ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DOTTORATO DI RICERCA IN

AUTOMOTIVE ENGINEERING FOR INTELLIGENT MOBILITY

Ciclo 38

Settore Concorsuale: 09/G1 - AUTOMATICA

Settore Scientifico Disciplinare: ING-INF/04 - AUTOMATICA

PREDICTIVE MOTION PLANNING AND CONTROL FOR AUTONOMOUS RACING UNDER REGULATIONS: FROM COMPLIANCE TO STRATEGIC AWARENESS

Presentata da: Francesco Prignoli

Coordinatore Dottorato

Davide Moro

Supervisore

Paolo Falcone

Co-supervisore

Marko Bertogna

Esame finale anno 2026

UNIMORE

UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA

La presente ricerca è stata svolta nell'ambito del Dottorato di Ricerca Automotive Engineering for Intelligent Mobility*, in collaborazione tra l'Università di Bologna e l'Università di Modena e Reggio Emilia.*

Acknowledgments

As I reach the end of this journey, I feel both the duty and the pleasure to thank the people who have shaped, guided, and supported me along the way.

First and foremost, I would like to thank my supervisor, Prof. Paolo Falcone. From the very beginning and throughout the entire Ph.D. journey, he has been constantly present, supporting and guiding me while giving me the freedom to explore and learn on my own, always remaining a steady point of reference.

I am also deeply grateful to my co-supervisor, Prof. Marko Bertogna, whom I hold in great esteem. He offered me the invaluable opportunity to collaborate on his projects, allowing me to face real-world challenges and apply my research to practical problems.

My sincere thanks go to the UNIMORE Racing Team and Hipert Srl for welcoming me into their group. With them, I shared two years of racing and intense work, as well as discussions, challenges, and great satisfaction. Seeing a real race car compete without a driver is astonishing, but seeing my own code drive that same car was even more extraordinary. This experience truly made my doctoral journey unique.

I would also like to express my gratitude to Prof. Francesco Borrelli, who welcomed me into his MPC Lab during my research stay at UC Berkeley. Those months were both deeply formative and inspiring, and their outcome has become an integral part of this thesis.

Finally, I wish to thank my family and friends, who, no matter where I was, always reassured me with their constant support. I am especially grateful to my parents, who have been by my side through every moment of joy and difficulty. Without them, reaching this point would not have been possible.

Abstract

Autonomous driving in complex and dynamic environments demands decision-making systems that can operate safely, efficiently, and strategically while accounting for the behavior of other agents. As autonomy moves toward real-world deployment, vehicles must not only ensure safety and performance but also adhere to the regulations that govern interactions among traffic participants. In this context, *autonomous racing* serves as a powerful research platform for studying autonomy at its physical and computational limits, where high speeds, large accelerations, and competitive interactions expose the fundamental challenges of motion planning and control.

This thesis addresses these challenges by proposing a comprehensive motion planning and control stack for head-to-head autonomous racing with full-scale vehicles. The proposed framework bridges *predictive optimization–based planning* with the requirements of competitive multi-vehicle racing, enabling real-time execution of dynamically feasible and strategically competitive maneuvers in full-scale autonomous racing competitions.

While real-world interactions are typically governed by explicit rules naturally understood and exploit by human drivers, most existing predictive planning approaches neglect such logical and asymmetric regulations. To address this gap, the thesis develops methods for *regulation-compliant planning*, in which racing rules are explicitly encoded through mixed-integer optimization, and for *regulation-aware planning*, in which agents strategically reason about regulations constraining their opponents within a game-theoretic framework.

Simulation and experimental results demonstrate that the proposed formulations improve behavioral realism, producing competitive yet rule-compliant interactions. The contributions advance the state of the art in predictive motion planning and control toward *real-time, regulation-aware autonomy* applicable to highly dynamic and interactive driving scenarios.

Sommario

La guida autonoma in ambienti complessi e dinamici richiede sistemi decisionali in grado di operare in modo sicuro, efficiente e strategico, tenendo conto del comportamento degli altri agenti. Con l'evoluzione della guida autonoma verso applicazioni reali, i veicoli devono non solo garantire sicurezza e prestazioni, ma anche rispettare le regole che governano le interazioni tra i diversi utenti della strada. In questo contesto, l'*autonomous racing* rappresenta una potente piattaforma di ricerca per studiare i sistemi di guida autonoma ai loro limiti fisici e computazionali, dove alte velocità, elevate accelerazioni e interazioni competitive mettono in risalto le sfide fondamentali della pianificazione e del controllo del moto.

Questa tesi affronta tali sfide proponendo un'architettura di pianificazione e controllo del moto per gare tra veicoli autonomi in scala reale. Il framework sviluppato integra la *pianificazione predittiva basata su ottimizzazione* con i requisiti delle competizioni multi-veicolo, consentendo l'esecuzione in tempo reale di manovre dinamicamente fattibili e strategicamente competitive.

Nonostante le interazioni nel mondo reale siano generalmente regolate da norme esplicite, naturalmente comprese e rispettate dagli umani, la maggior parte degli approcci attuali di pianificazione predittiva trascura tali regole logiche e asimmetriche. Per colmare questa lacuna, questa tesi propone metodi per la *pianificazione conforme alle regole*, in cui le normative di gara vengono esplicitamente codificate tramite ottimizzazione mista intera, e per la *pianificazione consapevole delle regole*, in cui gli agenti ragionano strategicamente sulle limitazioni imposte agli avversari, all'interno di un framework di teoria dei giochi.

I risultati ottenuti, sia in simulazione sia attraverso prove sperimentali, dimostrano che le formulazioni proposte migliorano il realismo delle interazioni, producendo manovre competitive e conformi alle regole. Nel complesso, la tesi avanza lo stato dell'arte della pianificazione e del controllo del moto predittivo, verso un'autonomia *in tempo reale e consapevole delle regole*, capace di operare in scenari di guida altamente dinamici e interattivi.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Autonomous racing has recently emerged as a powerful research platform for advancing motion planning and control methods in autonomous driving. Beyond the excitement of competition, racing provides an extreme yet controlled environment where hardware and software can be pushed to their physical and computational limits. Vehicles operate at high speeds and near the limits of handling, while engaging in close-proximity maneuvers against opponents under strict competition rules. These conditions stress both the physical capabilities of the vehicle and the decision-making pipeline of the autonomy stack.

This thesis investigates how optimization-based predictive methods, particularly Model Predictive Control (MPC), can be used to design, deploy, and experimentally validate a comprehensive motion planning and control framework for autonomous racing, with a specific focus on multi-agent interaction governed by regulations and on real-time implementability.

1.1 Background & Motivation

Autonomous driving has proven to be as one of the most active research areas in control and robotics, promising a disruptive impact on mobility, safety, and efficiency of transportation systems. The design of reliable autonomy stacks requires solving tightly coupled problems in perception, planning, and control, all under stringent safety and real-time requirements [1], [2]. In recent years, remarkable progress has been achieved in perception, particularly in vehicle localization and environmental understanding. Advances in modern sensing technologies, such as LiDAR, radar, and high-resolution cameras, together with data-driven methods, have enabled robust lane detection, semantic scene interpretation, and precise localization even in complex urban settings [3], [4]. Building upon these advances, *motion planning and control* algorithms have become increasingly accurate and better informed, leveraging rich environmental and state estimates to generate more precise and reliable trajectories. Nonetheless, while the problem of path following on structured roads is now considered largely mature [5], [6], the design of algorithms that generate dynamically feasible trajectories in the presence of obstacles and other road users remains challenging. Such challenges stem not only from the need to account for uncertainty, anticipate the behavior of surrounding agents, and operate under stringent safety and real-time requirements [7], [8], but also from the inherently *interactive nature of traffic scenarios*, where the motion of each agent is influenced by the intents of others. Addressing such challenges requires planning frameworks that can reason about multi-agent interactions and capture the mutual coupling between decisions. Within this context, motion planning and control play a central role in the autonomy stack, bridging perception of the environment with vehicle actuation.

While much of the research in autonomous driving has focused on urban and highway scenarios, autonomous racing has recently gained attention as an experimental platform to advance the state of the art [9]. Compared to conventional driving contexts, racing is characterized by extremely challenging conditions: vehicles operate close to their physical limits, subject to high speeds and accelerations, while interacting with opponents in strategic and competitive settings. These conditions push the boundaries of perception and control systems, making racing a unique testbed for studying autonomy in complex scenarios. Early research in this domain has been driven by the established F1TENTH platform, a widely adopted 1/10th-scale testbed that enables reproducible experiments and rapid prototyping of planning and control algorithms in an affordable and accessible manner [10]. More recently, full-scale autonomous racing competitions have gained significant attention and popularity, most notably the *Indy Autonomous Challenge (IAC)* [11] and the *Abu Dhabi Autonomous Racing League (A2RL)* [12]. The IAC, held primarily in the United States at venues such as the Indianapolis Motor Speedway, Las Vegas Motor Speedway, and Texas Motor Speedway, features modified Dallara Indy NXT racecars competing at speeds approaching 300 km/h. Launched in 2024, A2RL brings autonomous racing to the Yas Marina Formula 1 circuit using modified Dallara Super Formula cars, establishing an international benchmark for autonomy at the limit of handling on road courses. Both platforms employ similar sensor suites—including LiDAR, radar, cameras, and GNSS/INS systems—and high-performance on-board computing. Each team must deploy a complete autonomous driving stack capable of controlling the vehicle in complete autonomy, responding to race control and flag signals, and managing safety procedures. These competitions represent some of the most demanding and technologically advanced environments for applied research in perception, planning, and control for autonomous vehicles, pushing research efforts further by exposing autonomy stacks to the challenges of operating full-scale racecars in adversarial, high-speed environments. Since the hardware platform, including chassis, powertrain, and actuators, is standardized and identical across teams, progress is driven almost exclusively by advances in algorithms. This standardization also facilitates the enforcement of strict safety requirements, ensuring reliable vehicle operation at extreme speeds in complete autonomy. In this way, such competitions provide a unique opportunity to investigate real-time implementability, and the robustness of perception–planning–control integration under conditions that go well beyond those encountered in conventional driving.

Motion planning and control in racing require addressing multiple layers of complexity. First, the vehicle must be controlled along dynamically feasible trajectories close to the limit of handling, which calls for hierarchical control architectures and predictive methods that account for actuator and tire friction limits. Second, strategic interactions between competing agents must be explicitly modeled: autonomous racecars are subject to competition rules and interaction regulations that define permissible maneuvers, particularly in overtaking situations. Unlike conventional approaches that neglect the reaction of other vehicles to the ego's actions, a *regulation-aware* planner can exploit such rules to enable aggressive yet valid strategies. Finally, any racing framework must satisfy demanding computational requirements: optimization-based methods must operate under tight real-time constraints. Extending such frameworks to reason about logical rules or multi-agent interactions further increases the computational burden, motivating the need for efficient formulations and tailored numerical strategies.

This thesis is motivated by these challenges and investigates how optimization-based predictive methods, specifically MPC-based, can be extended to enable reliable motion planning and control for autonomous racing in multi-agent scenarios governed by interaction regulations. The overarching thesis goal is to propose planning and control frameworks that not only ensure safety and feasibility, but also exploit the structure of competition rules to gain strategic advantage, while remaining implementable in real time on autonomous vehicles.

1.2 Problem Statement

Despite recent advances in vehicle dynamics modeling and predictive control, achieving reliable and high-performance motion planning and control for full-scale autonomous racecars operating in multi-vehicle settings remains an open challenge. Existing approaches often focus on either single-vehicle performance or isolated control components, with limited integration across the full autonomy stack. Moreover, when competitive head-to-head racing is considered, the presence of explicit regulations introduces additional layers of complexity: planners must account for both compliance with rules and the behavior of rule-constrained opponents. Finally, the computational burden of solving such predictive optimization problems in real time poses a major obstacle to their practical deployment on embedded hardware.

These challenges span multiple levels of the autonomous driving stack, from the formulation of control laws to the modeling of multi-vehicle interactions and the computational aspects of real-time execution. They can be articulated along three core challenges:

- **Comprehensive predictive planning and control for full-scale autonomous racing.** Recent international competitions such as the IAC and the A2RL have demonstrated the feasibility of deploying autonomous driving algorithms on full-scale racecars. Nevertheless, the literature on motion planning and control for such platforms remains fragmented. Most existing approaches rely on trajectory sampling or graph-based planning methods, with optimization-based methods typically confined to the control layer. The open challenge is therefore to develop a unified autonomy framework that integrates optimization-based predictive planning and control and can be systematically validated on real racecars operating at high performance, and in head-to-head racing conditions.

- **Multi-agent interaction under regulations.** Traditional motion planning methods for autonomous driving are typically conservative, prioritizing safety by minimizing interaction with other agents. In competitive racing, however, such conservatism leads to loss of performance and missed overtaking opportunities. This context motivates two complementary dimensions of regulation-based planning. The first is *regulation-compliant planning*, which ensures that generated maneuvers respect the competition rules that govern safety and fairness. The second is *regulation-aware planning*, in which explicit knowledge of these rules is exploited to anticipate opponents' constrained behaviors and to enable strategically advantageous yet valid maneuvers. While both aspects are essential for autonomous racing, existing methods largely rely on heuristic or rule-based implementations, lacking predictive frameworks that can encode and exploit regulations systematically. The open challenge is therefore to unify regulation compliance and awareness within a predictive formulation that supports interaction reasoning in multi-vehicle racing scenarios.

- **Real-time implementability of regulation-based planning.** Encoding competition rules within optimization-based planning frameworks, whether to ensure compliance or to exploit them for strategic advantage, typically leads to mixed-integer formulations of game-theoretic problems. Such classes of problems are computationally demanding and often intractable under the real-time execution time constraints. The challenge is therefore to develop computational strategies that preserve the essential structure of regulation-based multi-agent planning while enabling reliable real-time performance.

Addressing these challenges requires a unified framework that combines predictive planning and control, regulation-constrained interactions modeling, and real-time execution. This thesis is devoted to advancing such a framework, guided by the research questions outlined in the next section.

1.3 Research Questions

The challenges identified above motivate the following research questions, which structure the developments presented in this thesis:

RQ1

How can optimization-based predictive methods be extended to achieve comprehensive motion planning and control for full-scale autonomous racing in both single- and multi-vehicle scenarios? This question investigates how optimization-based methods can be extended beyond the control layer to the planning layer, enabling a hierarchical framework that integrates overtaking strategy, motion planning, and control, while remaining suitable for real-world deployment in high-speed racing environments.

RQ2

How can regulations governing vehicle interactions be systematically incorporated into predictive planning frameworks for head-to-head

autonomous racing? This question comprises two complementary aspects of regulation-based planning: **(a)** *regulation-compliant planning*, which enforces competition rules to guarantee safety and fairness; and **(b)** *regulation-aware planning*, which leverages explicit knowledge of these rules to anticipate opponents' constrained behaviors and to enable strategically advantageous yet valid maneuvers. The overarching challenge is to unify these two aspects within a optimization-based predictive framework capable of reasoning about interactions subject to regulations.

RQ3

How can real-time feasibility be achieved for regulation-aware motion planning in competitive racing scenarios? This question focuses on the computational challenges arising from embedding logical and interactive components within optimization-based frameworks, which often lead to mixed-integer and game-theoretic formulations. The objective is to develop numerical strategies and problem reformulations that enable reliable real-time operation, while preserving the structure required for strategic reasoning.

1.4 Objectives & Contributions

The objective of this thesis is to advance the state of the art in optimization-based predictive motion planning and control for autonomous racing, with particular emphasis on multi-agent interaction under competition rules. To this end, the research addresses the three questions posed in Section 1.3, and the main contributions can be summarized as follows:

- **C1 — Hierarchical MPC-based motion planning and control architecture.** A unified autonomy stack is developed that integrates high-level overtaking strategy, predictive motion planning, and feedback control within a hierarchical framework. The architecture decouples longitudinal and lateral planning while maintaining dynamic consistency through shared predictions and constraint coupling. Vehicle interactions are handled through a convex decomposition of the drivable area, enabling efficient trajectory generation under collision-avoidance constraints. At the maneuver-selection level, regulation-aware decision making is approximated by evaluating multiple planner instances, each tailored to a specific maneuver, and selecting the one yielding the lowest predicted cost. The framework has been experimentally validated in both single-vehicle and head-to-head racing, achieving speeds approaching 290 km/h.

  This contribution addresses RQ1 and is presented in Chapter 3.

- **C2 — Regulation-compliant and regulation-aware planning.** A novel optimization-based framework is proposed to formally encode competition rules within MPC formulations for head-to-head autonomous racing. The approach distinguishes between two complementary layers: **(a)** *regulation-compliant planning*, which explicitly encodes rules within the optimization problem through mixed-integer constraints; and **(b)** *regulation-aware planning*, which model the interactions as a regulation-constrained racing game, thus leveraging explicit rule knowledge to reason over opponents' constrained behaviors and enabling strategic yet rule-compliant maneuvers. The proposed formulation reveals that the *asymmetry* introduced by race regulations, assigning different responsibilities to each vehicle, enables realistic and interpretable interactions, which are typically unattainable in formulations relying solely on symmetric collision-avoidance constraints. Overall, the framework allows autonomous vehicles to reason strategically under shared regulations, bridging game-theoretic decision-making with rule-constrained optimization-based planning.

This contribution addresses RQ2 and is presented in Chapter 4.

- **C3 — Real-time regulation-aware predictive planning.** To overcome the computational challenges of mixed-integer and game-theoretic formulations, a real-time capable solution framework is developed based on a mixed-integer Sequential Quadratic Programming (MI–SQP) approach inspired by the Real-Time Iteration (RTI) scheme. The method preserves the structure of regulation-aware planning while achieving computation times compatible with embedded deployment. Experimental validation on scaled autonomous racecars demonstrates that the proposed framework achieves regulation-compliant, strategically valid maneuvers within real-time constraints.

  This contribution addresses RQ3 and is presented in Chapter 5.

1.5 Thesis Outline

The remainder of the thesis is organized as follows:

- **Chapter 2** reviews the main theoretical background and state of the art in optimization-based motion planning and control for autonomous vehicles. It surveys relevant literature on MPC, hierarchical planning architectures, and game-theoretic approaches for multi-agent interaction, highlighting the existing limitations that motivate the developments presented in this thesis.

- **Chapter 3** addresses **RQ1** and presents the proposed hierarchical MPC-based autonomy stack for full-scale autonomous racing. The chapter details the design of longitudinal and lateral planners and controllers, highlighting how the lateral planner acts as a reference governor that incorporates the closed-loop vehicle–controller dynamics within its prediction model. Vehicle interactions are handled through convex decomposition of the drivable area, and maneuver-level decision logic is introduced for overtaking scenarios. Experimental validation is provided for both single-vehicle racing on road courses and head-to-head racing on oval tracks.

- **Chapter 4** addresses **RQ2** and introduces a novel framework for Regulation-Compliant Model Predictive Control (RC-MPC) based on the Mixed Logical Dynamical (MLD) formalism, which enables the integration of logical competition rules directly within the optimization problem. Building on this foundation, the chapter presents the Regulation-Aware Game-Theoretic Planner (RA-GTP), which extends RC-MPC to reason about opponents' regulation-constrained motion and to strategically plan overtaking maneuvers. The resulting framework combines logical rule enforcement with predictive interaction reasoning, enabling autonomous vehicles to perform realistic and strategic behaviors in head-to-head racing.

- **Chapter 5** addresses **RQ3** and presents the computational strategies developed to achieve real-time feasibility for regulation-aware planning. The chapter introduces a mixed-integer Sequential Quadratic Programming (MI–SQP) approach inspired by the Real-Time Iteration (RTI) scheme; a single-pass Iterative Best Response (IBR) approach is used to approximate the solution of the Generalized Nash Equilibrium Problem (GNEP) in real time. Experimental validation on scaled autonomous racecars compares the proposed Fast RA-GTP against a baseline attacker using a regulation-agnostic opponent model. Results show that, despite the approximation introduced for achieving real-time updates, the Fast-RA-GTP effectively exploits race regulations to complete overtaking maneuvers, whereas the baseline remains overly conservative and fails to complete the pass.

- **Chapter 6** concludes the thesis by summarizing the main findings of this research. It also outlines potential directions for future research, including the study of uncertainty in opponent modeling, the analysis of GNE existence in regulation-constrained racing games under mixed strategies, and the extension of the proposed framework to multi-vehicle racing scenarios.

1.6 Publications

The research presented in this thesis has resulted in, or is closely related to, the following publications.

Publications forming the basis of this thesis

- **Chapter 3** is based on: F. Prignoli, A. Toschi, N. Musiu, P. Falcone, and M. Bertogna, "*Bridging Predictive Optimization and Real-World Racing: Motion Planning and Control for Head-to-Head Autonomous Racing*," in preparation for submission to the *IEEE Transactions on Intelligent Vehicles*, 2025.

- **Chapter 4** is based on: F. Prignoli, F. Borrelli, P. Falcone, and M. Pustilnik, "*Regulation-Aware Game-Theoretic Motion Planning for Autonomous Racing*," presented at *IEEE Intelligent Transportation Systems Conference (ITSC)*, Gold Coast, Australia, November 2025.

- **Chapter 5** is based on: F. Prignoli, S. Cao, P. Falcone, and F. Borrelli, "*Real-Time Regulation-Aware Game-Theoretic Motion Planning for Head-to-Head Autonomous Racing*," under review at *IEEE Transactions on Control Systems Technology*, 2025.

Additional related publications

The following works were produced during the doctoral period, either as a corresponding author or contributor, but are not directly included in the thesis.

- Y. S. Quan, M. Jeddi, F. Prignoli, and P. Falcone, "*Priority-Driven Constraints Softening in Safe MPC for Perturbed Systems*," *IEEE Control Systems Letters*, vol. 9, pp. 1069–1074, 2025. doi: 10.1109/LCSYS.2025.3580494.

- A. Toschi, F. Prignoli, and M. Bertogna, "*Modular Decision-Making and Drivable Areas for Multi-Agent Autonomous Racing*," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hangzhou, China: IEEE, October 2025, pp. 12435–12441. doi: 10.1109/IROS60139.2025.11246897.

- F. Prignoli, P. Falcone, A. Raji, and M. Bertogna, "*RADAR-Based Safe Pull-Over of Autonomous Racing Cars in Localization Failure Scenarios*," in *European Control Conference (ECC)*, Stockholm, Sweden: IEEE, June 2024, pp. 3644–3649. doi: 10.23919/ECC64448.2024.10591321.

