehicles Symposium (IV)*, 2023, pp. 1–6.

[130] F. Altché and A. de La Fortelle, "An LSTM network for highway trajectory prediction," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 353–359.

[131] S. Rahmani, A. Baghbani, N. Bouguila, and Z. Patterson, "Graph neural networks for intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8846–8885, 2023.

[132] D. Lee, Y. Gu, J. Hoang, and M. Marchetti-Bowick, "Joint interaction and trajectory prediction for autonomous driving using graph neural networks," *arXiv preprint arXiv:1912.07882*, 2019.

[133] M. Farina, L. Giulioni, and R. Scattolini, "Stochastic linear Model Predictive Control with chance constraints-A review," *Journal of Process Control*, vol. 44, pp. 53–67, 2016.

[134] A. Mesbah, "Stochastic Model Predictive Control: An overview and perspectives for future research," *IEEE Control Systems Magazine*, vol. 36, no. 6, pp. 30–44, 2016.

[135] S. Vaskov, R. Quirynen, M. Menner, and K. Berntorp, "Friction-adaptive stochastic nonlinear Model Predictive Control for autonomous vehicles," *Vehicle system dynamics*, vol. 62, no. 2, pp. 347–371, 2024.

[136] G. Schildbach, L. Fagiano, C. Frei, and M. Morari, "The scenario approach for stochastic Model Predictive Control with bounds on closed-loop constraint violations," *Automatica*, vol. 50, no. 12, pp. 3009–3018, 2014.

[137] G. Schildbach and M. Morari, "Scenario MPC for linear time-varying systems with individual chance constraints," in *2015 American Control Conference (ACC)*. IEEE, 2015, pp. 415–421.

[138] S. H. Nair, V. Govindarajan, T. Lin, C. Meissen, H. E. Tseng, and F. Borrelli, "Stochastic MPC with multi-modal predictions for traffic intersections," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2022, pp. 635–640.

[139] S. H. Nair, V. Govindarajan, T. Lin, Y. Wang, E. H. Tseng, and F. Borrelli, "Stochastic MPC with dual control for autonomous driving with multi-modal interaction-aware predictions," *arXiv preprint arXiv:2208.03525*, 2022.

[140] Y. Chen, U. Rosolia, W. Ubellacker, N. Csomay-Shanklin, and A. D. Ames, "Interactive multi-modal motion planning with branch Model Predictive Control," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5365–5372, 2022.

[141] B. Ivanovic, A. Elhafsi, G. Rosman, A. Gaidon, and M. Pavone, "MATS: An interpretable trajectory forecasting representation for planning and control," in *Conference on Robot Learning (CoRL)*. PMLR, 2021, pp. 2243–2256.

[142] T. A. Dingus, S. G. Klauer, V. L. Neale, A. Petersen, S. E. Lee, J. Sudweeks, M. A. Perez, J. Hankey, D. Ramsey, S. Gupta *et al.*, "The 100-car naturalistic driving study, phase II-results of the 100-car field experiment," United States. Department of Transportation. National Highway Traffic Safety Administration, Tech. Rep., 2006.

[143] M. A. Perez, J. D. Sudweeks, E. Sears, J. Antin, S. Lee, J. M. Hankey, and T. A. Dingus, "Performance of basic kinematic thresholds in the identification of crash and near-crash events within naturalistic driving data," *Accident Analysis and Prevention*, vol. 103, pp. 10–19, 2017.

[144] Y. He, B. Ciuffo, Q. Zhou, M. Makridis, K. Mattas, J. Li, Z. Li, F. Yan, and H. Xu, "Adaptive cruise control strategies implemented on experimental vehicles: A review," *IFAC-PapersOnLine*, vol. 52, no. 5, pp. 21–27, 2019.

[145] S. Magdici and M. Althoff, "Adaptive cruise control with safety guarantees for autonomous vehicles," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 5774–5781, 2017.

[146] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," *arXiv preprint arXiv:1708.06374*, 2017.

[147] X. Xu, X. Wang, X. Wu, O. Hassanin, and C. Chai, "Calibration and evaluation of the Responsibility-Sensitive Safety model of autonomous car-following maneuvers

using naturalistic driving study data," *Transportation Research Part C: Emerging Technologies*, vol. 123, p. 102988, 2021.

[148] J. P. Alsterda and J. C. Gerdes, "Contingency Model Predictive Control for linear time-varying systems," *arXiv preprint arXiv:2102.12045*, 2021.