- A. Raji *et al.*, "*er.autopilot 1.1: A Software Stack for Autonomous Racing on Oval and Road Course Tracks*," *IEEE Transactions on Field Robotics*, vol. 1, pp. 332–359, 2024. doi: 10.1109/TFR.2024.3501252.

- A. Raji *et al.*, "*A Tricycle Model to Accurately Control an Autonomous Racecar with Locked Differential*," in *IEEE 11th International Conference on Systems and Control (ICSC)*, Sousse, Tunisia: IEEE, Dec. 2023, pp. 782–789. doi: 10.1109/ICSC58660.2023.10449744.

Chapter 2

State of the Art

2.1 Introduction

The development of autonomous driving systems has stimulated extensive research in motion planning and control, leading to a wide spectrum of approaches for urban, highway, and racing scenarios. Several surveys provide broad overviews of the field, covering sampling-based, search-based, learning-based, and optimization-based techniques [3], [7], [8], [13]. In this thesis, the focus is placed on *optimization-based predictive methods*, which have emerged as a powerful framework for ensuring safety, feasibility, and performance in dynamic environments, and which form the methodological foundation of the proposed contributions.

The purpose of this chapter is to frame this thesis within the broader context of *predictive optimal control* techniques, rather than providing a comprehensive survey of all planning and control techniques. To this end, the review builds upon the three research questions introduced in Chapter 1, which guide both the literature analysis and the proposed contributions.

The remainder of the chapter is structured as follows. Section 2.2 introduces predictive motion planning and control for autonomous driving in general, and discusses its extension to the racing context. Section 2.3 reviews hierarchical planning and control architectures for autonomous racing, with emphasis on full-stack designs. Section 2.4 addresses regulation-based multi-agent planning frameworks. Section 2.5 reviews computational strategies for achieving real-time feasibility in regulation-compliant and interaction-aware planning frameworks. Finally, Section 2.6 summarizes the identified gaps and positions the thesis contributions with respect to the literature.

2.2 Predictive Motion Planning and Control for Autonomous Driving

Model Predictive Control (MPC) [14], [15] has emerged as one of the most widely studied frameworks for motion planning and control in autonomous driving. By explicitly accounting for vehicle dynamics, constraints, and future predictions, MPC enables the generation of dynamically feasible, safe trajectories while optimizing performance objectives such as comfort and efficiency. This predictive capability has made MPC a natural choice in applications ranging from adaptive cruise control and lane keeping to trajectory tracking [6], [16] and obstacle avoidance [17], [18] in structured and unstructured environments.

Early applications of MPC to autonomous driving primarily targeted urban and highway scenarios, where the emphasis was on passenger comfort, safety, and energy efficiency. For example, flexible and safe trajectory tracking under uncertain environments has been addressed with novel formulations of MPC that preserve feasibility in the presence of varying constraints [19], while energy-efficient planning and control have been demonstrated through predictive use of connected traffic light information [20]. Tube- and scenario-based MPC formulations have been adopted to handle multi-modal predictions of obstacles [21]. Other works developed predictive cruise control with autonomous overtaking [22] or intersection management strategies using centralized MPC [23]. These contributions established a strong foundation for constraint handling, trajectory tracking, and integration with perception modules, but typically operated under moderate accelerations and conservative safety margins.

Autonomous racing has therefore attracted increasing attention as a complementary research platform [9]. In this setting, vehicles must negotiate high speeds, large accelerations, and close-proximity interactions with opponents, all under the strict rules of competition. Racing pushes the autonomy stack beyond the comfort and safety margins of conventional driving, providing a unique opportunity to stress-test predictive planning and control methods under extreme conditions. Several works have investigated MPC-based planning and control on scaled platforms such as F1TENTH [10], which offer reproducible, low-cost testbeds for developing and benchmarking algorithms. These platforms have demonstrated the feasibility of aggressive maneuvers, learning-based improvements [24], and real-time implementations in constrained computing platforms [25], [26]. More recently, full-scale competitions such as the Indy Autonomous Challenge (IAC) [11] and the Abu Dhabi Autonomous Racing League (A2RL) [12] have emerged, deploying optimization-based predictive methods on high-performance racecars equipped with state-of-the-art sensing and computing platforms. These initiatives highlight the potential of MPC to enable reliable autonomy at the limit of handling in real-world racing scenarios.

2.3 Hierarchical Planning and Control Architectures in Autonomous Racing

The complexity of autonomous racing requires motion planning and control frameworks that can address multiple timescales and layers of decision making: long-term tactical decisions, short-horizon trajectory generation, and high-rate control actions must be coherently integrated within a unified architecture. A common approach in full-scale autonomous racing is to adopt a hierarchical architecture, where high-level modules handle strategic decisions and motion planning, while low-level controllers execute trajectories and stabilize the vehicle dynamics [9].

In practice, hierarchical architectures used in competitions such as the IAC and the A2RL often share a broadly similar structure. At the top, a *mission or* *strategy layer* handles high-level decision making, including maneuver selection, race logic, and coordination with race control and base station, typically implemented as a finite-state machine or behavior tree. At the intermediate level, a *motion planning layer* generates dynamically feasible, safe trajectories in real time based on the current measurements and perception of the environment. Typically, one or more time-optimal or minimum-curvature racing lines are computed offline through a global planner, together with static track information such as boundaries and banking angle. These precomputed data serve as inputs to the local motion planner operating online, which refines the trajectory based on the current vehicle state, opponents detection, and dynamic constraints. At the lowest level, a *control layer* executes the reference trajectory by computing actuator commands, i.e., steering angle, throttle, and brake, that ensure accurate path and speed tracking. This layer possibly interfaces with vehicle-level safety and stability systems such as anti-lock braking (ABS) and traction control, which modulate tire forces and longitudinal slip near the limits of adhesion.

Several works have investigated hierarchical decompositions following this general architecture. Motion controllers such as linear quadratic regulators (LQR), PID controllers with feedforward compensation, or MPC-based tracking controllers have been used to stabilize the vehicle and track reference trajectories [27]–[30].

On top of these, local motion planners generate collision-free trajectories for obstacle avoidance or overtaking maneuvers. Due to the non-convex nature of obstacle avoidance constraints and the computational complexity of long-horizon optimization-based approaches, such planners are often based on trajectory sampling or graph-based methods. In trajectory sampling approaches, a large set of candidate trajectories is generated by varying control inputs or using geometric curves, and each candidate is evaluated with respect to dynamic feasibility, safety margins, and racing objectives. The best feasible trajectory is then selected according to a cost function [31], [32]. Graph-based approaches instead discretize the track into a lattice of nodes and edges, often constructed in Frenét coordinates relative to a reference line. Costs are assigned to each edge based on metrics such as curvature, lateral offset, or distance, and feasible paths are obtained by shortest-path search [33], [34]. These approaches are particularly appealing due to their flexibility, computational efficiency, and the possibility of performing post-evaluation checks for feasibility and rule compliance, although this comes at the cost of a coarse discretization of the solution space.

Despite the presence of a few works reporting hierarchical motion planning and control architectures validated in full-scale autonomous racing [35]–[37], such contributions remain scarce. The majority of studies still focus on scaled platforms or on isolated components of the autonomy stack, often without full-system validation at racing speeds. Meanwhile, autonomous full-scale racing competitions are continuously evolving, achieving higher levels of performance and complexity. These events have progressed from simplified head-to-head races on parallel lanes toward multi-vehicle competitions on free racing lines, demanding increasingly sophisticated autonomy stacks capable of real-time decision making, planning, and control near to the physical limits of handling. This evolution motivates the need for updated research on hierarchical predictive frameworks that can sustain reliable operation under higher-performance and more interactive racing conditions.

This gap motivates the first research question of this thesis (RQ1), which addresses how to design a hierarchical stack of MPC-based planners and controllers that can reliably operate on full-scale racecars, including in head-to-head racing.

2.4 Regulation-Based Planning in Multi-Agent Scenarios

Multi-agent interactions represent a fundamental challenge in autonomous driving, and even more so in autonomous racing, where vehicles operate in close proximity and compete for limited track space. Although significant advances have been made in motion prediction, ranging from model-based approaches grounded in vehicle dynamics to data-driven methods based on learning from demonstrations, most planning frameworks still treat these predictions as exogenous and fixed throughout the planning horizon. This assumption implies that the future motion of surrounding agents is independent of the ego vehicle's decisions. Consequently, the ego vehicle cannot rely on other agents to react or adapt to its actions, and must ensure safety and feasibility under all opponent behaviors given by these fixed predictions. The resulting behavior is therefore conservative, particularly in competitive or densely interactive situations. A more realistic representation of interaction arises from *joint prediction and planning*, in which each agent accounts for how its own actions influence the responses of others. In this case, by coupling prediction and decision making, the planner explicitly models the interdependence of agents' objectives and constraints, leading to strategies that better capture the mutual adaptation observed in real driving and racing scenarios.

Game-theoretic approaches provide a natural framework for modeling interactive decision making among autonomous agents whose objectives and constraints are mutually dependent. In the context of motion planning, these interactions are naturally formulated as *noncooperative dynamic games*, in which each agent governs its own dynamical system and seeks to minimize an individual cost function over a finite prediction horizon [38]. The strategy of each agent corresponds to a control sequence or, equivalently, to a planned trajectory resulting from its system dynamics. The coupling among agents arises from shared or interdependent elements in their decision problems, such as coupling objectives, or limited shared resources (e.g., track space or right-of-way). Consequently, the optimal decision of each agent depends not only on its own control inputs but also on the anticipated strategies of others, giving rise to a mutually dependent optimization structure characteristic of dynamic games.

A widely adopted solution concept for noncooperative dynamic games is the *Nash equilibrium (NE)*. An equilibrium is reached when each agent's trajectory is optimal given the trajectories of all others; that is, no agent can unilaterally modify its plan to obtain a lower cost while satisfying its own constraints [39]. The Nash equilibrium represents a mutually consistent outcome in which all agents act optimally with respect to one another. A related concept is the *Stackelberg equilibrium (SE)*, which models hierarchical or sequential interactions. In this case, one or more *leaders* commit to a strategy first, and the *followers* subsequently compute their optimal responses. The resulting equilibrium captures asymmetric decision structures that naturally arise in leader–follower scenarios, where one agent initiates an action and others adapt accordingly [40].

Computing these equilibria in continuous state and control spaces is challenging, as it entails solving a set of coupled optimal control problems. Numerical methods include formulations based on *variational inequalities (VI)* [41] and iterative schemes such as the *iterative best response (IBR)* method, in which each agent repeatedly solves its own optimization problem while treating the others' trajectories as fixed [42]. These approaches have been successfully applied to interactive driving tasks such as merging, lane changing, and intersection negotiation, demonstrating that explicitly modeling mutual adaptation among agents yields more efficient and human-like behaviors [43], [44].

In autonomous racing, this interactive aspect becomes even more critical due to the adversarial nature of the task and the presence of explicit competition rules. Overtaking, defending, or blocking maneuvers are subject to constraints on safety margins, track limits, and precedence between vehicles. These *regulations governing vehicle interactions* must be enforced to ensure compliance and safety, while also being leveraged to enable aggressive but valid strategies. This motivates a distinction between two levels of rule integration:

- **Regulation-compliant planning**, where rules are encoded as hard or soft constraints in the optimization problem to ensure admissibility of the solution.

- **Regulation-aware planning**, where the planner explicitly reasons about the opponent's constraints and obligations induced by the rules, and exploits this knowledge to gain a competitive advantage.

Early works have encoded rules in trajectory-sampling or graph-based planners [34], enabling compliance through constraint filtering or post-evaluation, but without explicitly modeling the reciprocal reasoning between agents. More recent predictive and game-theoretic formulations have begun to capture these strategic dependencies [45], yet the integration of explicit regulation models within optimization-based multi-agent frameworks remains limited.

Although recent literature has explored both game-theoretic approaches and optimization-based planning for racing, the explicit integration of competition rules into predictive frameworks remains limited. Most works treat regulations implicitly, through safety margins or heuristic logic, without systematically embedding them into multi-agent predictive optimization. As a result, existing approaches either lack compliance guarantees or fail to exploit regulations for strategic advantage.

This gap motivates the second research question of this thesis (RQ2), which addresses how regulations can be explicitly modeled within optimization-based predictive planning frameworks, both to ensure compliant behavior and to enable regulation-aware reasoning in multi-agent racing.

2.5 Algorithmic Methods for Real-Time Planning Under Regulations

Encoding race regulations within predictive and optimization-based planning frameworks typically requires logical representations of the rules, such as Signal Temporal Logic (STL) [46] or Mixed Logical Dynamical (MLD) models [47]. While these formulations enable precise rule enforcement (e.g., yielding obligations or overtaking permissions), they introduce integer variables that transform the resulting optimal control problem into a Mixed-Integer Nonlinear Program (MINLP). Such formulations are notoriously challenging for real-time control, as the combinatorial nature of logic constraints can dominate computation time even for short horizons. Recent progress in real-time mixed-integer optimization has begun to close this gap, enabling feasible updates at control frequencies on the order of tens of milliseconds for relatively small optimal control problems [48]–[50].

When regulations are exploited strategically—that is, when agents reason about them rather than merely obey them—the problem naturally extends to a game-theoretic setting. Each vehicle optimizes its own trajectory while anticipating the regulation-constrained responses of the others, leading to coupled feasibility regions across players. Iterative Best-Response schemes have been widely adopted to compute approximate Generalized Nash equilibria (GNE) in real time [51], [52]. However, most existing implementations are limited to symmetric coupling constraints, such as mutual collision avoidance, and therefore fail to capture the asymmetric regulatory responsibilities that are specific to each player (e.g., right-of-way obligations or overtaking permissions). To compensate for the resulting equilibria's tendency toward conservative or indecisive behavior, sensitivity-based coupling terms have been introduced to steer the solution toward more effective strategies for a given agent [53], [54]. While these heuristics can improve empirical performance, they compromise the realism of truly competitive settings, where all agents simultaneously pursue their own objectives under asymmetric constraints.

Overall, achieving real-time implementability of *regulation-aware* multi-agent planning remains largely unresolved. Existing approaches typically address either the *logical enforcement* of regulations without incorporating strategic reasoning, or conversely, the *game-theoretic interaction* without explicit regulatory constraints. There is therefore a need for computational strategies that (i) retain the logical structure required for rule compliance, (ii) efficiently approximate strategic interactions among agents, and (iii) scale to real-time operation on embedded hardware. This gap motivates the third research question of this thesis (RQ3), which investigates how computational methods can be developed to achieve real-time feasibility of regulation-based predictive planning problems, while preserving the essential aspects of strategic reasoning in multi-agent racing.

2.6 Summary and Gap Analysis

This chapter reviewed the state of the art in predictive motion planning and control for autonomous driving and racing, with a particular focus on approaches relevant to the three research questions of this thesis.

First, predictive methods based on Model Predictive Control have been widely studied in urban and highway driving, and more recently in scaled and full-scale racing. While these works demonstrate the potential of MPC for constraint handling and aggressive maneuvering, the literature on full-scale autonomous racing remains fragmented, particularly for optimization-based planning methods. Comprehensive frameworks that integrate overtaking strategy, motion planning, and control into a unified stack, and validate them under real-world head-to-head racing, are still scarce (RQ1).

Second, multi-agent planning has been addressed through game-theoretic formulations in both general autonomous driving and racing contexts. However, competition rules are rarely incorporated explicitly into predictive frameworks. Most approaches rely on conservative safety margins or heuristic strategies, which either lack compliance guarantees or miss opportunities for strategic advantage. The distinction between regulation-compliant and regulation-aware planning remains underexplored (RQ2).

Third, real-time implementability poses a major barrier. Mixed-integer MPC and game-theoretic formulations are computationally demanding, and existing solvers struggle to deliver solutions at the rates required for onboard execution in racing. While efficient real-time methods such as RTI and SQP exist for convex MPC, their extension to regulation-based and interactive planning remains limited. Only few attempts have been made to close this gap, mostly in simulation (RQ3).

Chapter 3

Predictive Motion Planning and Control for Full-Scale Autonomous Racing

3.1 Introduction

This chapter addresses Research Question 1, which investigates how optimization-based predictive methods can be extended to enable comprehensive motion planning and control for full-scale autonomous racing in both single- and multi-vehicle scenarios. The objective is to design and experimentally validate a hierarchical autonomy stack that integrates high-level maneuver selection, predictive motion planning, and feedback control within a unified framework.

To bridge predictive, optimization-based motion planning and real-world racing, this chapter presents a motion-planning and control architecture deployed on full-scale autonomous racecars competing in international events such as the *Indy Autonomous Challenge* (IAC) [11] and the *Abu Dhabi Autonomous Racing League* (A2RL) [12]. The proposed framework integrates decoupled MPC-based planners for longitudinal and lateral motion with dedicated low-level controllers, achieving a balance between predictive capability, modularity, and real-time implementability for head-to-head racing at competition-level performance. The *lateral planner* embeds a closed-loop prediction model that explicitly includes the lateral controller—an LQR with feedforward compensation—thus acting as a *reference governor* for the closed-loop lateral dynamics [55], [56]. The *longitudinal planner* optimizes acceleration trajectories subject to curvature-dependent bounds derived from the vehicle's dynamic envelope, computed from a GGV diagram, while the resulting references are tracked through a feedforward-augmented PID controller. This formulation enforces dynamic feasibility and constraint satisfaction directly at the planning level.

For head-to-head racing, the architecture is extended with an *overtaking-strategy layer* that coordinates the parallel execution of maneuver-specific planner instances, namely *follow*, *overtake-left*, and *overtake-right*. Each planner optimizes feasible trajectories within convex workspaces shaped by regulation-consistent predictions of the opponent's motion. The maneuver corresponding to the trajectory with the lowest predicted cost is then selected as the optimal one. In this formulation,

(a) A2RL EAV-24 at the Yas Marina Circuit. **(b)** IAC AV-24 at the Indianapolis Motor Speedway.

Figure 3.1. Autonomous racecars used in the A2RL and IAC competitions.

awareness of the regulations is introduced only at the *maneuver-selection level*: each maneuver is evaluated against a different, rule-consistent prediction of the opponent, while the underlying trajectory optimizations remain independent and agnostic. This abstraction reduces the complexity of the interactive planning problem by restricting the decision space to a finite set of high-level maneuvers, rather than embedding rules or game-theoretic reasoning directly within the trajectory optimization. With this premise, the approach admits a game-theoretic interpretation in the sense of a Stackelberg game, where the ego vehicle (leader) plans while anticipating the opponent's response to each maneuver realization. This formulation balances modularity and strategic decision-making that account for racing regulations, while maintaining computational tractability.

The motion-planning and control framework presented in this chapter was developed within the UNIMORE Racing team, as part of the team's autonomous driving stack for racing competitions. The deployment of the proposed framework began in September 2023 and underwent progressive on-track validation. Preliminary integration tests were carried out at the Las Vegas Motor Speedway in January 2024, followed by road-course racing at the Yas Marina Circuit during the Abu Dhabi Autonomous Racing League (A2RL 2024) in March–April 2024 (Fig. 3.1a). Finally, the framework was tested on oval tracks at the Kentucky Speedway and used at the Indianapolis Motor Speedway during the Indy Autonomous Challenge (IAC 2024) in August–September 2024 (Fig. 3.1b). Experimental results demonstrate stable trajectory tracking with top speeds approaching 290 km/h and safe execution of high-speed overtaking maneuvers, safe execution of high-speed overtaking maneuvers, including successful passes of opponents moving at 250 km/h.

The remainder of this chapter is organized as follows. Section 3.2 reviews prior research on motion planning and control for autonomous racing and identifies the open problems motivating this study. Section 3.3 presents the overall predictive motion-planning and control framework. Section 3.4 describes the design of the motion-control layer, while Section 3.5 details the implementation of the MPC-based longitudinal and lateral planners. Section 3.6 introduces the overtaking-strategy layer for head-to-head racing and provides its game-theoretic interpretation. Section 3.7 reports the experimental setup and full-scale validation results. Finally, Section 3.8

summarizes the main findings and discusses their strengths and limitations.

The material presented in this chapter builds upon the work

- *"Bridging Predictive Optimization and Real-World Racing: Motion Planning and Control for Head-to-Head Autonomous Racing,"* by F. Prignoli, A. Toschi, N. Musiu, P. Falcone, and M. Bertogna, *in preparation for submission* to the *IEEE Transactions on Intelligent Vehicles* (2025).

3.2 Background

Optimization-based control methods, specifically Model Predictive Control, have been extensively studied and applied to autonomous racing, providing a principled predictive framework for generating aggressive yet dynamically feasible maneuvers [26], [57]. Its capability to handle multivariable dynamics, actuator constraints, and preview information within a unified optimization problem makes it particularly suitable for high-performance driving. Nonlinear MPC (NMPC) has been shown to produce time-optimal and dynamically feasible trajectories for scaled racecars operating near the limits of handling [26], and hierarchical NMPC formulations have incorporated handling and actuator constraints in real time [28]. In [57], a two-layer scheme combining a high-level point-mass MPC constrained by a GG diagram with a low-level high-fidelity MPC has been proposed and validated in simulation, while complementary feedback–feedforward steering control designs accounting for sideslip have been developed for high-speed lateral control [27].

Despite these advances, most existing studies remain limited to simulations, laboratory-scale platforms, or single-vehicle scenarios. At full scale, MPC is primarily employed at the control layer for vehicle stabilization and path tracking, whereas its use at the planning level remains limited. Extending predictive, optimization-based planning to real-world racing is particularly challenging due to the long prediction horizons required and the nonconvex constraints arising in multi-vehicle interactions such as collision avoidance. As a result, achieving robust and flexible MPC-based planning and control under these conditions remains an open challenge.

Recent research in full-scale autonomous racing has emphasized hierarchical motion planning and control architectures that balance flexibility, computational complexity, and performance. Different design choices have emerged in the context of real-world competitions such as the IAC and the A2RL. In [32], an LQR-based controller is combined with trajectory-sampling planners inspired by [58]. In [59], [60], a point-mass tube MPC is adopted at the control layer, together with a spatio-temporal graph planner extending [61]. In [29], a high-fidelity dynamic-bicycle MPC is employed for lateral control and a feedforward with PID compensation for longitudinal dynamics, while trajectory sampling is used for the planning layer in head-to-head scenarios. These frameworks illustrate the trade-off between predictive accuracy and computational tractability: sampling and graph-based planning layers offer flexibility and ease of integration with rule checks, but at the cost of coarse discretization and suboptimal trajectories.

In multi-agent racing scenarios, traditional MPC formulations typically assume static or non-reactive environments, limiting their ability to capture the interactive and strategic nature of head-to-head competition. To address this limitation, several studies on simulation and scaled platforms have investigated game-theoretic and data-driven formulations for modeling interactive behaviors [51], [62]. Game-theoretic approaches model these coupled decision problems through Nash or Stackelberg equilibrium formulations [39], [63], [64], enabling each agent to reason about the opponent's potential reactions. Such methods have demonstrated qualitatively realistic and strategic behaviors in simulation and scaled experiments, but their application to full-scale racing remains limited by computational complexity and modeling assumptions. Consequently, existing full-scale frameworks often rely on simplified strategies such as rule-based decision logic, trajectory sampling, or graph-based planning, which ensure safe interactions but sacrifice optimality [34]. Overall, real-world demonstrations of predictive, optimization-based planners capable of handling interactive, regulation-aware racing at full scale are still lacking.

In contrast to prior interaction-aware approaches, this work introduces a maneuver-level strategy layer that embeds regulation awareness in the selection of overtaking maneuvers. Right-of-way regulations are leveraged to define admissible opponent maneuvers, preventing the overly conservative behaviors typical of regulation-agnostic interaction models. Through this formulation, optimization-based planning is extended to competitive head-to-head scenarios on full-scale platforms while maintaining real-time implementability.

3.3 System Architecture

The proposed motion planning and control stack relies on two complementary information sources: offline-computed racing data and real-time onboard perception. Offline, a time-optimal racing line and the corresponding nominal speed and acceleration profiles are precomputed and used as global references during online operation. The geometric properties of the reference racing line are parameterized by the arc length s and defined by the tuple

$$\Xi_{\mathrm{trk}}(s) := \Big\{\kappa(s),\, n_\ell(s),\, n_r(s),\, \theta(s)\Big\}, \tag{3.1}$$

where $\kappa(s)$ is the curvature, $n_\ell(s)$ and $n_r(s)$ are the signed distances to the left and right track boundaries, and $\theta(s)$ is the local banking angle. The nominal motion along the racing line is described by

$$\Xi_{\mathrm{ref}}(s) := \Big\{v_{x,\mathrm{ref}}(s),\, a_{x,\mathrm{ref}}(s)\Big\}, \tag{3.2}$$

where $v_{x,\mathrm{ref}}(s)$ and $a_{x,\mathrm{ref}}(s)$ denote the reference speed and acceleration profiles. In this work, all these quantities are assumed to be available, computed offline as the solution of a global time-optimal trajectory planning problem that accounts for the vehicle's dynamics and handling characteristics.

During online operation, the proposed motion-planning and control framework leverages these precomputed data to generate dynamically feasible trajectories that minimize tracking error with respect to the racing line and speed profile. During execution, the localization and perception modules provide real-time estimates of the ego vehicle state and of the opponent's motion. Ego-state quantities are expressed

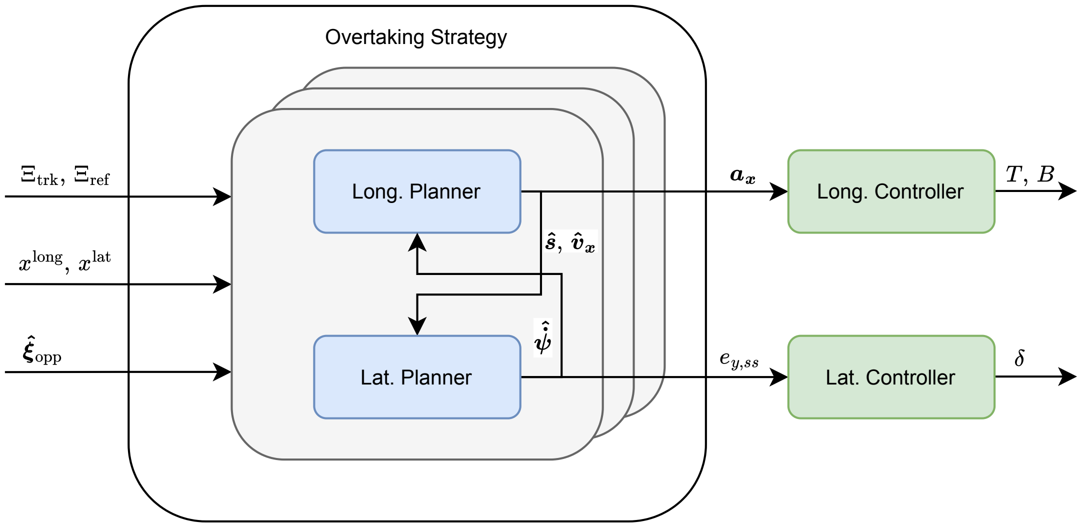


Figure 3.2. Block diagram of the motion planning and control architecture. A longitudinal and a lateral MPC-based planner form a joint planner module. Three maneuver-specific instances of this module are coordinated by the overtaking strategy layer, which also selects the active maneuver. The resulting acceleration trajectory and lateral offset command are fed to the longitudinal and lateral controllers, respectively.

in the racing-line frame (Frenet coordinates) and include the arc length s, lateral displacement e_y, heading error e_ψ, and the corresponding velocities and accelerations.

Fig. 3.2 illustrates the main modules of the system, which follows a hierarchical structure composed of three layers:

- **Control Layer:** This layer ensures accurate tracking of the reference trajectories provided by the planners. It includes a longitudinal controller based on feedforward-augmented PIDs and a lateral controller designed as a LQR with feedforward compensation. The controllers track, respectively, the reference acceleration and lateral displacement with respect to the racing line, as generated by the planning layer, and output actuator-level commands, i.e., throttle, brake, and steering angle.

- **Motion Planning Layer:** This layer generates dynamically feasible reference trajectories that respect vehicle dynamics and track constraints. It comprises two decoupled MPC-based planners governing the longitudinal and lateral motion. The *lateral planner* embeds a closed-loop prediction model that explicitly includes the lateral controller, thus acting as a reference governor to enforce constraints, including actuator bounds and handling limits derived from the vehicle's GGV envelope. The *longitudinal planner* optimizes the acceleration trajectory under curvature-dependent bounds on speed and acceleration, ensuring consistency with the same dynamic envelope. The two planners exchange predicted trajectories to maintain coherent constraint evaluation and model predictions over the horizon. In head-to-head scenarios, maneuver-specific constraints are implemented to enable overtaking and adaptive cruise control behaviors.

- **Overtake Strategy Layer:** Activated during head-to-head racing scenarios, this layer runs three planner instances in parallel corresponding to the *follow*, *overtake-left*, and *overtake-right* maneuvers. A convex decomposition of the drivable area defines the maneuver-specific constraint sets, which are provided to each planner instance. For each maneuver, the opponent's motion is predicted in a regulation-consistent manner; a maneuver-selection logic then evaluates the feasibility and predicted outcomes of all planners and selects the maneuver to execute.

The three layers operate asynchronously: the control layer runs at 100 Hz, while the planning and strategy layers run at 20 Hz. This hierarchical and modular design enables distributed implementation, component-level validation, and reliable real-time operation.

3.4 Control Design

The control layer is responsible for tracking the reference acceleration and lateral displacement provided by the motion planning layer. Two independent controllers are employed: a longitudinal controller composed of two feedforward terms for throttle and brake actuation, complemented by separate PID feedback loops acting on the acceleration tracking error; and a lateral controller based on a gain-scheduled LQR with feedforward compensation.

3.4.1 Longitudinal Control

The longitudinal controller tracks the acceleration trajectory generated by the planner using a feedforward–feedback structure. At each control step t, the planner provides an acceleration sequence $\boldsymbol{a_x} \coloneqq \{a_{x,k|t}\}_{k=0}^{N}$, defined over the planner prediction horizon N. The target acceleration $a_{x,\mathrm{tar}}(t)$ applied to the controller at each time instant t, is selected as the value corresponding to a short look-ahead index k_{LA}, i.e.,

$$a_{x,\mathrm{tar}}(t) = a_{x,k_{\mathrm{LA}}|t}, \tag{3.3}$$

representing the near-term target of the planned trajectory. This target is then mapped to a desired longitudinal force $F_{x,\mathrm{ref}}$ through a quasi-steady equilibrium relation [65]:

$$F_{x,\mathrm{ref}} = \left(m + \tfrac{4J_0}{r_w^2}\right) a_{x,\mathrm{tar}} + F_{\mathrm{drag}}(v_x) + F_{\mathrm{roll}}(v_x), \tag{3.4}$$

where m is the vehicle mass, J_0 the wheel rotational inertia (per wheel, reflected as $4J_0/r_w^2$ to the chassis), r_w the effective tire radius, and v_x the measured longitudinal speed. The resistive terms are modeled as

$$F_{\mathrm{drag}}(v_x) = \tfrac{1}{2}\,\rho\, S\, C_x\, v_x^2, \quad F_{\mathrm{roll}}(v_x) = F_z(v_x)\, C_{r0}\, e^{-2/v_x}, \tag{3.5}$$

with ρ the air density, S the frontal area, C_x the aerodynamic drag coefficient, C_{r0} the rolling-resistance coefficient, and $F_z(v_x)$ the total normal load (sum of front and rear axle loads, including static and aerodynamic contributions).

The desired wheel torque corresponding to the target longitudinal force is computed as

$$T_{w,\text{ref}} = F_{x,\text{ref}}\, r_w, \tag{3.6}$$

which is then reflected to the engine crankshaft through the currently engaged gear g as

$$T_{\text{eng,ref}} = \frac{T_{w,\text{ref}}}{\tau(g)}, \tag{3.7}$$

where $\tau(g)$ is the overall gear ratio in gear g.

The throttle feedforward command is obtained by inverting the steady-state engine map $T_{\text{eng}} = f_{\text{eng}}(n_{\text{eng}}, u^{\text{thr}})$,

$$u_{\text{ff}}^{\text{thr}} = f_{\text{eng}}^{-1}\Big(T_{\text{eng,ref}},\, n_{\text{eng}}\Big), \tag{3.8}$$

where n_{eng} is the measured engine speed and $u^{\text{thr}} \in [0, 1]$ is the normalized throttle command. The inversion is implemented as a two-dimensional lookup of the measured engine map with bilinear interpolation, providing a direct torque-based feedforward command for traction phases.

The brake feedforward command is computed from the balance between the desired decelerating force and the passive engine-braking contribution. The effective braking force to be generated by the hydraulic system is

$$F_{\text{brk,req}} = F_{x,\text{ref}} - F_{\text{eng brk}}, \tag{3.9}$$

where $F_{\text{eng brk}}$ denotes the resistive force produced by the engine at low throttle, obtained from the negative-torque region of the engine map. This required force is converted into a normalized brake command as

$$u_{\text{ff}}^{\text{brk}} = \frac{-F_{\text{brk,req}}}{(C_{Bf} + C_{Br})\,\eta_b}, \tag{3.10}$$

where C_{Bf} and C_{Br} are the front and rear brake gains (in newtons per unit command) derived from the nominal brake bias, and $\eta_b \in (0, 1]$ is the overall braking efficiency.

For feedback control, two independent discrete-time PID controllers refine the feedforward commands for traction and braking, acting on the acceleration tracking error; integral action includes saturation with back-calculation anti-windup [66]. Each loop is tuned separately to account for actuator asymmetry and different dynamics responses of traction and braking systems. To prevent simultaneous throttle and brake actuation, only one loop is active at a time: a new command is computed for one actuator only once the other's command has reached zero, ensuring bumpless command transitions. The final throttle and brake commands are

$$u^{\text{thr}} = \text{sat}_{[0,1]}\Big(u_{\text{ff}}^{\text{thr}} + u_{\text{PID}}^{\text{thr}}\Big), \tag{3.11}$$

$$u^{\text{brk}} = \text{sat}_{[0,1]}\Big(u_{\text{ff}}^{\text{brk}} + u_{\text{PID}}^{\text{brk}}\Big), \tag{3.12}$$

where the operator $\text{sat}_{[0,1]}(\cdot)$ enforce the output to stay within the specified interval.

This control structure ensures accurate tracking of the planner's acceleration profile while maintaining smooth and physically consistent transitions between traction and braking phases.

3.4.2 Lateral Control

The lateral motion control is implemented using a Linear Quadratic Regulator (LQR) designed on a linearized dynamic bicycle model expressed in the racing-line-aligned Frenet frame [5]. The controller regulates the vehicle's lateral motion relative to the racing line, with gain scheduling on the longitudinal velocity to ensure local closed-loop stability across the operating range. A feedforward term, computed at each control cycle from the linearized discrete model, compensates curvature- and banking-induced steady-state disturbances, enabling nominal offset-free tracking of a desired steady-state lateral displacement $e_{y,ss}$. This design ensures dynamically consistent tracking of the lateral offset trajectory commanded by the planner.

Let the scheduling parameter be the longitudinal velocity v_x. The continuous-time lateral dynamics, expressed in the racing-line-aligned Frenet frame, are modeled as

$$\dot{x}^{\text{lat}}(t) = \mathcal{A}(v_x)\, x^{\text{lat}}(t) + \mathcal{B}_u\, u(t) + \mathcal{B}_d(v_x)\, d(t), \tag{3.13}$$

with state $x^{\text{lat}} \in \mathbb{R}^5$ and input $u \in \mathbb{R}$ defined as

$$x^{\text{lat}} := \begin{bmatrix} e_y & \dot{e}_y & e_\psi & \dot{e}_\psi & \delta \end{bmatrix}^\top, \tag{3.14a}$$

$$u := \delta_{\text{com}}. \tag{3.14b}$$

Here, e_y and e_ψ denote the lateral and heading errors with respect to the reference racing line, δ is the front-wheel steering angle, and δ_{com} is the commanded steering input. The disturbance vector $d \in \mathbb{R}^2$

$$d := \begin{bmatrix} \kappa\, v_x \\ g\, \sin\theta \end{bmatrix}, \tag{3.15}$$

collects exogenous effects due to path curvature κ and road bank angle θ. The steering actuator dynamics are modeled as a first-order lag. Explicit expressions for $\mathcal{A}(v_x)$, $\mathcal{B}_u$, and $\mathcal{B}_d(v_x)$ are reported in the Appendix A.1.

Under a zero-order-hold (ZOH) discretization with sampling time $T_s^{\text{c}} = 10\,\text{ms}$, the continuous-time model (3.13) yields the discrete-time dynamics

$$x_{k+1}^{\text{lat}} = A(v_x)\, x_k^{\text{lat}} + B_u\, u_k + B_d(v_x)\, d_k, \tag{3.16}$$

where x_k^{lat} is the sampled state, $u_k = \delta_{\text{com},k}$ is the piecewise constant commanded steering input, and the matrices $A(v_x)$, B_u, $B_d(v_x)$ result from the numerical ZOH discretization of $\mathcal{A}(v_x)$, $\mathcal{B}_u$, $\mathcal{B}_d(v_x)$, respectively.

To penalize variations in steering rate, the control input is reformulated in incremental form [14]. Since the steering state δ_k is already included in the state x_k^{lat}, this can be done without augmenting the system. The incremental input Δu_k is defined as

$$\Delta u_k := \delta_{\text{com},k} - \delta_k, \tag{3.17}$$

where δ_k is extracted from the state vector using the selector matrix $S_\delta \in \mathbb{R}^{1\times 5}$

$$S_\delta := \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \delta_k = S_\delta\, x_k^{\text{lat}}. \tag{3.18}$$

Substituting $u_k = \Delta u_k + S_\delta\, x_k^{\mathrm{lat}}$ into the dynamics (3.16) yields

$$x_{k+1}^{\mathrm{lat}} = A(v_x)\, x_k^{\mathrm{lat}} + B_u\, (\Delta u_k + S_\delta\, x_k^{\mathrm{lat}}) + B_d(v_x)\, d_k. \tag{3.19}$$

Grouping terms, we define the modified state matrix:

$$A_\Delta(v_x) \coloneqq A(v_x) + B_u S_\delta, \tag{3.20}$$

so that the incremental-input dynamics become

$$x_{k+1}^{\mathrm{lat}} = A_\Delta(v_x)\, x_k^{\mathrm{lat}} + B_u\, \Delta u_k + B_d(v_x)\, d_k. \tag{3.21}$$

The control law is defined as

$$\Delta u_k = -K(v_x)\, x_k + \Delta u_{\mathrm{ff},k}, \tag{3.22}$$

where $K(v_x)$ is the gain-scheduled LQR feedback matrix obtained by solving the discrete-time algebraic Riccati equation for the system (3.21). The feedforward term $\Delta u_{\mathrm{ff},k}$ is designed to compensate steady-state disturbances arising from path curvature and road banking, thereby reducing steady-state tracking error. For notational simplicity, the dependence on the scheduling variable v_x is omitted in the following derivation. The resulting closed-loop dynamics with the control law (3.22) are

$$x_{k+1}^{\mathrm{lat}} = (A_\Delta - B_u K)\, x_k^{\mathrm{lat}} + B_u\, \Delta u_{\mathrm{ff},k} + B_d\, d_k. \tag{3.23}$$

At steady state, $x_{k+1}^{\mathrm{lat}} = x_k^{\mathrm{lat}} = x_{\mathrm{ss}}^{\mathrm{lat}}$, and the closed-loop system (3.23) satisfies

$$x_{\mathrm{ss}}^{\mathrm{lat}} = (A_\Delta - B_u K)\, x_{\mathrm{ss}}^{\mathrm{lat}} + B_u\, \Delta u_{\mathrm{ff}} + B_d\, d, \tag{3.24}$$

Rearranging terms gives

$$M\, x_{\mathrm{ss}}^{\mathrm{lat}} = B_u\, \Delta u_{\mathrm{ff}} + B_d\, d, \qquad M \coloneqq I - (A_\Delta - B_u K), \tag{3.25}$$

which admits the solution $x_{\mathrm{ss}}^{\mathrm{lat}} = M^{-1}(B_u\, \Delta u_{\mathrm{ff}} + B_d\, d)$. The matrix M is nonsingular under standard LQR design conditions ensuring asymptotic closed-loop stability. Let $S_{e_y} \in \mathbb{R}^{1\times 5}$ be the selector row vector extracting the lateral error:

$$S_{e_y} \coloneqq \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad e_{y,\mathrm{ss}} = S_{e_y} x_{\mathrm{ss}}^{\mathrm{lat}}. \tag{3.26}$$

Then, the steady-state lateral error is

$$e_{y,\mathrm{ss}} = S_{e_y} M^{-1}(B_u\, \Delta u_{\mathrm{ff}} + B_d\, d). \tag{3.27}$$

Solving for Δu_{ff}, assuming $S_{e_y} M^{-1} B_u \neq 0$, yields

$$\Delta u_{\mathrm{ff}} = \underbrace{\frac{1}{S_{e_y} M^{-1} B_u}}_{\alpha(v_x)}\, e_{y,\mathrm{ss}} + \underbrace{\frac{-S_{e_y} M^{-1} B_d}{S_{e_y} M^{-1} B_u}}_{\beta(v_x)}\, d. \tag{3.28}$$

Hence, the feedforward term takes the form

$$\Delta u_{\mathrm{ff}} = K_{\mathrm{ff}}(v_x) \begin{bmatrix} e_{y,\mathrm{ss}} \\ d \end{bmatrix}, \qquad K_{\mathrm{ff}}(v_x) \coloneqq \begin{bmatrix} \alpha(v_x) & \beta(v_x) \end{bmatrix}, \tag{3.29}$$

where $e_{y,\text{ss}} \in \mathbb{R}$ denotes the desired steady-state lateral offset and $d \in \mathbb{R}^2$ is the disturbance vector (3.15). The feedforward coefficients $\alpha(v_x) \in \mathbb{R}$ and $\beta(v_x) \in \mathbb{R}^{1\times 2}$ are computed from the system matrices $A_\Delta(v_x), B_u, B_d(v_x)$ and the LQR gain $K(v_x)$, based on the steady-state analysis described above.

Finally, substituting the feedforward term (3.29) into the closed-loop dynamics (3.23) yields the following closed-loop model:

$$x_{k+1}^{\text{lat}} = \underbrace{(A_\Delta - B_u K)}_{A_{\text{cl}}}\, x_k^{\text{lat}} + \underbrace{B_u\, \alpha}_{B_{u,\text{cl}}}\, e_{y,\text{ss}} + \underbrace{\Big(B_u\, \beta + B_d\Big)}_{B_{d,\text{cl}}}\, d_k, \tag{3.30}$$

which defines the closed-loop dynamics governing the interaction between planning and control: the planner commands a desired steady-state lateral deviation $e_{y,\text{ss}}$ with respect to the racing line, while the controller stabilizes the closed-loop system and compensates for curvature- and banking-induced disturbances through the combined action of feedback and feedforward terms.

3.5 Motion Planning

The motion planning layer computes reference trajectories that are dynamically feasible, with respect to the vehicle's handling limits, and safe under head-to-head racing conditions. Two decoupled MPC-based receding-horizon planners are employed for longitudinal and lateral motion: the longitudinal planner generates acceleration trajectories that track the desired speed and acceleration profiles (3.2), while the lateral planner computes a desired lateral offset trajectory with respect to the reference racing line (3.1). The two planners exchange predicted trajectories over the horizon to ensure consistency in constraints and prediction model evaluation over the planning horizon. In head-to-head racing, this layer embeds interaction constraints with the opponent vehicle, enabling the predictive planners to generate both following and overtaking maneuvers, while maintaining safe separation.

Dynamic feasibility is enforced through the vehicle's combined longitudinal–lateral acceleration envelope, or *GGV envelope*. Let $\mathcal{G}^\pi(v,\theta) \subset \mathbb{R}^2$ denote the admissible set of longitudinal and lateral accelerations at speed v and road banking θ, parameterized by the vehicle and tire characteristics π (including powertrain, braking, aerodynamic, and friction limits), which are assumed to be available. The envelope is derived from the vehicle-tire dynamics using Pacejka (Magic Formula) [67] tire models with parameters identified from experimental data, thereby capturing the combined acceleration limits across the operating range.