[149] Y. Chen, U. Rosolia, C. Fan, A. Ames, and R. Murray, "Reactive motion planning with probabilistic safety guarantees," in *Conference on Robot Learning (CoRL)*. PMLR, 2021, pp. 1958–1970.

[150] X. Zhang, S. Zeinali, and G. Schildbach, "Interaction-aware traffic prediction and Scenario-based Model Predictive Control for autonomous vehicles on highways," *IEEE Transactions on Control Systems Technology*, vol. 33, no. 4, pp. 1235–1245, 2025.

[151] D. Q. Mayne, "Model Predictive Control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.

[152] M. B. Saltık, L. Özkan, J. H. Ludlage, S. Weiland, and P. M. Van den Hof, "An outlook on robust Model Predictive Control algorithms: Reflections on performance and computational aspects," *Journal of Process Control*, vol. 61, pp. 77–102, 2018.

[153] S. V. Raković, "Model Predictive Control: Classical, robust, and stochastic," *IEEE Control Systems Magazine*, vol. 36, no. 6, pp. 102–105, 2016.

[154] A. Gazar, M. Khadiv, A. Del Prete, and L. Righetti, "Stochastic and Robust MPC for bipedal locomotion: A comparative study on robustness and performance," in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2021, pp. 61–68.

[155] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained Model Predictive Control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[156] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, computation, and design*. Nob Hill Publishing, 2017.

[157] H. Borhan, A. Vahidi, A. M. Phillips, M. L. Kuang, I. V. Kolmanovsky, and S. Di Cairano, "MPC-based energy management of a power-split hybrid electric vehicle," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 3, pp. 593–603, 2012.

[158] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 566–580, 2007.

[159] D. Q. Mayne, M. M. Seron, S. V. Raković *et al.*, "Robust Model Predictive Control: A survey," *Automatica*, vol. 41, no. 12, pp. 2199–2211, 2005.

[160] M. Schuurmans, A. Katriniok, H. E. Tseng, and P. Patrinos, "Learning-based risk-averse Model Predictive Control for Adaptive Cruise Control with stochastic driver models," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 15 128–15 133, 2020.

[161] M. Schuurmans, A. Katriniok, C. Meissen, H. E. Tseng, and P. Patrinos, "Safe, learning-based MPC for highway driving under lane-change uncertainty: A distributionally robust approach," *Artificial Intelligence*, vol. 320, p. 103920, 2023.

[162] R. Soloperto, J. Köhler, F. Allgöwer, and M. A. Müller, "Collision avoidance for uncertain nonlinear systems with moving obstacles using robust Model Predictive Control," in *2019 European Control Conference (ECC)*. IEEE, 2019, pp. 811–817.

[163] T. Brüdigam, R. Jacumet, D. Wollherr, and M. Leibold, "Safe stochastic Model Predictive Control," in *2022 IEEE 61st Annual Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 1796–1802.

[164] G. Cesari, G. Schildbach, A. Carvalho, and F. Borrelli, "Scenario Model Predictive Control for lane change assistance and autonomous driving on highways," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 3, pp. 23–35, 2017.

[165] T. Brüdigam, M. Olbrich, M. Leibold, and D. Wollherr, "Combining stochastic and Scenario Model Predictive Control to handle target vehicle uncertainty in an autonomous driving highway scenario," in *2018 IEEE 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 1317–1324.

[166] G. Schildbach and F. Borrelli, "Scenario Model Predictive Control for lane change assistance on highways," in *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 611–616.

[167] X. Li, X. Ying, and M. C. Chuah, "Grip: Graph-based interaction-aware trajectory prediction," in *2019 IEEE 22nd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2019, pp. 3960–3966.

[168] I. Batkovic, U. Rosolia, M. Zanon, and P. Falcone, "A robust Scenario MPC approach for uncertain multi-modal obstacles," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 947–952, 2021.

[169] J. P. Alsterda, M. Brown, and J. C. Gerdes, "Contingency Model Predictive Control for automated vehicles," in *2019 American Control Conference (ACC)*, 2019, pp. 717–722.

[170] J. Hardy and M. Campbell, "Contingency planning over probabilistic obstacle predictions for autonomous road vehicles," *IEEE Transactions on Robotics*, vol. 29, no. 4, pp. 913–929, 2013.

[171] L. Jung, A. Estornell, and M. Everett, "Contingency constrained planning with MPPI within MPPI," *arXiv preprint arXiv:2412.09777*, 2024.

[172] T. Li, L. Zhang, S. Liu, and S. Shen, "Marc: Multipolicy and risk-aware contingency planning for autonomous driving," *IEEE Robotics and Automation Letters*, vol. 8, no. 10, pp. 6587–6594, 2023.