Given the two pairs (v,θ), (a_x, a_y), we define the following intervals of admissible longitudinal and lateral accelerations, respectively:

$$\Big[\underline{a}_x(v,\theta,a_y),\, \overline{a}_x(v,\theta,a_y)\Big] \coloneqq \mathcal{G}^\pi_{a_x}(v,\theta), \tag{3.31}$$

$$\Big[\underline{a}_y(v,\theta,a_x),\, \overline{a}_y(v,\theta,a_x)\Big] \coloneqq \mathcal{G}^\pi_{a_y}(v,\theta), \tag{3.32}$$

where

$$\mathcal{G}^\pi_{a_y}(v,\theta) \triangleq \{a_x : (a_x, a_y) \in \mathcal{G}^\pi(v,\theta)\} \quad \subset \mathbb{R}, \tag{3.33a}$$

$$\mathcal{G}^\pi_{a_x}(v,\theta) \triangleq \{a_y : (a_x,\, a_y) \in \mathcal{G}^\pi(v,\theta)\} \quad \subset \mathbb{R}. \tag{3.33b}$$

3.5.1 MPC-based Planning Problem Formulation

Both the longitudinal and lateral planners are formulated as constrained optimal control problems over a finite-time horizon, which result in quadratic programs (QP) solved in a receding-horizon fashion. The QPs feature a convex quadratic objective, affine dynamic constraints, and polyhedral inequality constraints, and can be efficiently solved using well-known numerical solvers for convex quadratic programming [68], [69]. The linear, possibly time-varying, system dynamics are enforced by the equality constraints

$$\begin{aligned} x_{k+1|t} &= A_{k|t}\, x_{k|t} + B_{k|t}\, u_{k|t} + E_{k|t}\, w_{k|t} \qquad k \in \mathcal{I}_0^N, \\ x_{0|t} &= x_t, \end{aligned} \tag{3.34}$$

where $x_{k|t} \in \mathbb{R}^{n_x}$ and $u_{k|t} \in \mathbb{R}^{n_u}$ denote the predicted state and input, and $w_{k|t} \in \mathbb{R}^{n_w}$ represents known external inputs.

The cost function penalizes state and input deviations from their references, as well as violations of soft constraints:

$$\begin{aligned} J_t = \sum_{k=0}^{N-1} \Big(&\|x_{k|t} - x_{\mathrm{ref},k|t}\|_Q^2 + \|u_{k|t} - u_{\mathrm{ref},k|t}\|_R^2 \\ &+ \ell_{\mathrm{slack}}(s_{k|t}^-, s_{k|t}^+) \Big) + \|x_{N|t} - x_{\mathrm{ref},N|t}\|_P^2 \\ &+ \ell_{\mathrm{slack}}(s_{N|t}^-, s_{N|t}^+), \end{aligned} \tag{3.35}$$

where $Q \succeq 0$, $R \succ 0$, and $P \succeq 0$ are weighting matrices, penalizing the deviation of the state and input from the references $x_{\mathrm{ref}} \in \mathbb{R}^{n_x}$ and $u_{\mathrm{ref}} \in \mathbb{R}^{n_u}$, respectively, over the prediction horizon N. The variables $s^- \in \mathbb{R}_{\geq 0}^{n_{s^-}}$, $s^+ \in \mathbb{R}_{\geq 0}^{n_{s^+}}$ slack the lower and upper bounds, respectively, and are weighted by ℓ_1 and ℓ_2 penalty functions as

$$\ell_{\mathrm{slack}}(s^-, s^+) \;\triangleq\; \rho_{1,-}^\top s^- + \rho_{1,+}^\top s^+ + \|s^-\|_{\Sigma_-}^2 + \|s^+\|_{\Sigma_+}^2. \tag{3.36}$$

Polyhedral constraints on state, input, and mixed state–input pairs are optionally softened with the slack variables $s_{k|t}^-$, $s_{k|t}^+$ as

$$\underline{x}_{k|t} - J_x\, s_{k|t}^- \;\leq x_{k|t} \;\leq\; \overline{x}_{k|t} + J_x\, s_{k|t}^+ \qquad k \in \mathcal{I}_0^N, \tag{3.37a}$$

$$\underline{u}_{k|t} - J_u\, s_{k|t}^- \;\leq u_{k|t} \;\leq\; \overline{u}_{k|t} + J_u\, s_{k|t}^+ \qquad k \in \mathcal{I}_0^{N-1}, \tag{3.37b}$$

$$\underline{g}_{k|t} - J_g\, s_{k|t}^- \;\leq C_{k|t} x_{k|t} + D_{k|t} u_{k|t} \;\leq\; \overline{g}_{k|t} + J_g\, s_{k|t}^+ \qquad k \in \mathcal{I}_0^{N-1}, \tag{3.37c}$$

$$\underline{g}_{N|t} - J_g^N\, s_{N|t}^- \;\leq C_{N|t} x_{N|t} \;\leq\; \overline{g}_{N|t} + J_g^N\, s_{N|t}^+, \tag{3.37d}$$

$$0 \;\leq s_{k|t}^-,\, s_{k|t}^+ \qquad k \in \mathcal{I}_0^N \tag{3.37e}$$

where J_x, J_u, J_g, J_g^N are component-wise nonnegative matrices that introduce the slack variables into the corresponding bounds (unused entries can be zeroed to disable softening on selected inequalities).

Let the stacked decision variable be

$$\boldsymbol{z}_t := \Big\{ \{x_{k|t}\}_{k=0}^N,\ \{u_{k|t}\}_{k=0}^{N-1},\ \{s_{k|t}^-\}_{k=0}^N,\ \{s_{k|t}^+\}_{k=0}^N \Big\}. \tag{3.38}$$

The system dynamics equality constraints (3.34) and the polyhedral inequality constraints (3.37) are collected into the admissible set

$$\mathcal{F}_t \;:=\; \Big\{ \boldsymbol{z}_t \;\Big|\; (3.34),\; (3.37) \Big\}. \tag{3.39}$$

The resulting finite-horizon optimization problem is stated below.

Problem 1 (Finite-Horizon Planning Problem)**.** Given the quadratic cost function $J_t(\boldsymbol{z}_t)$ defined in (3.35), the admissible polytope $\mathcal{F}_t$ in (3.39), and the stacked decision variable $\boldsymbol{z}_t$ in (3.38), the finite-horizon optimization problem, solved in a receding-horizon fashion at each planning step t, is formulated as

$$\mathcal{P}_t : \min_{\boldsymbol{z}_t} \; J_t(\boldsymbol{z}_t) \quad \text{s.t.} \quad \boldsymbol{z}_t \in \mathcal{F}_t. \tag{3.40}$$

The overall motion-planning problem consists of two instances of the generic formulation introduced in Problem 1, $\mathcal{P}_t^{\text{long}}$ and $\mathcal{P}_t^{\text{lat}}$, which operate on the longitudinal and lateral vehicle dynamics, respectively. Their joint formulation is stated below.

Definition 1 (Joint Motion-Planning Problem)**.** At each planning time instant t, the motion planner solves two instances of the finite-horizon problem (3.40), formulated for the longitudinal and lateral motion dynamics. The overall planning problem is thus defined as

$$\Pi_t := \Big(\mathcal{P}_t^{\text{long}},\, \mathcal{P}_t^{\text{lat}} \Big), \tag{3.41}$$

where

$$\mathcal{P}_t^{\text{long}} := \min_{\boldsymbol{z}_t^{\text{long}}} \; J_t^{\text{long}}(\boldsymbol{z}_t^{\text{long}}) \quad \text{s.t.} \quad \boldsymbol{z}_t^{\text{long}} \in \mathcal{F}_t^{\text{long}}, \tag{3.42}$$

$$\mathcal{P}_t^{\text{lat}} := \min_{\boldsymbol{z}_t^{\text{lat}}} \; J_t^{\text{lat}}(\boldsymbol{z}_t^{\text{lat}}) \quad \text{s.t.} \quad \boldsymbol{z}_t^{\text{lat}} \in \mathcal{F}_t^{\text{lat}}. \tag{3.43}$$

The formulations of $\mathcal{P}_t^{\text{long}}$, $\mathcal{P}_t^{\text{lat}}$ are detailed in the following subsections.

Remark 1 (Constraint Softening and Slack Variables)*.* In the proposed approach, constraints play a central role in shaping the planned trajectory, and many of them are typically active at the optimal solution. As a consequence, small constraint violations are likely to occur in practice due to modeling errors, discretization effects, or measurement noise. To enhance robustness, all constraints are therefore softened through the introduction of nonnegative slack variables as in (3.37), ensuring that the solver always returns a solution even when the original hard-constrained problem becomes infeasible. This strategy is preferable in real-world deployment, where a solution that slightly relaxes the constraints is more desirable than solver failure. To guarantee that constraint violations occur only when no strictly feasible solution exists for the original problem, the slack variables are penalized using ℓ_1-norm terms with properly sized weights, as discussed in [70]. Alternative methods for enforcing constraint priorities exist, such as lexicographic optimization [71] or data-driven methods [72], but are not considered here. In the following derivations, all constraints are assumed to be soft, unless stated otherwise.

3.5.2 Longitudinal Motion Planner

The longitudinal motion planner computes smooth acceleration trajectories that track the nominal offline speed profile in (3.2) while enforcing speed and acceleration constraints derived from the GGV handling envelope. In head-to-head scenarios, the planner provides adaptive cruise control functionality as well, to maintain a safe longitudinal gap to the leading vehicle.

Prediction Model

The longitudinal planner is formulated as a discrete-time MPC Problem (1) using a jerk-limited kinematic model. The state vector $x^{\text{long}} \in \mathbb{R}^3$ is defined as

$$x^{\text{long}} = \begin{bmatrix} s & v_x & a_x \end{bmatrix}^\top, \tag{3.44}$$

where s denotes the arc length along the reference path, v_x the longitudinal velocity, and a_x the longitudinal acceleration. The control input is the jerk j_x, which enables direct penalization and constraint of abrupt acceleration changes. The discrete-time dynamics are obtained via zero-order hold of the continuous-time integrator chain:

$$\dot{s} = v_x, \quad \dot{v}_x = a_x, \quad \dot{a}_x = j_x, \tag{3.45}$$

and enforced as equality constraints (3.34). Here, we assume a decoupled evolution of the arc length s, neglecting its nonlinear dependence on the path curvature, and adopting the approximation

$$\dot{s} = \frac{v \cos(e_\psi)}{1 - \kappa(s)\, e_y} \approx v_x, \tag{3.46}$$

which holds for small lateral and heading errors, namely e_y and e_ψ. This preserves linearity while providing sufficient accuracy at the planning level. The model is discretized with a sampling time of $T_d^{\text{long}} = 0.1\,\text{s}$ and a prediction horizon of $N^{\text{long}} = N = 80$ steps. Notably, the planner runs at $T_s^{\text{p}} = 0.05\,\text{s}$ cycle time, updating all variables at every iteration, while maintaining a coarser discretization to extend the prediction horizon without increasing the problem size.

Reference Tracking

The longitudinal planner tracks the offline reference speed and acceleration profiles in (3.2). Let $\boldsymbol{s}_{t-1} \coloneqq \{s_{k|t-1}\}_{k=0}^{N} \in \mathbb{R}^{N+1}$ denote the predicted trajectory for s at the previous planning step $t-1$, and let $\mathcal{S} : \mathbb{R}^{N+1} \to \mathbb{R}^{N+1}$ be the receding-horizon shift operator, defined as

$$\mathcal{S}\Big(\boldsymbol{s}_{t-1}\Big) \coloneqq \big\{\{s_{k|t-1}\}_{k=0}^{N-1},\, s_{N|t-1} + \big(s_{N|t-1} - s_{N-1|t-1}\big)\big\}, \tag{3.47}$$

with the terminal element obtained by linear extrapolation of the last segment[1]. Accordingly, an estimate $\hat{\boldsymbol{s}}_t \in \mathbb{R}^{N+1}$ for the predicted arc-length trajectory at time t is obtained as

$$\hat{\boldsymbol{s}}_t \coloneqq \{\hat{s}_{k|t}\}_{k=0}^{N} = \mathcal{S}\Big(\boldsymbol{s}_{t-1}\Big). \tag{3.48}$$

[1] Other extrapolation methods may be used depending on the selected variable, to preserve domain consistency.

The speed and acceleration component of the reference state $x_{\mathrm{ref,k|t}}$ over the horizon N are then obtained by sampling the offline profiles (3.2) along predicted arc-length $\hat{\boldsymbol{s}}_t$:

$$\boldsymbol{v}_{\mathbf{x,ref}_t} \coloneqq \{v_{\mathrm{x,ref}}(\hat{s}_{k|t})\}_{k=0}^{N}, \qquad \boldsymbol{a}_{\mathbf{x,ref}_t} = \{a_{\mathrm{x,ref}}(\hat{s}_{k|t})\}_{k=0}^{N}. \tag{3.49}$$

This approach preserves linearity by breaking the nonlinear dependence of the reference speed and acceleration profiles on the state s, thus avoiding online linearization or Jacobian computation, at the cost of mild suboptimality when consecutive predictions differ significantly.

Finally, the arc-length s is not penalized in the cost function, i.e., the corresponding weight is set to zero, and the reference for the jerk command is always set to zero.

Constraint Generation

To keep the constraints linear in the decision variables, we break the nonlinear dependence on the state variables by using predictions from the previous solution to compute estimates for the next predictions. As for the state reference, the shift operator (3.48) is used to compute estimates for the predicted speed and acceleration trajectories as $\hat{\boldsymbol{v}}_{\boldsymbol{x}t} \coloneqq \mathcal{S}(\boldsymbol{v}_{\boldsymbol{x}t}) \in \mathbb{R}^{N+1}$ and $\hat{\boldsymbol{a}}_{\boldsymbol{x}t} \coloneqq \mathcal{S}(\boldsymbol{a}_{\boldsymbol{x}t}) \in \mathbb{R}^{N+1}$, respectively. The corresponding estimates for predicted track curvature and banking angle are obtained by sampling the racing line (3.1):

$$\hat{\boldsymbol{\kappa}}_t \coloneqq \{\kappa(\hat{s}_{k|t})\}_{k=0}^{N}, \qquad \hat{\boldsymbol{\theta}}_t \coloneqq \{\theta(\hat{s}_{k|t})\}_{k=0}^{N}. \tag{3.50}$$

The lateral acceleration demand at each step k is then estimated as

$$a_{y,k|t}^{\mathrm{dem}} = \hat{v}_{k|t}^2\, \hat{\kappa}_{k|t} - g\sin(\hat{\theta}_{k|t}), \tag{3.51}$$

representing the lateral load due to curvature and road banking. Then, it is possible to extract the longitudinal acceleration limits at step k from the GGV envelop projection (3.31) as

$$\begin{aligned} \underline{a}_{x,k|t} &= \underline{a}_x(\hat{v}_{k|t}, \hat{\theta}_{k|t}, a_{y,k|t}^{\mathrm{dem}}), \\ \overline{a}_{x,k|t} &= \overline{a}_x(\hat{v}_{k|t}, \hat{\theta}_{k|t}, a_{y,k|t}^{\mathrm{dem}}), \end{aligned} \tag{3.52}$$

and are enforced as simple state bounds

$$\underline{a}_{x,k|t} \;\leq\; a_{x,k|t} \;\leq\; \overline{a}_{x,k|t}, \quad k \in \mathcal{I}_0^N. \tag{3.53}$$

Similarly, the maximum admissible longitudinal velocity at each step is obtained by imposing that the lateral acceleration required by the racing line curvature does not exceed its upper bound in (3.32). Given the lateral demand (3.51), the active lateral limit is selected as

$$a_{y,k|t}^{\mathrm{lim}} = \begin{cases} \overline{a}_y(\hat{v}_{k|t}, \hat{\theta}_{k|t}, \hat{a}_{x,k|t}), & \text{if } a_{y,k|t}^{\mathrm{dem}} \geq 0, \\ \underline{a}_y(\hat{v}_{k|t}, \hat{\theta}_{k|t}, \hat{a}_{x,k|t}), & \text{if } a_{y,k|t}^{\mathrm{dem}} < 0, \end{cases} \tag{3.54}$$

where the appropriate branch of the GGV envelope is used depending on the turning direction, i.e., the sign of $a_{y,k|t}^{\mathrm{dem}}$. The maximum admissible longitudinal velocity then follows from the steady-state relation between curvature and lateral acceleration:

$$\overline{v}_{k|t} = \sqrt{\frac{|a^{\lim}_{y,k|t}| + g\sin(\hat{\theta}_{k|t})}{\max(|\hat{\kappa}_{k|t}|,\, \kappa_{\min})}}, \tag{3.55}$$

with a small threshold $\kappa_{\min} > 0$ preventing division by zero on nearly straight segments. The resulting constraint

$$0 \leq v_{x,k|t} \leq \overline{v}_{k|t}, \quad k \in \mathcal{I}_0^N \tag{3.56}$$

limits the velocity trajectory to values consistent with the predicted curvature, banking, and longitudinal load.

Adaptive Cruise Control Constraints To ensure safe interaction with a leading opponent, the longitudinal planner enforces constraints on the arc-length trajectory $\boldsymbol{s}_t$. Given a forecast of the opponent's trajectory $\hat{\boldsymbol{s}}_{\mathbf{opp}_t} \coloneqq \{\hat{s}_{\text{opp},k|t}\}_{k=0}^N$, a minimum spatial gap is imposed as

$$s_{k|t} \leq \hat{s}_{\text{opp},k|t} - s_{\text{safe}}, \quad k \in \mathcal{I}_0^N, \tag{3.57}$$

where s_{safe} is a fixed safety margin.

In addition, a time-gap constraint ensures a minimum temporal headway proportional to the ego velocity:

$$s_{k|t} + t_{\text{gap}}\, v_{x,k|t} \leq \hat{s}_{\text{opp},k|t}, \quad k \in \mathcal{I}_0^N, \tag{3.58}$$

where t^{gap} is the desired time headway parameter. Both (3.57)–(3.58) are affine in the optimization variables $(s_{k|t}, v_{x,k|t})$, and can be casted as state bounds (3.37a) and affine state constraints (3.37c)-(3.37d), respectively. These constraints embed the adaptive cruise control (ACC) functionality directly within the optimization-based planning problem, allowing the vehicle to regulate its longitudinal motion and maintain safe inter-vehicle distances in head-to-head racing scenarios.

3.5.3 Lateral Motion Planner

The lateral motion planner optimizes a sequence of steady-state lateral error set-points $\boldsymbol{e}_{\boldsymbol{y},\mathbf{ss}_t} \coloneqq \{e_{y,\text{ss},k|t}\}_{k=0}^N$. At each planning step, the first element $e_{y,\text{ss},0|t}$ is sent to the low-level feedback controller introduced in Section 3.4.2, in a receding horizon fashion.

It is worth noting that, even in single-vehicle scenarios, the planner remains essential, as it endows the control layer with predictive capability against the known upcoming disturbances, i.e., curvature and banking variations along the racing line. Without this predictive layer, the lateral controller would act solely in a reactive manner, relying on instantaneous error feedback rather than anticipating future path changes. Moreover, the predictive formulation allows the planner to enforce future feasibility with respect to the vehicle handling, actuator constraints, and track limits.

In head-to-head racing, the lateral planner provides overtaking functionality by enforcing a safe lateral separation from the opponent vehicle.

Remark 2 (Temporal Discretization Consistency). The longitudinal and lateral planners operate at the same update rate $T_s^{\mathrm{p}}=0.05\,\mathrm{s}$ but on distinct temporal grids: the longitudinal planner employs a discretization step of 0.1 s, whereas the lateral planner uses a finer step of 0.05 s. To ensure temporal consistency, the predicted longitudinal trajectories are resampled onto the lateral planner's time grid via linear interpolation prior to the computation of the speed-dependent system matrices and feasibility constraints. This resampling step is omitted in the following derivations for notational simplicity.

Prediction Model

The prediction model is derived from the closed-loop dynamics of the vehicle–controller system defined in (3.30). At each time t, the discrete-time LPV model used in the planner is

$$x_{k+1|t}^{\mathrm{lat}} = A_{k|t}^{\mathrm{p}}\, x_{k|t}^{\mathrm{lat}} + B_{u,k|t}^{\mathrm{p}}\, e_{y,\mathrm{ss},k|t} + B_{d,k|t}^{\mathrm{p}}\, \hat{d}_{k|t}, \tag{3.59}$$

where $x_{k|t}^{\mathrm{lat}} \in \mathbb{R}^5$ is the lateral state vector (3.14a), $e_{y,\mathrm{ss},k|t} \in \mathbb{R}$ the commanded steady-state lateral offset, and $\hat{d}_{k|t}$ the predicted disturbance (3.15) according the longitudinal planner predictions for speed $\hat{\boldsymbol{v}}_{\boldsymbol{x}t}$, curvature $\hat{\boldsymbol{\kappa}}_t$, and banking $\hat{\boldsymbol{\theta}}_t$ as in (3.50):

$$\hat{d}_{k|t} = \begin{bmatrix} \hat{\kappa}_{k|t}\, \hat{v}_{x,k|t} \\ g \sin(\hat{\theta}_{k|t}) \end{bmatrix}, \quad k \in \mathcal{I}_0^N \tag{3.60}$$

To account for the different sampling times of the controller (T_s^{c}) and the planner (T_s^{p}), the matrices A^{p}, B_u^{p}, and B_d^{p} are obtained by downsampling the controller-level closed-loop matrices by a factor[2] $n_{\mathrm{d}} = T_s^{\mathrm{p}}/T_s^{\mathrm{c}}$:

$$\begin{aligned} A_{k|t}^{\mathrm{p}} &= (A_{\mathrm{cl}}(\hat{v}_{x,k|t}))^{n_{\mathrm{d}}}, \\ B_{u,k|t}^{\mathrm{p}} &= \sum_{i=0}^{n_{\mathrm{d}}-1} (A_{\mathrm{cl}}(\hat{v}_{x,k|t}))^i B_{u,\mathrm{cl}}, \\ B_{d,k|t}^{\mathrm{p}} &= \sum_{i=0}^{n_{\mathrm{d}}-1} (A_{\mathrm{cl}}(\hat{v}_{x,k|t}))^i B_{d,\mathrm{cl}}(\hat{v}_{x,k|t}). \end{aligned} \tag{3.61}$$

This aggregation corresponds to an exact zero-order-hold discretization over n_{d} controller cycles, ensuring that the planner prediction model captures the effective closed-loop response at the slower planning rate. Finally, the prediction horizon length is $N^{\mathrm{lat}} = N = 80$ steps.

By optimizing over the sequence of steady-state set-points $\boldsymbol{e}_{\boldsymbol{y},\mathbf{ss}_t}$, the lateral planner embeds the closed-loop vehicle–controller dynamics into its prediction model, and enforce actuator and vehicle handling constraints directly in the reference generation. It therefore acts as a *reference governor* that adjust the reference for the controller, ensuring constraint satisfaction and feasibility under the actual closed-loop response of the system.

[2]The planner sampling time T_s^{p} must be an integer multiple of the controller sampling time T_s^{c}.

Constraint Generation

Besides steering actuator constraints enforced as bounds on the steering state component, to ensure lateral stability and respect the vehicle's handling limits, the planner enforces state constraints on the yaw-rate error and side-slip angle, derived from the admissible lateral acceleration envelope.

Yaw-rate constraints. The yaw-error rate satisfies

$$\dot{e}_\psi = \dot{\psi} - \dot{\psi}_{\text{path}} = \dot{\psi} - v_x\,\kappa, \tag{3.62}$$

where $\dot{\psi}_{\text{path}} = v_x\,\kappa$ is the yaw rate of the reference path. From the GGV envelope (3.32), the admissible yaw-rate range is

$$\dot{\psi} \in \left[\frac{\underline{a}_y(v_x, \theta, a_x)}{v_x}, \frac{\overline{a}_y(v_x, \theta, a_x)}{v_x}\right], \tag{3.63}$$

which yields bounds on the yaw-error rate:

$$\underline{\dot{e}}_{\psi,k|t} = \frac{\underline{a}_y(\hat{v}_{x,k|t}, \hat{\theta}_{k|t}, \hat{a}_{x,k|t})}{\hat{v}_{x,k|t}} - \hat{v}_{x,k|t}\,\hat{\kappa}_{k|t}, \tag{3.64}$$

$$\overline{\dot{e}}_{\psi,k|t} = \frac{\overline{a}_y(\hat{v}_{x,k|t}, \hat{\theta}_{k|t}, \hat{a}_{x,k|t})}{\hat{v}_{x,k|t}} - \hat{v}_{x,k|t}\,\hat{\kappa}_{k|t}. \tag{3.65}$$

Here the limits are evaluated using the predicted quantities $\hat{v}_{x,k|t}$, $\hat{a}_{x,k|t}$, $\hat{\kappa}_{k|t}$, and $\hat{\theta}_{k|t}$ These bounds are applied directly to the corresponding state component of $x^{\text{lat}}_{k|t}$ along the horizon N.

Side-slip angle constraints. Under the small-angle assumption, the vehicle side-slip angle can be approximated as

$$\beta \simeq \frac{v_y}{v_x} = \frac{1}{v_x}\Big(\dot{e}_y - v_x e_\psi\Big), \tag{3.66}$$

where v_y denotes the lateral velocity in the vehicle frame. Given the admissible interval for the side-slip angle $\beta \in [\,\underline{\beta}, \overline{\beta}\,]$ yields the linear inequalities

$$\underline{\beta} \leq \begin{bmatrix} 0 & \frac{1}{\hat{v}_{x,k|t}} & -1 & 0 & 0 \end{bmatrix} x^{\text{lat}}_{k|t} \leq \overline{\beta}, \tag{3.67}$$

which defines a polytopic constraint on the lateral state vector, where we used the predicted longitudinal speed $\hat{v}_{x,k|t}$, ensuring consistency with the longitudinal planner predictions.

Importantly, these constraints help keep the state trajectory within a region where the underlying modeling assumptions, i.e., linearized vehicle dynamics and small-angle approximations, remain valid.

Overtaking corridor constraints. To guarantee safe lateral interactions with an opponent vehicle during head-to-head racing, the lateral displacement e_y is constrained within a maneuver-specific corridor that represents the available space between track boundaries or between the ego and opponent vehicles. At each prediction step k, the admissible region is defined as

$$\underline{e}_{y,k|t} \leq e_{y,k|t} \leq \overline{e}_{y,k|t}, \quad k \in \mathcal{I}_0^N \tag{3.68}$$

where $\underline{e}_{y,k|t}$ and $\overline{e}_{y,k|t}$ denote the right and left corridor limits, respectively. For single-vehicle laps, these bounds coincide with the track edges sampled along the predicted arc-length trajectory $\hat{s}_{k|t}$:

$$\underline{e}_{y,k|t} = n_r(\hat{s}_{k|t}), \qquad \overline{e}_{y,k|t} = n_\ell(\hat{s}_{k|t}), \quad k \in \mathcal{I}_0^N. \tag{3.69}$$

In contrast, given a forecast of the opponent trajectory $\hat{\boldsymbol{\xi}}_{\text{opp},t} \coloneqq (\hat{\boldsymbol{s}}_{\text{opp},t},\, \hat{\boldsymbol{n}}_{\text{opp},t})$ in the Frenet frame, a lateral safety margin is enforced depending on the selected overtaking direction:

$$\begin{aligned} \underline{e}_{y,k|t} &= \begin{cases} \hat{n}_{\text{opp}}(\hat{s}_{k|t}) + n_{\text{safe}}, & |\hat{s}_{k|t} - \hat{s}_{\text{opp},k|t}| < s_{\text{safe}}, \\ n_r(\hat{s}_{k|t}), & \text{otherwise}, \end{cases} \\ \overline{e}_{y,k|t} &= n_\ell(\hat{s}_{k|t}). \end{aligned} \tag{3.70}$$

for a left overtake, *or*

$$\begin{aligned} \underline{e}_{y,k|t} &= n_r(\hat{s}_{k|t}), \\ \overline{e}_{y,k|t} &= \begin{cases} \hat{n}_{\text{opp}}(\hat{s}_{k|t}) - n_{\text{safe}}, & |\hat{s}_{k|t} - \hat{s}_{\text{opp},k|t}| < s_{\text{safe}}, \\ n_\ell(\hat{s}_{k|t}), & \text{otherwise}. \end{cases} \end{aligned} \tag{3.71}$$

for a right overtake.

In words, these two sets define obstacle-free corridors, obtained by shrinking the track boundaries at those prediction steps where a longitudinal interaction between the two vehicles is expected to occur. Although each corridor is individually convex, their union forms a nonconvex admissible region for the lateral displacement e_y. The overtaking strategy layer resolves this nonconvexity by selecting one active corridor, as discussed in Section 3.6.1.

3.6 Overtaking Strategy Layer

While the longitudinal and lateral MPC-based planners described in the previous section generate dynamically feasible and collision-free trajectories, head-to-head racing requires an additional layer of decision-making: selecting the appropriate overtaking maneuver. To this end, we introduce an *overtaking strategy layer* that orchestrates multiple planner instances in parallel, each corresponding to a distinct maneuver.

The proposed design is motivated by the 2024 Indy Autonomous Challenge Passing Competition at Indianapolis Motor Speedway. This format proceeds in successive head-to-head rounds with assigned roles: one vehicle acts as the *attacker*

(A), the other as the *defender* (D). Each round specifies a round speed that caps the defender's maximum velocity, while this cap is increased in subsequent rounds. Roles alternate between vehicles, and the contest ends at the first failed attack, i.e., if the designated attacker does not complete an overtake within its round, the opponent wins. Defensive behavior is regulated by two key rules: (i) *blocking is prohibited*, meaning that the defender D must not deviate reactively from its racing line to obstruct the attacker A; and (ii) once the attacker reduces the longitudinal gap below a threshold Δ_s (i.e., it acquires the *right-of-way*), the defender must guarantee a lateral clearance Δ_g on the overtaking side. Conversely, the attacker is not allowed to force the defender off its racing line, beyond the granted lateral space Δ_g, during an overtaking maneuver.

The overtaking strategy layer operates above the decoupled longitudinal and lateral planners and determines, at each planning cycle, which maneuver hypothesis to pursue: *follow*, *overtake-left*, or *overtake-right*. By reasoning at the maneuver level rather than over continuous trajectories, this layer provides a tractable mechanism to handle the nonconvexities arising from discrete choices of overtaking side and to accounts for the regulations governing vehicle interactions, by capturing their effect at the maneuver level. This design prevents the overly conservative behaviors that would otherwise result from predicting non-admissible opponent trajectories.

3.6.1 Convex Decomposition into Maneuver-Specific Planners

Collision-avoidance and overtaking constraints are inherently nonconvex, as they exclude regions of the state space corresponding to occupied or unsafe trajectories. In much of the literature, such nonconvexity is accepted and addressed using local nonlinear solvers, at the cost of obtaining only suboptimal solutions. In a racing context, however, selecting a locally suboptimal maneuver, such as committing to the wrong overtaking side, can lead not only to failed passing attempts but also to unsafe vehicle interactions. To address these issues, the proposed strategy layer adopts a convex decomposition of the collision-free area, allowing each maneuver hypothesis to be solved efficiently within a convex domain.

Convexity is achieved by leveraging the decoupled longitudinal and lateral vehicle dynamics introduced in the previous section and by partitioning the nonconvex admissible region into a finite set of maneuver-specific convex subsets. Let the set of maneuver hypotheses be defined as

$$\mathcal{M} := \{\text{follow, left, right}\}. \tag{3.72}$$

Each maneuver $m \in \mathcal{M}$ induces a distinct pair of admissible sets $(\mathcal{F}_m^{\text{long}}, \mathcal{F}_m^{\text{lat}})$ associated with the maneuver-specific joint planning problem Π_m defined in (3.41). Specifically:

- For the longitudinal planner, $\mathcal{F}_m^{\text{long}}$ includes the ACC constraints in (3.57)–(3.58), which enforce a safe longitudinal distance from the defender when $m = \text{follow}$, and are omitted otherwise.

- For the lateral planner, $\mathcal{F}_m^{\text{lat}}$ enforces the overtaking corridor constraints (3.70)–(3.71), corresponding to the selected maneuver, i.e., the left or right overtaking corridor when $m = \text{left}$ or $m = \text{right}$, respectively.

The three pairs of maneuver-specific planners $(\mathcal{P}_m^{\text{long}}, \mathcal{P}_m^{\text{lat}})$ underlying Π_m are solved in parallel at each planning cycle, yielding locally optimal trajectories $(\boldsymbol{z}_m^{\text{long},*}, \boldsymbol{z}_m^{\text{lat},*})$ and corresponding optimal costs $J_m^* := J_m^{\text{long},*} + J_m^{\text{lat},*}$. The overtaking strategy layer then performs the maneuver selection as

$$m^* = \arg\min_{m \in \mathcal{M}} J_m^*, \tag{3.73}$$

where m^* denotes the maneuver yielding the trajectory with the lowest predicted cost, while satisfying dynamic feasibility and regulation constraints.

This structure effectively decomposes the original nonconvex trajectory-planning problem into a finite set of convex subproblems, each corresponding to a locally convex region of the configuration space. For each maneuver $m \in \mathcal{M}$, the associated admissible sets $\mathcal{F}_m^{\text{long}}$ and $\mathcal{F}_m^{\text{lat}}$ are parameterized by regulation-consistent predictions of the defender's motion, as detailed in the following subsection.

3.6.2 Regulation-Consistent Constraint Generation

Let us focus on the overtaking phase, where the ego vehicle acts as the *attacker* and the opponent as the *defender*. Each maneuver-specific problem Π_m in (3.41) requires a prediction of the opponent's motion to impose tailgating or overtaking constraints. For tractability, the longitudinal trajectories of both vehicles are assumed to be fixed during the maneuver, allowing the acquisition of right-of-way (RoW) to be predicted deterministically from the known longitudinal plans and enabling the adjustment of the opponent's lateral prediction accordingly.

Let the ego vehicle's predicted longitudinal trajectory be $\hat{\boldsymbol{s}}_{\text{ego},t} := \{\hat{s}_{k|t}\}_{k=0}^{N}$, and let the opponent's open-loop prediction in Frenet coordinates be $\hat{\boldsymbol{\xi}}_{\text{opp},t} := (\hat{\boldsymbol{s}}_{\text{opp},t}, \hat{\boldsymbol{n}}_{\text{opp},t})$. The regulation-consistent forecast of the defender's motion, conditional on the ego maneuver $m \in \mathcal{M}$, is obtained by adjusting the open-loop opponent prediction through the mapping

$$\tilde{\boldsymbol{\xi}}_{\text{opp},t}^{m} := \mathcal{R}_m\left(\hat{\boldsymbol{s}}_{\text{ego},t}, \hat{\boldsymbol{\xi}}_{\text{opp},t}\right), \tag{3.74}$$

where $\mathcal{R}_m : \mathbb{R}^{N+1} \times \mathbb{R}^{2(N+1)} \to \mathbb{R}^{2(N+1)}$ is the *regulation-consistent projection operator*. The operator $\mathcal{R}_m$ enforces the asymmetric interaction prescribed by racing regulations: if the attacker is predicted to acquire the right-of-way, the defender's lateral trajectory is modified to yield a lateral clearance Δ_{g} on the overtaking side m; otherwise, the defender is assumed to maintain its nominal lateral position. In the present implementation, this is achieved by clamping the predicted opponent lateral coordinate $\hat{\boldsymbol{n}}_{\text{opp},t}$ to ensure that the granted clearance Δ_{g} is not violated at those prediction steps where the attacker holds RoW. Alternative definitions of $\mathcal{R}_m$ may be employed to obtain more realistic opponent trajectories without altering the general framework.

Remark 3 (Computation of the Opponent Prediction). The computation of the regulation-consistent opponent prediction $\tilde{\boldsymbol{\xi}}_{\text{opp},t}^{m}$ requires additional considerations to ensure physical and regulatory feasibility. In particular, it involves verifying the availability of lateral space at the moment the right-of-way is acquired, as well as other interaction-dependent conditions. These aspects are omitted here for brevity; further implementation details are provided in [73].

The adjusted opponent prediction $\tilde{\boldsymbol{\xi}}^{m}_{\text{opp},t} := \{(\hat{s}_{\text{opp},k|t},\ \tilde{n}_{\text{opp},k|t})\}_{k=0}^{N}$ is then used to construct the overtaking-corridor and longitudinal separation constraints, yielding maneuver-dependent admissible sets $\mathcal{F}_m^{\text{long}}$ and $\mathcal{F}_m^{\text{lat}}$ that embed regulation-consistent interactions between the attacker and defender.

3.6.3 Game-Theoretic Interpretation

The structure described above admits a natural game-theoretic interpretation. The interaction between the attacker (A) and defender (D) can be interpreted as a Stackelberg game, where the attacker acts as the *leader* and the defender as the *follower*. The attacker optimizes its decision variables $\boldsymbol{z}_{\text{A},t}$ (both longitudinal and lateral, omitted in the rest of this section), and maneuver $m \in \mathcal{M}$ while anticipating the defender's reaction to its predicted motion. The two agents are coupled through feasibility constraints that encode collision avoidance and regulation-consistent behavior.

Formally, the ideal interaction can be written as a bilevel program:

$$\begin{aligned} \min_{m\in\mathcal{M},\, \boldsymbol{z}_{\text{A},t}} \quad & J_{\text{A},t}\Big(\boldsymbol{z}_{\text{A},t}\Big) \\ \text{s.t.} \quad & \boldsymbol{z}_{\text{A},t} \in \mathcal{F}_t^{\text{A}}\Big(m,\, \boldsymbol{\xi}^{\star}_{\text{D},t}\Big), \\ & \boldsymbol{\xi}^{\star}_{\text{D},t} \in \text{BR}_{\text{D}}\Big(\boldsymbol{z}_{\text{A},t}\Big), \end{aligned} \tag{3.75}$$

where $\text{BR}_{\text{D}}(\cdot)$ denotes the defender's best-response mapping. The exact structure of the defender problem is not known, but it is coupled to the attacker trajectory through regulation constraints.

To make the problem tractable, we approximate the follower's best response with a regulation-consistent prediction of the defender trajectory

$$\tilde{\boldsymbol{\xi}}^{m}_{\text{D},t} \;=\; \mathcal{R}_m\Big(\hat{\boldsymbol{s}}_{\text{A},t};\, \hat{\boldsymbol{\xi}}_{\text{D},t}\Big), \tag{3.76}$$

where $\hat{\boldsymbol{s}}_{\text{A},t}$ is the attacker longitudinal plan and $\hat{\boldsymbol{\xi}}_{\text{D},t}$ is a open-loop prediction of the defender. This substitution transforms (3.75) into a single-level approximation:

$$\min_{m\in\mathcal{M},\, \boldsymbol{z}_{\text{A},t}} \; J_{\text{A},t}\Big(m, \boldsymbol{z}_{\text{A},t}\Big) \quad \text{s.t.} \quad \boldsymbol{z}_{\text{A},t} \in \mathcal{F}_{\text{A},t}\Big(m,\, \tilde{\boldsymbol{\xi}}^{m}_{\text{D},t}\Big), \tag{3.77}$$

which remains convex for fixed m but discrete in the maneuver selection. The three problem realizations for $m \in \mathcal{M}$ are solved in parallel and the minimum-cost solution is selected as the optimal maneuver at step t, which is equivalent to the maneuver selection (3.73).

This formulation preserves the hierarchical reasoning of the Stackelberg game while ensuring real-time implementability through the regulation-consistent approximation of the defender response.

3.7 Experimental Validation

The proposed motion planning and control framework was validated through full-scale experiments conducted during the international autonomous racing competitions A2RL and IAC. The A2RL took place in April 2024 at the Yas Marina Circuit (Abu Dhabi, UAE), a 5.28 km Formula 1-grade road course characterized by high curvature sections and multiple low-speed corners, providing an ideal benchmark for testing planning and control under high longitudinal and lateral accelerations. The IAC event was held at the Indianapolis Motor Speedway (Indiana, USA) in September 2024, featuring a 4.02 km oval track with long straights and banked turns of 9.2°, enabling validation at top speeds approaching 290 km/h during head-to-head competitive racing.[3] Official recordings of both events are available online [74], [75].

Both competitions offered complementary conditions to assess the performance, robustness, and real-time feasibility of the proposed planning and control stack. In this section, we first present results from the A2RL *Time Trial* sessions, focusing on single-vehicle operation on a road-course layout. These experiments validate the longitudinal and lateral controllers as well as the corresponding predictive planners. Subsequently, we report results from the IAC *Passing Competition*, in which the same framework was use to perform longitudinal cruise control and overtaking maneuvers against an opponent vehicle. Finally, an analysis of solver timing and computational performance is provided to assess the real-time capabilities of the proposed implementation.

3.7.1 Experimental Setup

The experiments were conducted using the UNIMORE Racing full-scale autonomous vehicles in the two competition series. In the road-course single-vehicle scenario of the A2RL, the vehicle was based on the Dallara EAV-24 chassis (derived from Super Formula), while in the head-to-head high-speed scenario of the IAC, the platform was the Dallara AV-24 chassis (derived from the Indy NXT/IndyLights specification). Both vehicles were equipped with a sensor suite including radar, camera, LiDAR and RTK-GNSS, as well as drive-by-wire actuators for steering, braking and throttle. As computational platforms, the AV-24 (IAC) features an Intel® Xeon® D-2166NT CPU, 128 GB of RAM, and an NVIDIA Quadro RTX A5000 GPU. The EAV-24 (A2RL) is equipped with an AMD EPYC™ 7003 series "Milan" processor (32 cores and 64 threads) and an NVIDIA RTX 6000 GPU.

The motion planning and control modules were integrated within the UNIMORE Racing autonomy stack, which comprises all core components required for full autonomous operation in racing competitions, including perception, localization, mission-planning, and vehicle stability controllers. All modules were implemented in `C++`, with the low-level controllers executing at $T_s^{\mathrm{c}} = 10\,\mathrm{ms}$ and the planners operating at $T_s^{\mathrm{p}} = 50\,\mathrm{ms}$. Optimization problems were formulated using `acados` [76] and solved with `HPIPM` [68]. The architecture was deployed through a ROS 2-like

[3]Over the course of the IAC competition and test sessions, the UNIMORE Racing vehicle completed more than 1,500 mi (≈ 2,500 km) of fully autonomous operation.

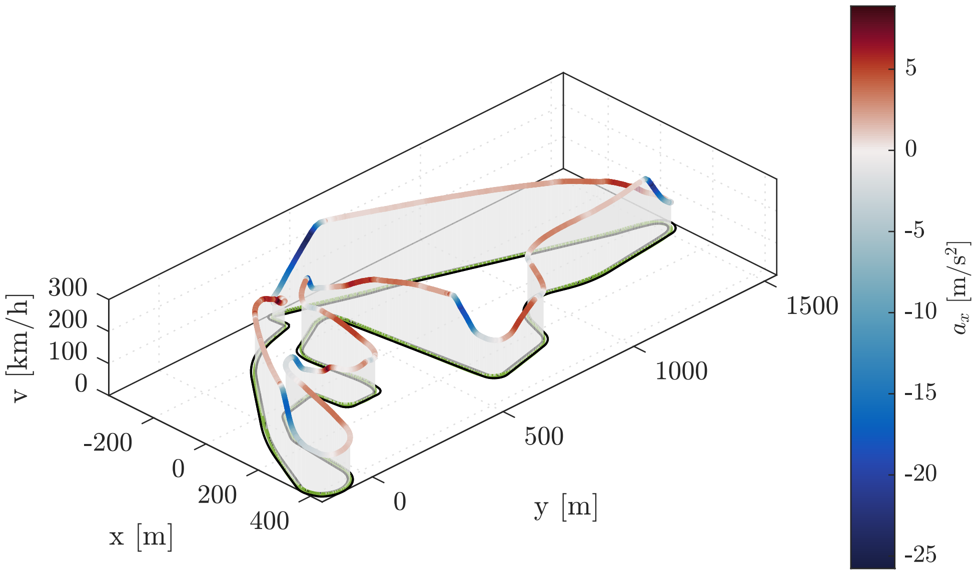


Figure 3.3. Speed and acceleration profile of the UNIMORE Racing EAV-24 during the Time-Trial event at the A2RL 2024, Yas Marina Circuit. The 3D plot shows the track layout on the xy plane and vehicle speed along the vertical axis, with line color denoting longitudinal acceleration.

interface allowing synchronized logging and real-time visualization of trajectories, constraints, and diagnostics metrics.

3.7.2 Path-Tracking Validation in Single-Vehicle Scenario

The single-vehicle experiments were used to validate the path-tracking capabilities of the proposed motion planning and control framework under high-curvature and high-accelerations conditions. The objective was to assess the consistency between planned and executed trajectories, and to verify that both the lateral and longitudinal subsystems maintain constraint satisfaction while exploiting the vehicle's dynamic limits.

Fig. 3.3 illustrates the overall vehicle motion along the track, where elevation encodes speed and color represents longitudinal acceleration. The distribution of lateral and longitudinal accelerations in Fig. 3.4 confirms that the achieved operating points remain within the estimated GGV envelope, demonstrating the dynamic feasibility of the generated trajectories. The subsequent analysis focuses on the performance of the lateral and longitudinal planning and control modules.