[173] G. Xie, L. Sun, T. Wen, X. Hei, and F. Qian, "Adaptive transition probability matrix-based parallel IMM algorithm," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 5, pp. 2980–2989, 2019.

[174] R. Schmied, H. Waschl, and L. Del Re, "Extension and experimental validation of fuel efficient predictive adaptive cruise control," in *2015 American Control Conference (ACC)*. IEEE, 2015, pp. 4753–4758.

[175] C. Liu, S. Lee, S. Varnhagen, and H. E. Tseng, "Path planning for autonomous vehicles using Model Predictive Control," in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 174–179.

[176] D. Wang, L. Gao, Z. Lan, W. Li, J. Ren, J. Zhang, P. Zhang, P. Zhou, S. Wang, J. Pan *et al.*, "An intelligent self-driving truck system for highway transportation," *Frontiers in Neurorobotics*, vol. 16, p. 843026, 2022.

[177] J. Löfberg, "Oops! I cannot do it again: Testing for recursive feasibility in MPC," *Automatica*, vol. 48, no. 3, pp. 550–555, 2012.

[178] F. Alizadeh and D. Goldfarb, "Second-order cone programming," *Mathematical Programming*, vol. 95, no. 1, pp. 3–51, 2003.

[179] Gurobi Optimization, LLC, "Gurobi optimizer reference manual," 2024. [Online]. Available: https://www.gurobi.com.

[180] B. Evens, M. Schuurmans, and P. Patrinos, "Learning MPC for interaction-aware autonomous driving: A game-theoretic approach," in *2022 European Control Conference (ECC)*. IEEE, 2022, pp. 34–39.

[181] IPG Automotive GmbH, *CarMaker reference manual*, IPG Automotive GmbH, Karlsruhe, Germany, 2025, version 12.0. [Online]. Available: https://ipg-automotive.com.

[182] M. Treiber and A. Kesting, *Traffic flow dynamics*. Springer, 2013, vol. 1.

LIST OF PUBLICATIONS

1. X. Zhang, S. Zeinali and G. Schildbach, “Interaction-aware traffic prediction and Scenario-based Model Predictive Control for autonomous vehicles on highways,” in *2024 European Control Conference (ECC)*, IEEE, 2024, pp. 3351-3357.

2. X. Zhang, S. Zeinali and G. Schildbach, “Interaction-aware traffic prediction and Scenario-based Model Predictive Control for autonomous vehicles on highways,” *IEEE Transactions on Control Systems Technology*, vol. 33, no. 4, pp. 1235-1245, 2025.

3. X. Zhang, S. Zeinali, H. Wen, and G. Schildbach, “MOBIL-based traffic prediction and interaction-aware Model Predictive Control for autonomous highway driving,” *Control Engineering Practice*, vol. 164, pp. 106434, 2025.

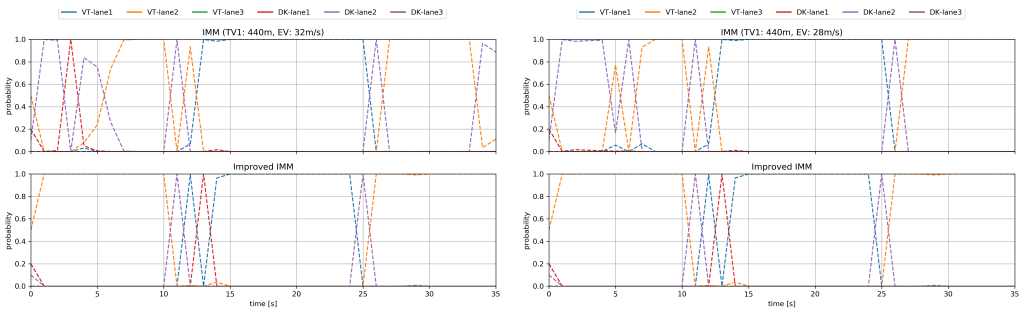
4. X. Zhang, H. Wen, H. Abbas, and G. Schildbach, “Efficient Contingency Model Predictive Control for safe and less conservative autonomous driving”. (Manuscript under revision for resubmission)