Lateral Planning and Control Validation

Fig. 3.5 illustrates the performance of the lateral control loop during a representative lap. The steady-state lateral error command $e_{y,\mathrm{ss}}$ generated by the planner is provided to the feedforward component of the controller, while the LQR term performs state-feedback regulation around the origin. The feedforward component therefore acts both as curvature compensation and as a reference adjustment, enabling the controller to track the steady-state offset prescribed by the planner. It is worth noting that $e_{y,\mathrm{ss}}$

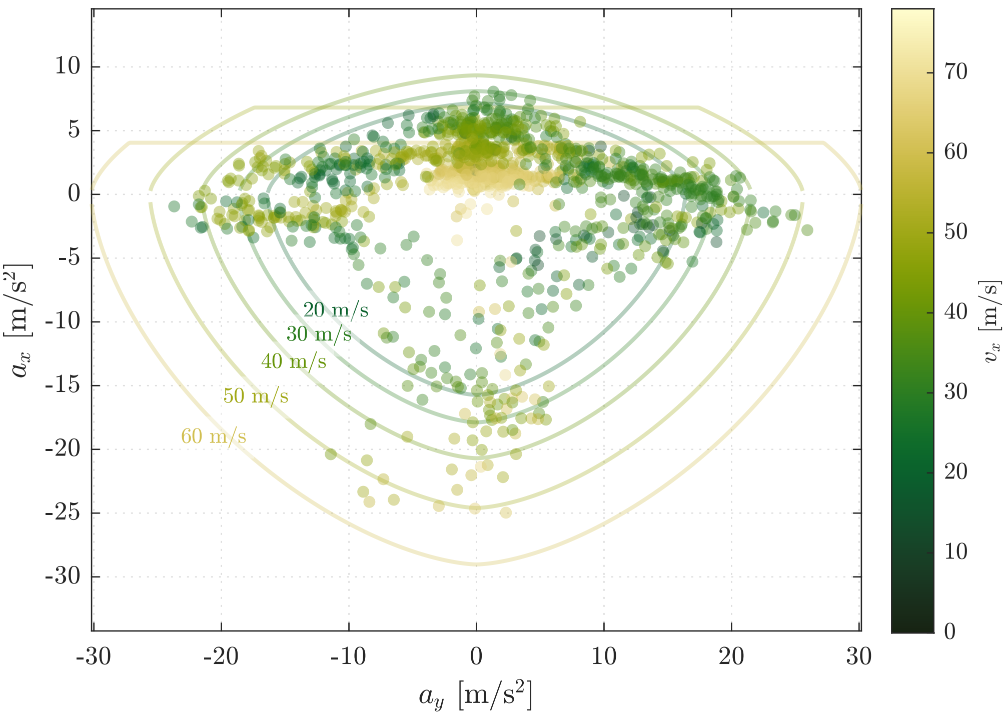


Figure 3.4. Measured longitudinal and lateral accelerations of the UNIMORE Racing EAV-24 during the Time-Trial event at the A2RL 2024, Yas Marina Circuit. Each marker represents a sampled vehicle state, color-coded by speed, while the overlaid iso-speed slices of the GGV envelope illustrate the theoretical acceleration limits at selected velocities.

is not necessarily zero: although this may appear counterintuitive with respect to path-tracking objectives, the planner accounts for the closed-loop vehicle-controller dynamics within its prediction model and uses $e_{y,\mathrm{ss}}$ to (i) anticipate the upcoming path curvature and (ii) to enforce handling and actuator constraints satisfaction (at the planner sampling rate). Since the controller operates at a higher frequency, small disturbances or state variations may occur between planner updates, for which the feedback term ensures high-rate corrections. These corrective adjustments may locally violate the constraints enforced by the planner; however, in practice, constraint tightening provides robustness against such small deviations, which are then reabsorbed at the next planning update.

The corresponding performance of the lateral motion planner is shown in Fig. 3.6. The top panel reports the predicted lateral deviation along the horizon together with the admissible region imposed by the track boundaries. The bottom panel compares the measured yaw rate with the the admissible limits derived from the GGV envelope, as in (3.64)-(3.65).

Longitudinal Planning and Control Validation

The performance of the longitudinal control loop is illustrated in Fig. 3.7. The first panel compares the measured longitudinal acceleration a_x with the planner target $a_{x,\mathrm{tar}}$, showing accurate tracking across both acceleration and braking phases. The following panels report the throttle and brake actuation signals, decomposed into

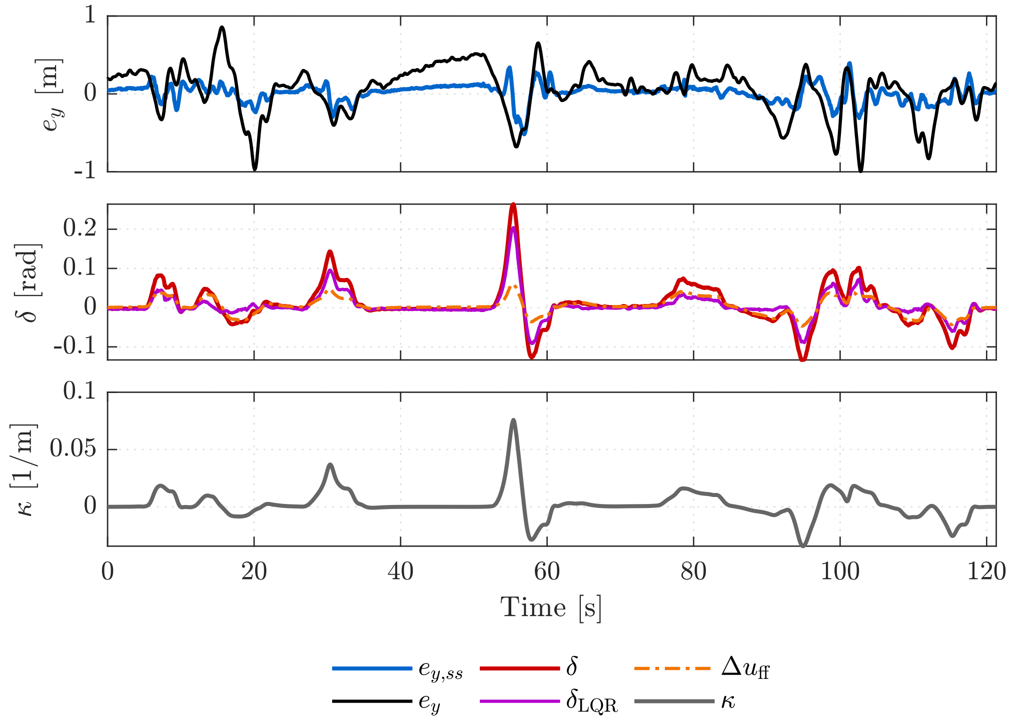


Figure 3.5. Lateral controller behavior during the Time-Trial event at the A2RL 2024, Yas Marina Circuit. The top panel shows the lateral tracking error and the lateral offset command $e_{y,\mathrm{ss}}$ generated by the planner. The middle panel reports the steering command, decomposed into feedback (LQR) and feedforward contributions. The bottom panel shows the curvature of the reference racing line.

feedforward and feedback contributions. The feedforward component compensates for the nominal powertrain and braking dynamics, while the PID feedback term corrects residual tracking errors and accounts for unmodeled effects such as drivetrain delays and temperature-dependent braking performance. As shown in the figure, the magnitude of the PIDs contribution remains only a fraction of the feedforward term throughout the lap, for both throttle and brake commands. This confirms that the controller operates predominantly in feedforward mode, with the feedback action providing local correction and robustness to disturbances. Overall, the controller achieves smooth and continuous transitions between traction and braking, ensuring stable vehicle behavior at high speed and under large acceleration and braking demand.

The longitudinal planner performance is shown in Fig. 3.8. The top plot reports the offline speed profile together with the measured speed and the upper bound derived from the GGV envelope as in (3.55). The bottom plot shows the corresponding longitudinal acceleration trajectories and admissible limits, computed as in (3.52); the planner maintains the longitudinal acceleration within admissible bounds, including aggressive braking and corner-exit acceleration.

Table 3.1 summarizes the main performance indicators achieved during the reference lap. The vehicle reached a top speed of approximately $254\,\mathrm{km\,h^{-1}}$ and lateral accelerations up to $26\,\mathrm{m\,s^{-2}}$, while maintaining sub-meter lateral tracking accuracy.

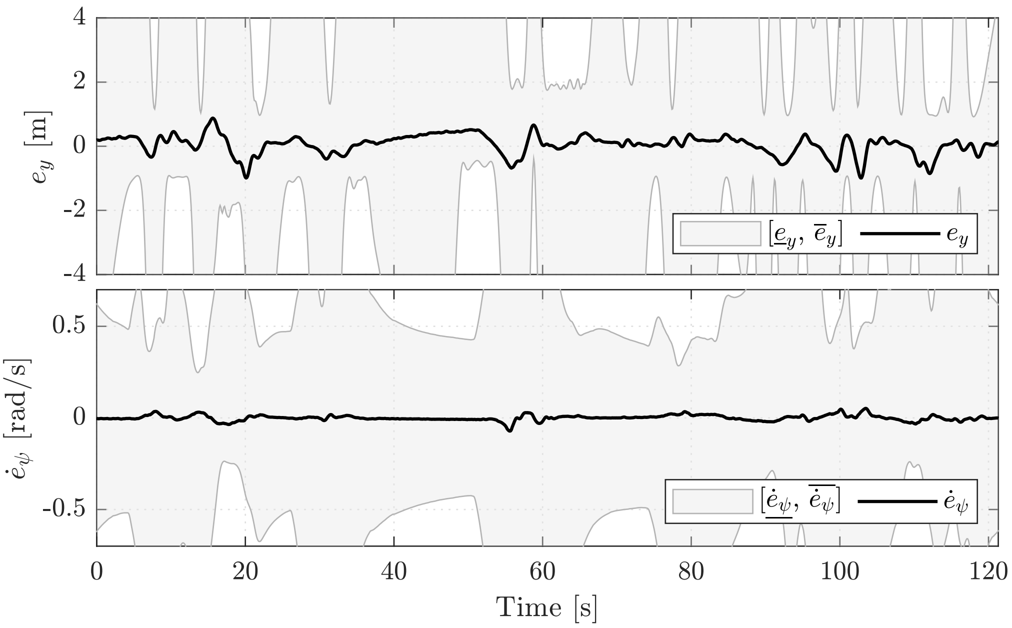


Figure 3.6. Lateral planner behavior during the Time-Trial event at the A2RL 2024, Yas Marina Circuit. The top panel shows the lateral error and the admissible region delimited by the track boundaries. The bottom panel includes the corresponding yaw-rate profile together with the admissible range derived from the GGV envelope.

For completeness, Fig. 3.9 reports the vehicle motion during the IAC Time Trial sessions at the Indianapolis Motor Speedway. The color-coded 3D representation of the oval track shows the speed and longitudinal acceleration profiles, illustrating the characteristic high-speed and low-curvature nature of the circuit. Table 3.2 summarizes the main performance indicators achieved in the best lap. The vehicle reached a top speed of nearly $286\,\mathrm{km\,h^{-1}}$ while maintaining lateral path tracking error below $0.9\,\mathrm{m}$, confirming the consistency of the proposed planning and control framework under the diverse dynamic conditions, such as those of the IMS track. The higher RMS lateral error observed with respect to the Yas Marina configuration is attributed to increased modeling uncertainty caused by banking-angle variations and the asymmetric mechanical setup of the vehicle, which lead to discrepancies between the predicted and actual steady-state responses.

Table 3.1. Vehicle performance metrics from the UNIMORE Racing best lap during the Time-Trial event at the A2RL 2024, Yas Marina Circuit (lap time 2:01.314).

Metric	**Value**	**Unit**	**Metric**	**Value**	**Unit**
$v_{\max}$	254.07	$\mathrm{km\,h^{-1}}$	e_y^{rms}	0.29	m
$a_{x,\max}$	8.06	$\mathrm{m\,s^{-2}}$	$e_y^{\max}$	0.99	m
$a_{x,\min}$	−24.97	$\mathrm{m\,s^{-2}}$	e_ψ^{rms}	0.47	°
$\lvert a_y\rvert_{\max}$	25.94	$\mathrm{m\,s^{-2}}$	$e_\psi^{\max}$	2.32	°

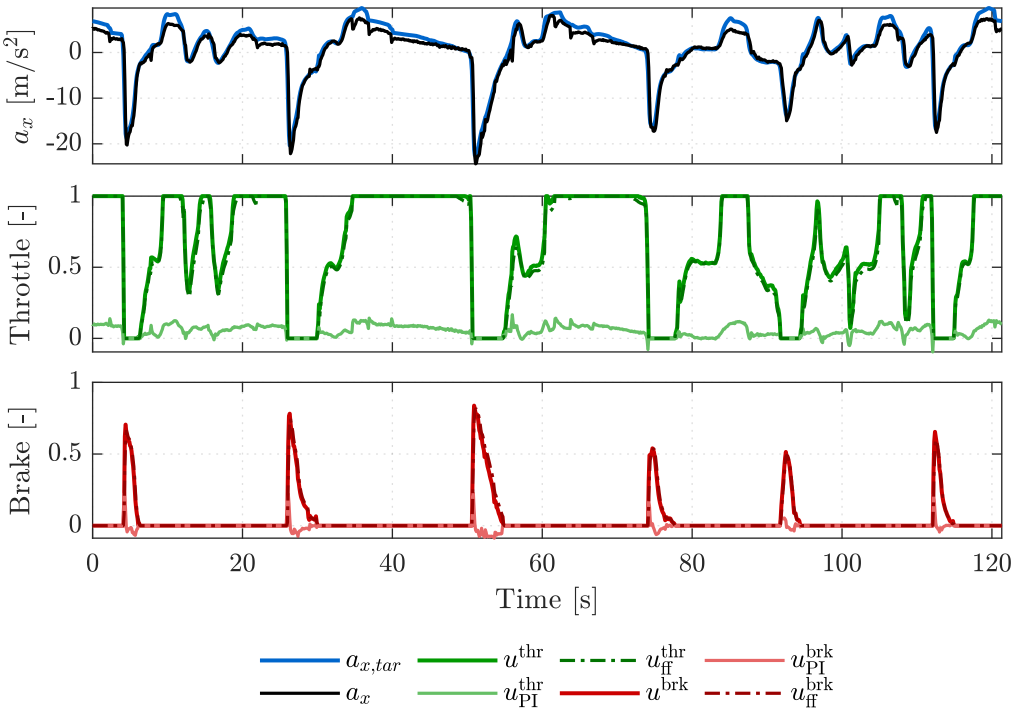


Figure 3.7. Longitudinal control data from the UNIMORE Racing EAV-24 during the Time-Trial event at the A2RL 2024, Yas Marina Circuit. The upper panel shows tracking of the planner target acceleration. The lower panels report the throttle and brake commands: dashed lines the feedforward (FF) contributions, and dotted lines the PID corrective terms.

3.7.3 Head-to-Head Racing Experiments

The head-to-head experiments were carried out during the IAC 2024 Passing Competition at the Indianapolis Motor Speedway, involving direct interaction with an opponent vehicle. In this configuration, the overtaking strategy layer coordinates the three maneuver-specific planner instances (*follow*, *overtake-left*, and *overtake-right*) whose outputs are then evaluated and selected as described in Section 3.6. This setup enabled the longitudinal planner to perform adaptive cruise control (ACC) when following an opponent and to transition seamlessly to an overtaking maneuver once permission was granted by the race-control interface. The lateral planner simultaneously generated dynamically feasible lateral offset trajectories within the selected overtaking corridor while respecting track boundaries and vehicle handling constraints.

Table 3.2. Vehicle performance metrics from the UNIMORE Racing best lap during the Time-Trial event at the IAC 2024, Indianapolis Motor Speedway (lap time 54.932).

Metric	Val.	Unit	Metric	Val.	Unit
$v_{\max}$	286.39	$\mathrm{km\,h^{-1}}$	e_y^{rms}	0.45	m
$a_{x,\max}$	1.53	$\mathrm{m\,s^{-2}}$	$e_y^{\max}$	0.86	m
$a_{x,\min}$	−5.56	$\mathrm{m\,s^{-2}}$	e_ψ^{rms}	0.31	°
$\lvert a_y\rvert_{\max}$	24.22	$\mathrm{m\,s^{-2}}$	$e_\psi^{\max}$	0.91	°

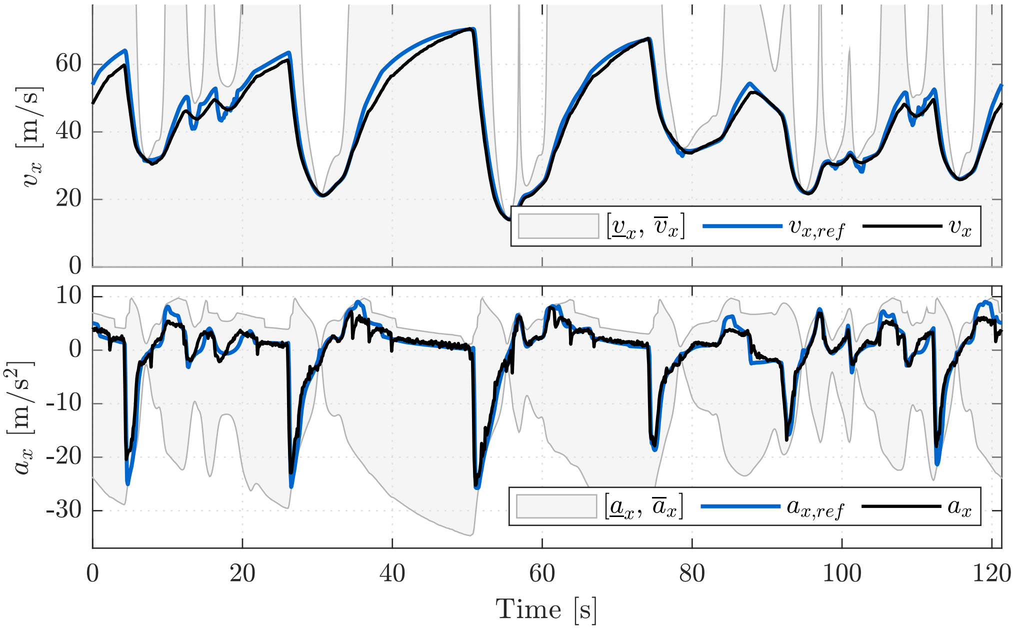


Figure 3.8. Longitudinal planner data from the UNIMORE Racing EAV-24 during the Time-Trial event at the A2RL 2024, Yas Marina Circuit. The upper panel reports velocity tracking, and the lower panel shows the corresponding acceleration. Black lines denote measured quantities, dashed blue lines the offline reference speed and acceleration profiles, and the gray shaded area the admissible ranges derived from the GGV envelope.

Fig. 3.10 illustrates the adaptive cruise control functionality of the longitudinal planner and the maneuver selection logic of the overtaking strategy. At t_0, the ego vehicle detects the opponent and adapts its speed to maintain the desired time gap. The time-gap reference is scheduled as a function of speed so that, at each defender round speed, the corresponding spatial separation equals approximately 30 m, as required by the competition rules. It can be observed that the time-gap distance $v_x \cdot t_{\text{gap}}$ closely approaches the admissible upper bound set by the opponent distance, without violating it, confirming proper constraint enforcement by the planner. At t_1, the ego receives permission to overtake, triggering an internal logic that reduces the time-gap value to close the distance (tailgating phase) and prepare for overtake. Finally, at t_2, the overtaking strategy layer activates the *overtake-left* maneuver, switching to the corresponding planner in which the longitudinal-distance constraint is not enforced, thereby enabling acceleration. The behavior observed in the plots confirms the correct coordination of the high-level maneuver logic and the seamless transition between the *follow* and *overtake* modes.

Fig. 3.11 shows an overtaking maneuver performed in the final match of the Passing Competition, with the defender's round speed set to 155 mph. The eight frames illustrate the progression from the approach on the main straight, through the side-by-side alignment in Turns 1 and 2, to the completion of the pass on the back straight. The shaded overtaking corridor highlights how the lateral planner is constrained to remain on the outside line (*right-overtake*), thereby deviating from the nominal racing line to safely complete the maneuver. A video recording showing the onboard cameras view and telemetry is available at [77].

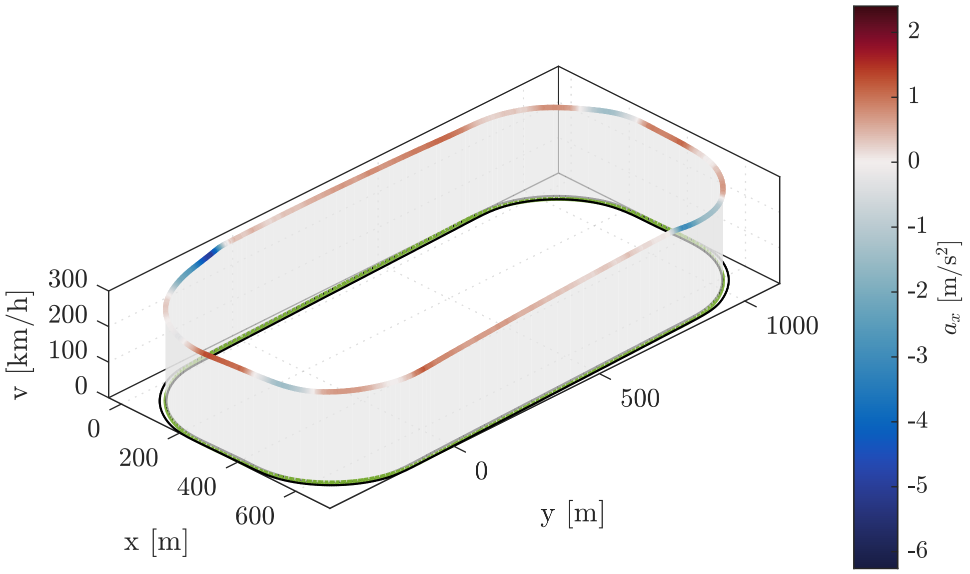


Figure 3.9. Speed and acceleration profile of the UNIMORE Racing AV-24 during the Time-Trial event at the IAC 2024, Indianapolis Motor Speedway. The 3D plot shows the track layout on the xy plane and vehicle speed along the vertical axis, with line color denoting longitudinal acceleration.

3.7.4 Solver Timing and Real-Time Performance

The computational performance of the proposed planning architecture was assessed using the solver statistics collected during the A2RL 2024 Time-Trial sessions and the IAC 2024 Passing Competition. Figures 3.12a and 3.12b illustrate the distribution of solver times (τ) and QP iterations (ν) for the longitudinal and lateral planners, while Tables 3.3 and 3.4 report the corresponding summary statistics. In both cases, the optimization problems were solved well within the nominal planner update period of 50 ms, confirming the real-time feasibility of the implementation. Average computation times remained below 5 ms in A2RL and 11 ms in IAC, corresponding to less than 10% and 25% of the available planning cycle, respectively. Since the iteration counts are comparable across both datasets, the higher solver times observed in IAC are primarily attributed to differences in the onboard computing hardware specifications rather than to difference of the problem complexity in the specific racing scenario.

Overall, the results confirm the deterministic and efficient real-time performance of both planners under diverse dynamic conditions, including single-vehicle racing in road-course and head-to-head racing in oval-track.

These performance levels are achieved thanks to the formulation of both planners as convex quadratic programs, at the cost of adopting decoupled dynamics and the modeling approximations discussed previously. This trade-off effectively enables real-time optimization at the planning level without sacrificing the prediction horizon length, while retaining sufficient fidelity for high-speed autonomous driving.

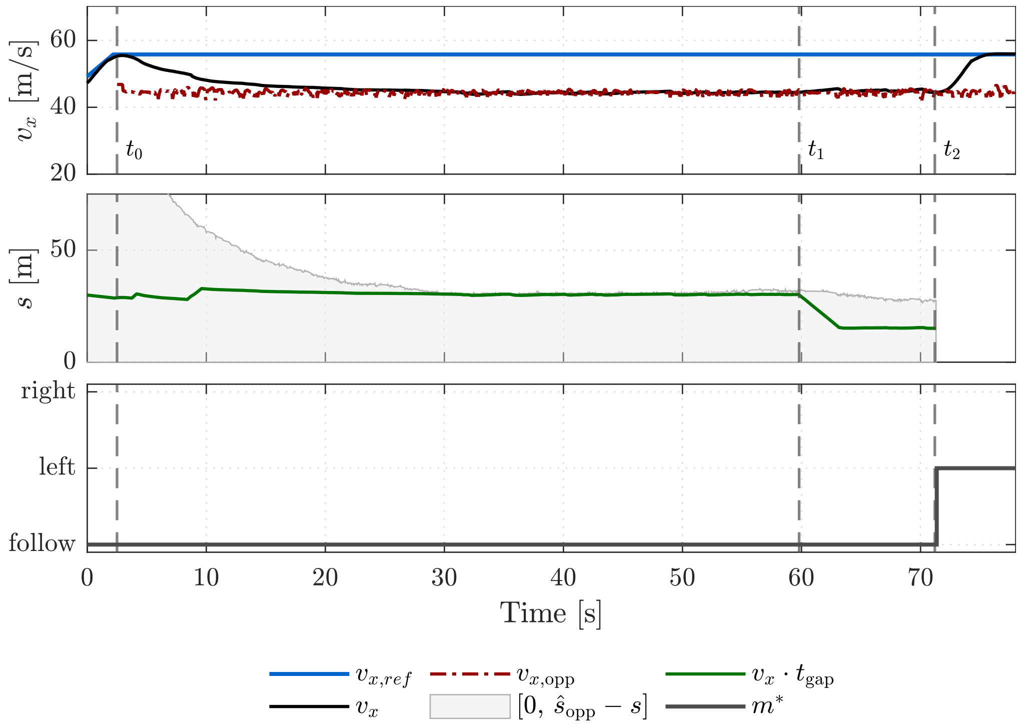


Figure 3.10. Adaptive cruise control functionality of the longitudinal planner during the IAC 2024 Passing Competition, Indianapolis Motor Speedway. The top panel shows the target, measured, and opponent speeds. The middle panel reports the desired time-gap distance and the admissible region upper-bounded by the opponent's longitudinal distance. The bottom panel shows the maneuver selector, indicating the active mode among *follow*, *overtake-left*, and *overtake-right*. At t_0, the opponent is detected and the planner adapts the speed to maintain the desired time gap. At t_1, the overtake permission is granted and the time-gap value is reduced accordingly. At t_2, the overtaking strategy layer selects the left-overtake maneuver, removing the longitudinal-distance bound and enabling acceleration.

Table 3.3. Solver timing (τ) and QP iteration (ν) statistics collected during the A2RL 2024 Time-Trial sessions at the Yas Marina Circuit.

Metric	τ^{long} [ms]	τ^{lat} [ms]	ν^{long}	ν^{lat}
Median	2.12	3.75	11	13
95th Percentile	2.88	4.83	14	13
Max	4.09	5.97	19	14
Min	1.18	3.19	6	13
Mean	2.20	3.87	11	13

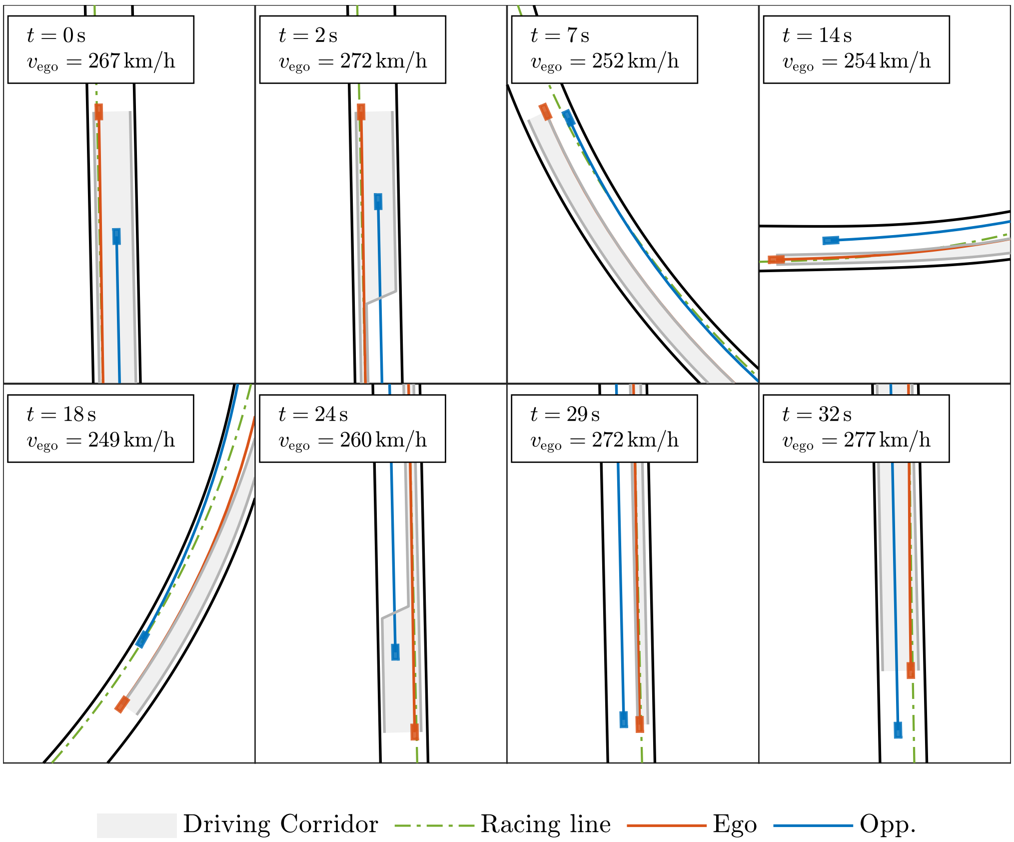
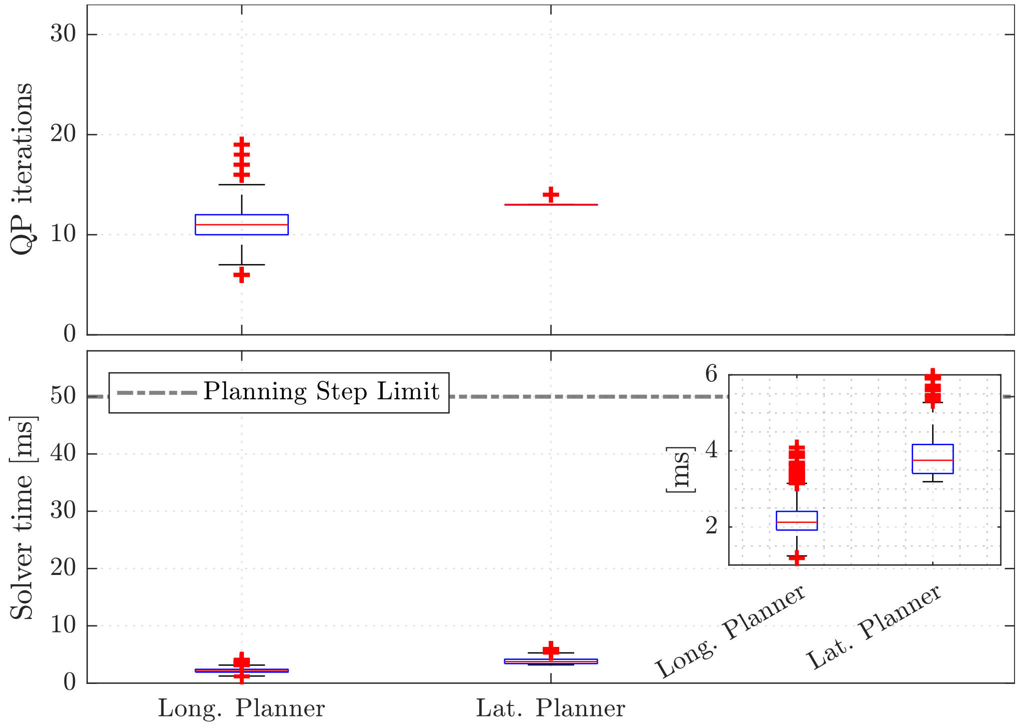


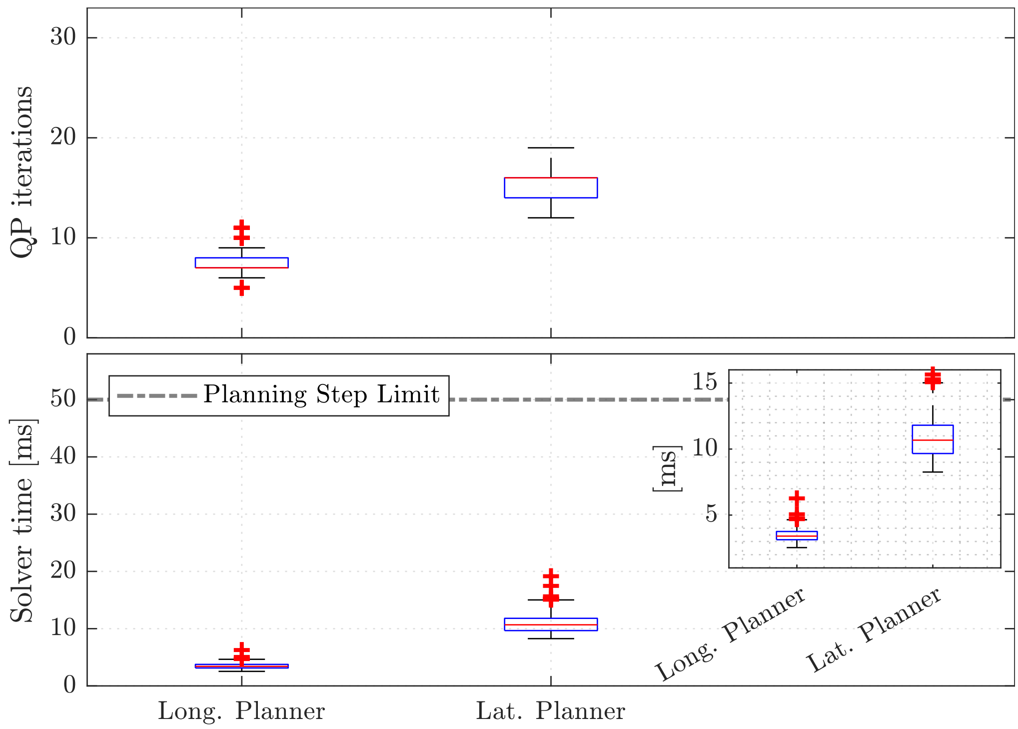
Figure 3.11. Sequence of snapshots illustrating the overtaking maneuver performed by the UNIMORE Racing vehicle during the 155 mph ($\approx 250\,\mathrm{km/h}$) round in the final of the IAC 2024 Passing Competition at the Indianapolis Motor Speedway. Orange and blue rectangles represent the ego (attacking) and opponent (defender) vehicles, respectively, while the shaded gray area denotes the overtaking corridor constraints for the lateral planner. The eight panels show the evolution of the maneuver from the approach on the main straight, through the alignment in Turns 1 and 2, to the completion of the overtake on the back straight.

Table 3.4. Solver timing (τ) and QP iteration (ν) statistics collected during the IAC 2024 Passing Competition at the Indianapolis Motor Speedway.

Metric	τ^{long} [ms]	τ^{lat} [ms]	ν^{long}	ν^{lat}
Median	3.41	10.67	7	16
95th Percentile	4.14	13.64	9	17
Max	6.26	19.16	11	19
Min	2.54	8.26	5	12
Mean	3.44	10.91	8	15



(a) A2RL 2024 Time-Trial, Yas Marina Circuit.



(b) IAC 2024 Passing Competition, Indianapolis Motor Speedway.

Figure 3.12. Box plots of solver execution times and QP iterations for the longitudinal and lateral planners. The dashed gray line marks the 50 ms real-time threshold, while the inset in each plot provides a zoomed view of the nominal operating region, showing solver times well below the limit.

3.8 Summary and Discussion

This chapter presented the design of a motion planning and control framework that brings optimization into the planning layer of full-scale autonomous racing. The proposed architecture combines decoupled longitudinal and lateral planners with dedicated feedback–feedforward controllers, enabling predictive, dynamically feasible trajectory generation in real time. A maneuver-level overtaking strategy coordinates multiple planner instances to effectively address the inherent nonconvexity of overtaking constraints, while also accounting for the distinct, regulation-consistent opponent responses associated with each maneuver. Although the strategic reasoning is abstracted to the maneuver-selection layer, this structure overcomes the limitations of fixed opponent predictions and provides a practical mechanism for incorporating interactive behavior within a real-time optimization framework, deployable on full-size autonomous racecars. Experimental validation in real-world autonomous racing competitions confirmed the effectiveness of the proposed approach, which achieved podium results in both single-vehicle and head-to-head racing events.

Despite its demonstrated effectiveness, the proposed approach relies on modeling simplifications that introduce inherent limitations. The decoupling of longitudinal and lateral dynamics, together with the use of linearized tire-force models, enabled the design of a planning layer that already accounts for dynamic feasibility. This design choice addresses a common drawback of hierarchical architectures, in which the low-level controller must track reference trajectories that may be physically inadmissible, potentially leading to instability or recursive infeasibility. However, since racing inherently involves strong coupling between longitudinal and lateral dynamics, the proposed formulation relies on constraint tightening to prevent the system from operating in regions where these assumptions no longer hold. As a result, while the planner achieves high-performance driving close to the vehicle's limit of handling, its accuracy and performance may degrade when operating *at* the limit, where tire behavior becomes highly nonlinear.

The proposed overtaking strategy enforces regulation consistency in the opponent predictions used to derive the driving-corridor constraints, providing an approximate form of interaction-aware planning. This formulation, however, relies on the assumption that the opponent vehicle strictly complies with racing regulations at all times. If this assumption is violated, safety can no longer be guaranteed, as the derived separation constraints become invalid. While such behavior would transfer responsibility to the rule-breaking agent, additional safety layers capable of detecting and reacting to clear violations are necessary to prevent avoidable collisions in practice.

Finally, extending the proposed framework to multi-vehicle scenarios poses additional scalability challenges. In principle, the maneuver set grows combinatorially with the number of opponents, as each possible combination of overtaking sides must be considered. Such growth may remain tractable for a limited number of agents by exploiting parallel evaluation and branch-and-cut techniques to discard clearly infeasible maneuver hypotheses. In practice, however, pre-selecting a consistent set of few plausible maneuvers, potentially using data-driven priors, represents the most viable solution for real-time implementation.

Chapter 4

Predictive Motion Planning under Racing Regulations

4.1 Introduction

This chapter addresses Research Question 2, which investigates how competition regulations governing vehicle interactions can be systematically incorporated into optimization-based predictive planning frameworks for head-to-head autonomous racing. The objective is to move beyond conservative motion planners that merely ensure safety, toward optimization-based formulations that explicitly encode, and strategically exploit, racing rules to produce competitive yet rule-compliant behaviors.

Racing rules introduce logical dependencies between agents, such as right-of-way or yielding obligations, that define which vehicle must give space during an overtaking maneuver. Encoding such logic within optimization-based planners requires bridging discrete decisions and continuous vehicle dynamics. Moreover, the ability to reason over how an opponent's motion is constrained by these same rules allows an autonomous agent to plan more assertive yet regulation-compliant maneuvers.

While Chapter 3 introduced a maneuver-level decision-making scheme that implicitly accounted for opponents' regulatory obligations, the underlying optimization problems remained regulation-agnostic. In contrast, this chapter formalizes the integration of racing rules into the predictive planning framework itself, introducing explicit formulations for *regulation compliance*, through rule encoding in the problem constraints, and *regulation awareness*, through a game-theoretic planning layer.

Motivated by real-world autonomous racing competitions, the focus is on the enforcement of *right-of-way* rules, which restrict the lateral motion of the leading vehicle during overtaking maneuvers. To formally encode these rules within an optimization-based planner, the proposed framework adopts the Mixed Logical Dynamical (MLD) formulation [47], which allows logical decisions (i.e., binary variables) and continuous vehicle dynamics to be integrated into a unified predictive control problem.

The contributions of this chapter are threefold:

- a formal mathematical description of the overtaking regulations adopted in autonomous racing, capturing how right-of-way and yielding obligations constrain vehicle motion;

- the development of a *Regulation-Compliant MPC* (RC-MPC) formulation based on the MLD framework, which enforces overtaking rules within the predictive control problem;
- the design of a *Regulation-Aware Game-Theoretic Planner* (RA-GTP) that reasons over opponents' regulation-constrained trajectories by solving a Generalized Nash Equilibrium Problem (GNEP) via an Iterative Best-Response (IBR) scheme.

Simulation results show that the proposed RA-GTP outperforms baseline approaches, which relies on fixed opponent predictions, and captures the asymmetric responsibilities imposed by racing regulations, promoting natural and realistic interactions between competitors.

The remainder of the chapter is organized as follows. Section 4.2 reviews related work on multi-agent planning, game-theoretic formulations, and regulation modeling. Section 4.3 defines the problem setting and the right-of-way regulation. Section 4.5 presents the Regulation-Compliant MPC formulation, and Section 4.6 details the Regulation-Aware Game-Theoretic Planner. Simulation results are reported in Section 4.7, while conclusions are summarized in Section 4.8.

The content of this chapter builds upon the work

- *"Regulation-Aware Game-Theoretic Motion Planning for Autonomous Racing,"* by F. Prignoli, F. Borrelli, P. Falcone, and M. Pustilnik, presented at the *IEEE Intelligent Transportation Systems Conference (ITSC 2025)*, Gold Coast, Australia, November 2025.

4.2 Background

Multi-agents autonomous racing competitions drive research towards trajectory planning methods that aim to achieve optimal goal-reaching behaviors while safely navigating complex interactions with other agents. Notably, traditional predict-then-plan approaches often lead to overly conservative maneuvers, resulting in passive behaviors that fail to account for how other agents might react during the planning phase. To overcome these limitations, game-theoretic approaches have been proposed, aiming to devise strategies that explicitly consider and potentially leverage the influence of the planned trajectory on other agents.

In full-scale autonomous racing competitions such as the Indy Autonomous Challenge and the Abu Dhabi Autonomous Racing League, trajectory-sampling-based planning methods have been widely adopted due to their flexibility and the ease of integrating post-evaluation checks, primarily for collision avoidance, but not limited to it [32], [36]. However, this explicit assessment typically comes at the cost of coarse sampling resolution, which can result in suboptimal trajectories. Moreover, while these approaches can handle static rule sets effectively, they often struggle to incorporate interaction-based regulations—such as right-of-way rules—where the planning outcome depends on the dynamic behavior and intent of other agents on the track. In addition, regulation-compliant approaches are typically limited to regulation assessment [34], and are not exploited to enhance the effectiveness of overtaking strategies . In reinforcement learning (RL)-based approaches [62], such strategic exploitation of regulations may be implicitly embedded within the learned policy. However, these methods generally lack formal guarantees of compliance, limiting their applicability in safety-critical racing scenarios.

By modeling vehicle interactions in head-to-head racing as a non-cooperative game, different competitive behaviors can be captured by analyzing either Nash or Stackelberg equilibria [63]. When agents are subject to coupling constraints such as collision avoidance, the solution concept extends to Generalized Nash Equilibria (GNE) [42]. In such settings, Iterative Best Response (IBR) algorithms has been adopted to approximate GNE solutions [51]. IBR methods guarantee that their fixed points correspond to GNE under certain assumptions [78]. However, in dynamic games, multiple Nash equilibria may exist, and the specific equilibrium reached by IBR often depends on the initial conditions used in the iteration. While each agent optimizes its individual objective given the strategies of the others, there is generally no notion of global optimality in the multi-agent sense. Instead, optimality is assessed from the perspective of each individual agent. To address this ambiguity, the works in [52], [64] introduces a sensitivity term that influences how the joint strategy converges, effectively allowing a designer to steer the final equilibrium toward more favorable outcomes for a given agent. This capability is especially relevant when the agents have symmetric objectives or constraints, as such symmetry can otherwise lead to equilibria in which no agent's preference dominates, and small changes in initial conditions can yield drastically different equilibrium outcomes. When all the coupling constraints are shared among agents, the problem reduces to finding a non-normalized solution of a GNE [79]. However, in real-world scenarios, it is precisely *asymmetry* that grants one agent an advantage over others. Such asymmetry can arise from differences in vehicle performance or varying risk tolerances among competitors. Crucially, in organized racing, regulations often amplify these imbalances intentionally, not only to prohibit unsafe maneuvers, but also to increase entertainment value by creating strategic inequities among participants. This highlights the importance of considering regulatory compliance when designing planning algorithms, ensuring natural behaviors and competitive realism.