Appendices

Appendix A

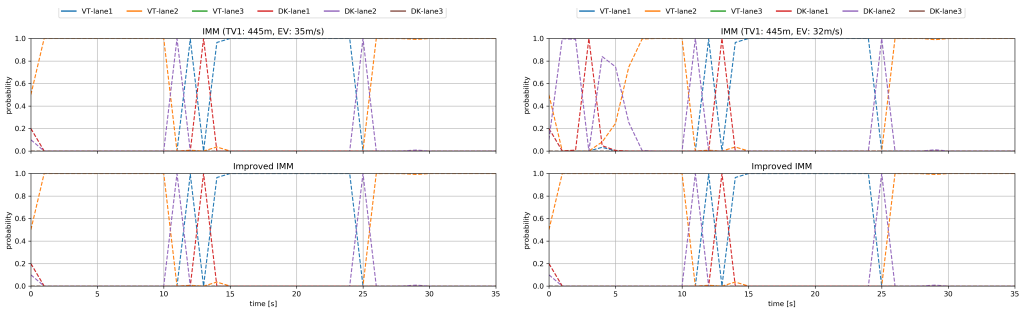
Probability Distribution Comparison of Two IMM-KFs

For completeness, this appendix provides probability distribution comparisons of the two IMM-KF-based algorithms across the remaining eight traffic scenarios not included in the main text (Section 3.4). These cases exhibit similar interaction patterns and confirm that the improved IMM-KF-based algorithm predicts motion mode changes earlier, more stably, and with higher accuracy by adaptively updating the transition probability matrix.



(a) Scenario 2

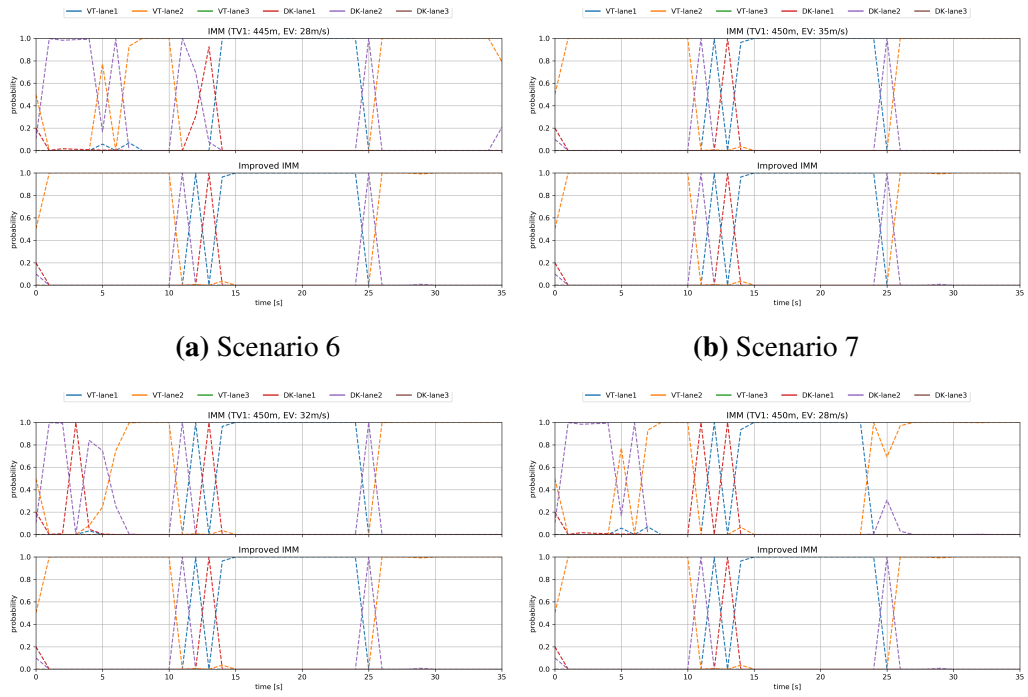
(b) Scenario 3



(c) Scenario 4

(d) Scenario 5

Fig. A.1 Probability distribution comparison of two IMM-KFs across Scenarios 2–5



(a) Scenario 6

(b) Scenario 7

(c) Scenario 8

(d) Scenario 9

Fig. A.2 Probability distribution comparison of two IMM-KFs across Scenarios 6–9

Appendix B

Parameter Setting Used in IPG CarMaker

The following parameter settings are used in the HDM of IPG CarMaker to configure different driving styles of the TVs [181].

Table B.1 Parameter setting in HDM for different driving styles

Driving Style	Defensive (D)	Normal (N)	Aggressive (A)
Estimation Ability	0.4	0.5	0.6
Safety Need	0.8	0.5	0.2
Acceleration Behavior	0.2	0.5	0.8
Speed Limit Compliance	0.7	0.5	0.3
Foresighted Driving	0.7	0.5	0.3