4.3 Problem Setting

We consider a bounded, planar track and a reference path—the racing line—defined as an arc-length parameterized curve $\boldsymbol{\xi} : [0, S_{\max}] \to \mathbb{R}^2$, assumed to be at least twice continuously differentiable (i.e., $\boldsymbol{\xi}(s) \in C^2$). Let $s \in [0, S_{\max}]$ denote the arc length parameter along this curve. The corresponding curvature profile is given by a scalar function $\kappa : [0, S_{\max}] \to \mathbb{R}$, which is assumed to be continuous. In the Frenet frame defined w.r.t. $\boldsymbol{\xi}(s)$, the left and right track boundaries are described by two continuous functions $\beta_\ell(s),\ \beta_r(s) : [0, S_{\max}] \to \mathbb{R}$, respectively, defined as the signed lateral distances from $\boldsymbol{\xi}(s)$. These offsets are not constant and may vary along the track, as the racing line does not, in general, coincide with the geometric centerline. Let $\mathbf{q} \in \mathcal{Q} \subseteq \mathbb{R}^2$ denote the position of the vehicle's center of gravity (CoG) in the

Frenet frame, with $\mathbf{q} \triangleq \begin{bmatrix} s & n \end{bmatrix}^\top$, where $s \in [0, S_{\max}]$ is the arc length along the racing line, and $n \in \mathbb{R}$ is the signed lateral displacement from $\boldsymbol{\xi}(s)$. The set

$$\mathcal{Q} \triangleq \left\{ \mathbf{q} \in \mathbb{R}^2 \mid n_r(s) \leq n \leq n_\ell(s) \right\} \tag{4.1}$$

defines the admissible positions such that the vehicle is within the track boundaries. Here, the functions $n_\ell(s),\ n_r(s) : [0, S_{\max}] \to \mathbb{R}$ are derived by tightening the geometric track boundaries $\beta_\ell(s)$ and $\beta_r(s)$ by a fixed margin that accounts for the vehicle's physical dimensions, the curvature of the racing line, and the expected vehicle's maximum heading error.

4.4 Racing Regulation Modeling

In this work, we consider a head-to-head autonomous racing scenario in which we refer to the vehicle attempting to overtake as the *attacker* and the vehicle being overtaken as the *defender*. Inspired by real-world autonomous racing competitions, we focus on the regulation that constrains the vehicles' motion during an overtaking maneuver.

Definition 2 (Overtaking Regulation)**.** The *Overtaking Regulation* defines the rules for the attacker (A) and the defender (D) during an overtaking maneuver (see Fig. 4.1):

(R1) **Right-of-Way Acquisition:** The attacker gains the right-of-way once its relative longitudinal distance to the defender is smaller than a prescribed distance Δ_s^{row} and their lateral separation is above a prescribed margin Δ_n^{row}, which also determines the overtaking side.

(R2) **Yielding Obligation:** Once the attacker gains the right-of-way, the defender must provide sufficient lateral space Δ_g^{row} on the overtaking side, provided that such space was available at the moment of the right-of-way acquisition.

(R3) **Collision Avoidance Responsibility:** Throughout the overtaking maneuver, the attacker is responsible for avoiding collisions.

Remark 4. The proposed set of rules offers a simplified yet practical abstraction of overtaking regulation. Different parameter choices can reflect more restrictive or permissive interpretations of right-of-way. A detailed modeling of competition-specific regulations is beyond the scope of this work.

Having formally defined the race regulations, we now translate them into a mathematical formulation in the form of conditional constraints. Define the joint vehicle position in the Frenet frame as $\boldsymbol{\eta} \triangleq \begin{bmatrix} \mathbf{q}^{\mathrm{D}} & \mathbf{q}^{\mathrm{A}} \end{bmatrix}^\top \in \mathcal{Q}^2$, where $\mathbf{q}^{\mathrm{D}}$ and $\mathbf{q}^{\mathrm{A}}$ are the defender's and attacker's positions, respectively. To characterize the relative longitudinal and lateral positioning between the two vehicles, we introduce the following four scalar functions $h_* : \mathcal{Q}^2 \to \mathbb{R}$:

$$h_{s^\pm}(\boldsymbol{\eta}) \triangleq \pm s_{\mathrm{D}} \mp s_{\mathrm{A}} - \Delta_s^{\text{row}}, \tag{4.2a}$$

$$h_{n^\pm}(\boldsymbol{\eta}) \triangleq \pm n_{\mathrm{D}} \mp n_{\mathrm{A}} + \Delta_n^{\text{row}}, \tag{4.2b}$$

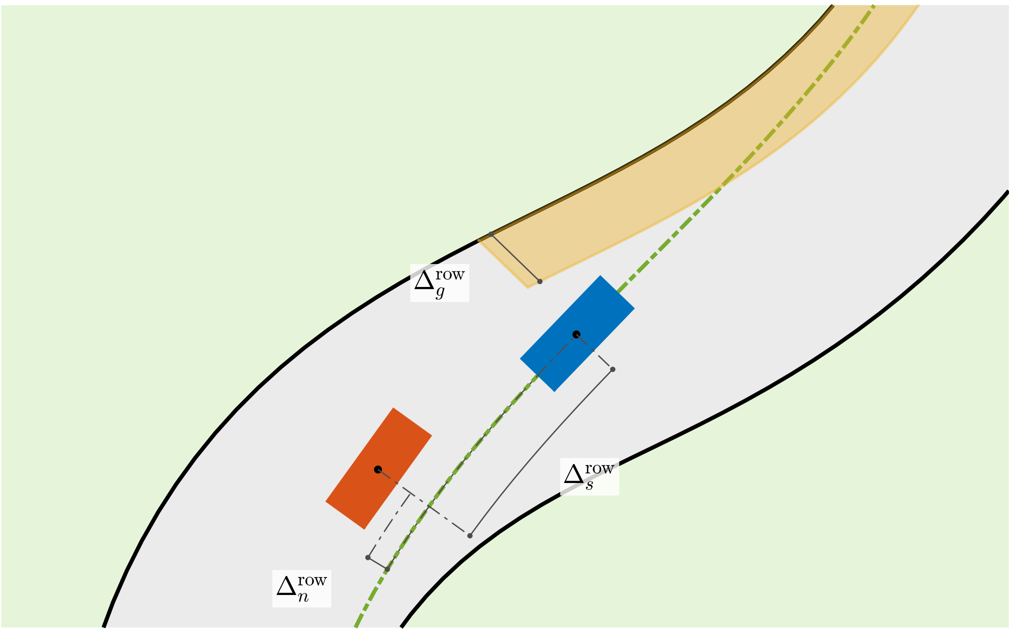


Figure 4.1. Illustration of the attacker (orange) acquiring the right of way per rules 2. The defender (blue) must yield the yellow shaded region.

along with the corresponding sets $\mathcal{S}_* \subseteq \mathcal{Q}^2$:

$$\mathcal{S}_* \triangleq \left\{\boldsymbol{\eta} \in \mathcal{Q}^2 \mid h_*(\boldsymbol{\eta}) \leq 0\right\} \quad \forall * \in \mathcal{I}_{sn}, \tag{4.3}$$

where $\mathcal{I}_{sn} \triangleq \{s^+, s^-, n^+, n^-\}$. Additionally, we define

$$\mathcal{S}_s \triangleq \mathcal{S}_{s^+} \cap \mathcal{S}_{s^-}, \tag{4.4}$$

which denotes the set of joint positions $\boldsymbol{\eta}$ where the relative longitudinal distance between the attacker and the defender is less than Δ_s^{row}. Let $\bar{\boldsymbol{\eta}} \triangleq \boldsymbol{\eta}(\bar{t})$ be the joint *crossing* position, with

$$\bar{t} \triangleq \sup\left\{t \in \mathbb{R}^+ \mid h_{s^+}(\boldsymbol{\eta}(t)) = 0\right\}, \tag{4.5}$$

representing the latest time instant the attacker closed the longitudinal gap to Δ_s^{row}. Hence, $\bar{\boldsymbol{\eta}}$ corresponds to the joint configuration of the two vehicles at the time of the most recent crossing of the longitudinal distance threshold Δ_s^{row}.

Let $\Delta_\ell,\ \Delta_r : \mathcal{Q} \to \mathbb{R}$ be the signed lateral distance to the left and right tightened track margins, respectively:

$$\begin{aligned} \Delta_\ell(\mathbf{q}) &= n_\ell(s) - n, \\ \Delta_r(\mathbf{q}) &= n - n_r(s). \end{aligned} \tag{4.6}$$

Additionally, let $\Delta_{g\ell},\ \Delta_{gr} : \mathcal{Q} \to \mathbb{R}$ be the *effective* granted lateral space that the defender must leave on the left and right side, respectively, defined as:

$$\begin{aligned} \Delta_{g\ell}(\mathbf{q}) &\triangleq \min\Big(\Delta_g^{\text{row}}, \Delta_\ell(\mathbf{q})\Big), \\ \Delta_{gr}(\mathbf{q}) &\triangleq \min\Big(\Delta_g^{\text{row}}, \Delta_r(\mathbf{q})\Big). \end{aligned} \tag{4.7}$$

Finally, we define the sets $\mathcal{Q}_\ell^{\text{row}}$, $\mathcal{Q}_r^{\text{row}} \subseteq \mathcal{Q}$ as the admissible defender positions corresponding to a given crossing position $\bar{\mathbf{q}}_D$, for a left and right overtake, respectively:

$$\begin{aligned} \mathcal{Q}_\ell^{\text{row}}(\bar{\mathbf{q}}^{\text{D}}) &\triangleq \{\mathbf{q}^{\text{D}} \in \mathcal{Q} \mid h_\ell^{\text{row}}(\mathbf{q}^{\text{D}}, \bar{\mathbf{q}}^{\text{D}}) \leq 0\}, \\ \mathcal{Q}_r^{\text{row}}(\bar{\mathbf{q}}^{\text{D}}) &\triangleq \{\mathbf{q}^{\text{D}} \in \mathcal{Q} \mid h_r^{\text{row}}(\mathbf{q}^{\text{D}}, \bar{\mathbf{q}}^{\text{D}}) \leq 0\}, \end{aligned} \tag{4.8}$$

where

$$\begin{aligned} h_\ell^{\text{row}}(\mathbf{q}^{\text{D}}, \bar{\mathbf{q}}^{\text{D}}) &\triangleq -\Delta_\ell(\mathbf{q}^{\text{D}}) + \Delta_{g\ell}(\bar{\mathbf{q}}^{\text{D}}), \\ h_r^{\text{row}}(\mathbf{q}^{\text{D}}, \bar{\mathbf{q}}^{\text{D}}) &\triangleq -\Delta_r(\mathbf{q}^{\text{D}}) + \Delta_{gr}(\bar{\mathbf{q}}^{\text{D}}). \end{aligned} \tag{4.9}$$

In words, the sets in (4.8) define the admissible defender positions obtained by reducing the nominal lateral margins n_ℓ and n_r by the effective granted space $\Delta_{g\ell}$ and Δ_{gr}, each calculated as the smallest between the prescribed offset Δ_g^{row} and the available lateral clearance $\Delta_\ell(\bar{\mathbf{q}}^{\text{D}})$ and $\Delta_r(\bar{\mathbf{q}}^{\text{D}})$ at time $\bar{t}$, respectively.

Definition 3 (Right-of-Way Compliance)**.** The defender's position $\mathbf{q}^{\text{D}} \in \mathcal{Q}$ is said to be *right-of-way compliant* with respect to the joint current position $\boldsymbol{\eta}$ and the joint crossing position $\bar{\boldsymbol{\eta}}$ if the following conditions hold:

$$\begin{cases} \text{if } \boldsymbol{\eta} \in \mathcal{S}_s \text{ and } \bar{\boldsymbol{\eta}} \in \mathcal{S}_{n^+}, & \text{then } \mathbf{q}^{\text{D}} \in \mathcal{Q}_\ell^{\text{row}}(\bar{\mathbf{q}}^{\text{D}}), \\ \text{if } \boldsymbol{\eta} \in \mathcal{S}_s \text{ and } \bar{\boldsymbol{\eta}} \in \mathcal{S}_{n^-}, & \text{then } \mathbf{q}^{\text{D}} \in \mathcal{Q}_r^{\text{row}}(\bar{\mathbf{q}}^{\text{D}}). \end{cases} \tag{4.10}$$

The conditions (4.10) formalize the notion of right-of-way compliance. Specifically, when the longitudinal gap between the two vehicles is smaller than Δ_s^{row} ($\boldsymbol{\eta} \in \mathcal{S}_s$), the defender is required to yield lateral space to the attacker according to the side from which the overtake was initiated. If the attacker was on the right (left) of the defender by at least Δ_n^{row} at the time the longitudinal gap was closed ($\bar{\boldsymbol{\eta}} \in \mathcal{S}_{n^+}$ or $\bar{\boldsymbol{\eta}} \in \mathcal{S}_{n^-}$), the defender must maintain its position within the admissible region $\mathcal{Q}_\ell^{\text{row}}(\bar{\mathbf{q}}^{\text{D}})$ ($\mathcal{Q}_r^{\text{row}}(\bar{\mathbf{q}}^{\text{D}})$, respectively). This region is obtained by reducing the nominal track boundaries by an amount equal to the minimum between the defender's lateral distance to the corresponding boundary at the time of gap closure $\bar{t}$ and the prescribed limit Δ_g^{row}. This ensures that the defender yields exactly the effective lateral space defined by the right-of-way rule, only on the overtaking side.

Formulation (4.10) highlights both the conditional nature of the right-of-way constraints imposed on the defender and their dependence on the joint crossing position $\bar{\boldsymbol{\eta}}$. The remainder of this section discusses how to encode these constraints as inequalities, so they can be embedded in an optimization problem.

Let us introduce the indicator function associated with the sets (4.3) as:

$$\mathbf{1}_{\mathcal{S}_*}(\boldsymbol{\eta}) \triangleq \begin{cases} 1, & h_*(\boldsymbol{\eta}) \leq 0 \\ 0, & \text{otherwise} \end{cases} \tag{4.11}$$

Hence, logical constraints (4.10) can be reformulated as inequality constraints introducing *big-M* constants [47]:

$$\begin{cases} h_\ell^{\text{row}}(\mathbf{q}^{\text{D}}, \bar{\mathbf{q}}^{\text{D}}) \leq M_\ell \left(1 - \mathbf{1}_{\mathcal{S}_s}(\boldsymbol{\eta}) \mathbf{1}_{\mathcal{S}_{n^+}}(\bar{\boldsymbol{\eta}})\right), \\ h_r^{\text{row}}(\mathbf{q}^{\text{D}}, \bar{\mathbf{q}}^{\text{D}}) \leq M_r \left(1 - \mathbf{1}_{\mathcal{S}_s}(\boldsymbol{\eta}) \mathbf{1}_{\mathcal{S}_{n^-}}(\bar{\boldsymbol{\eta}})\right), \end{cases} \tag{4.12}$$

where $M_* \in \mathbb{R} : h_*^{\text{row}}(\mathbf{q}^{\mathrm{D}}, \bar{\mathbf{q}}^{\mathrm{D}}) \leq M_* \; \forall \, \mathbf{q}, \bar{\mathbf{q}} \in \mathcal{Q}, \; * \in \{\ell, r\}$.

We finally introduce the dynamics of $\bar{\boldsymbol{\eta}}$, which, together with the constraint functions in (4.12), enable the formulation of a predictive optimal motion planning problem. The evolution of $\bar{\boldsymbol{\eta}}$ can be modeled as a *Sample-and-Hold* system:

$$\bar{\boldsymbol{\eta}}_{k+1} = \begin{cases} \bar{\boldsymbol{\eta}}_k & \text{if } \boldsymbol{\eta} \in \mathcal{S}_{s^+}, \\ \boldsymbol{\eta}_k & \text{if } \boldsymbol{\eta} \notin \mathcal{S}_{s^+}. \end{cases} \tag{4.13}$$

Using the indicator function:

$$\bar{\boldsymbol{\eta}}_{k+1} = \bar{\boldsymbol{\eta}}_k \mathbf{1}_{S_{s^+}}(\boldsymbol{\eta}_k) + \boldsymbol{\eta}_k \left(1 - \mathbf{1}_{S_{s^+}}(\boldsymbol{\eta}_k)\right). \tag{4.14}$$

It is worth noting that the right-of-way constraints (4.12) and the sample-and-hold dynamics (4.14) are inherently discontinuous due to the presence of indicator functions, and therefore cannot be directly incorporated into optimization problems solved by traditional gradient-based methods. While smooth approximations of the indicator functions may be employed to recover differentiability, such approaches generally introduce approximation errors that may compromise the effectiveness of the proposed formulation. To address this issue, the next section presents an exact encoding of such constraints by leveraging the Mixed Logical Dynamical modeling framework.

4.5 Regulation-Compliant Motion Planning

Starting from the indicator function formulation, we leverage the MLD modeling framework to derive a formulation that integrates both continuous and integer variables—specifically binary variables—into constraints where such integer variables enter *linearly*.

4.5.1 MLD Right-of-Way Constraints

For each function defined in (4.2) we introduce a binary variable $\delta_* \in \{0, 1\}$ along with the corresponding logical implication:

$$[\delta_* = 1] \iff [h_*(\boldsymbol{\eta}) \leq 0] \quad \forall * \in \mathcal{I}_{sn}. \tag{4.15}$$

This relation can be encoded via the following set of mixed-integer inequalities:

$$\begin{cases} h_*(\boldsymbol{\eta}) \leq M_*(1 - \delta_*), \\ h_*(\boldsymbol{\eta}) \geq \epsilon + (m_* - \epsilon)\delta_*, \end{cases} \qquad \forall * \in \mathcal{I}_{sn} \tag{4.16}$$

where $M_*, m_* \in \mathbb{R}$ are chosen such that $m_* \leq h_*(\boldsymbol{\eta}) \leq M_* \; \forall \, \boldsymbol{\eta} \in \mathcal{Q}^2$, and ϵ is a small tolerance introduced for numerical robustness [47]. Note that, with a slight abuse of notation, the functions $h_*(\cdot)$ in (4.15) and (4.16) are evaluated at $\bar{\boldsymbol{\eta}}$ for $* \in \{n^+, n^-\}$, in accordance with Definition 3.

Notably, the product of two indicator functions translates into a logical AND condition. We then introduce the binary variables $\delta_s \triangleq \delta_{s^+ \wedge s^-}$, $\delta_\ell \triangleq \delta_{s \wedge n^+}$, and

$\delta_r \triangleq \delta_{s\wedge n^-}$, where we adopt the notation $\delta_{a\wedge b} \triangleq \delta_a \wedge \delta_b$. The logical AND condition is enforced through the following inequalities:

$$\begin{cases} \delta_{a\wedge b} \leq \delta_a, \\ \delta_{a\wedge b} \leq \delta_b, \\ \delta_a + \delta_b \leq \delta_{a\wedge b} + 1, \end{cases} \qquad \forall\,(a,b) \in \mathcal{I}_\wedge \tag{4.17}$$

with $\mathcal{I}_\wedge \triangleq \{(s^+, s^-),\ (s, n^+),\ (s, n^-)\}$. The constraints in (4.12) can now be rewritten using the binary variables δ_ℓ and δ_r as follows:

$$\begin{cases} h_\ell^{\text{row}}(\mathbf{q}^{\text{D}}, \bar{\mathbf{q}}^{\text{D}}) \leq M_\ell \left(1 - \delta_\ell\right), \\ h_r^{\text{row}}(\mathbf{q}^{\text{D}}, \bar{\mathbf{q}}^{\text{D}}) \leq M_r \left(1 - \delta_r\right). \end{cases} \tag{4.18}$$

Note that the $\min(\cdot)$ operation in (4.7) can be explicitly modeled within the MLD framework. To this end, we introduce binary variables $\delta_{g\ell}$ and δ_{gr}, as well as auxiliary continuous variables $z_{g\ell}$ and z_{gr}. The following mixed-integer constraints enforce $z_{g\ell} = \Delta_{g\ell}(\bar{\mathbf{q}}^{\text{D}})$ and $z_{gr} = \Delta_{gr}(\bar{\mathbf{q}}^{\text{D}})$:

$$\begin{cases} \Delta_g^{\text{row}} - M_{g*}(1 - \delta_{g*}) \leq z_{g*} \leq \Delta_g^{\text{row}}, \\ \Delta_*(\bar{\mathbf{q}}^{\text{D}}) - M_{g*}\delta_{g*} \leq z_{g*} \leq \Delta_*(\bar{\mathbf{q}}^{\text{D}}), \end{cases} \qquad \forall * \in \{\ell, r\} \tag{4.19}$$

where $M_{g*} \in \mathbb{R} : |\Delta_g^{\text{row}} - \Delta_*(\mathbf{q})| \leq M_{g*}\ \forall\, \mathbf{q} \in \mathcal{Q}$. The MLD form of constraints (4.12) becomes:

$$\begin{cases} -\Delta_\ell(\mathbf{q}^{\text{D}}) + z_{g\ell} \leq M_\ell \left(1 - \delta_\ell\right), \\ -\Delta_r(\mathbf{q}^{\text{D}}) + z_{gr} \leq M_r \left(1 - \delta_r\right). \end{cases} \tag{4.20}$$

Finally, we collect all constraints introduced in (4.16), (4.17), (4.19), and (4.20), and express them in compact form by moving all terms to the left-hand side:

$$\boldsymbol{\gamma}^{\text{row}}\left(\boldsymbol{\eta}, \bar{\boldsymbol{\eta}}, \boldsymbol{\delta}^{\text{row}}, \mathbf{z}^{\text{row}}\right) \leq \mathbf{0}, \tag{4.21}$$

where the stacked binary and continuous auxiliary variables are defined as:

$$\boldsymbol{\delta}^{\text{row}} \triangleq \begin{bmatrix} \{\delta_*\}_{*\in\mathcal{I}_{sn}} & \delta_s & \delta_{g\ell} & \delta_{gr} & \delta_\ell & \delta_r \end{bmatrix}^\top, \tag{4.22}$$

$$\mathbf{z}^{\text{row}} \triangleq \begin{bmatrix} z_{g\ell} & z_{gr} \end{bmatrix}^\top. \tag{4.23}$$

4.5.2 MLD Sample-and-Hold Dynamics

The dynamics of $\bar{\boldsymbol{\eta}}$ in (4.14) can be rewritten using the previously introduced binary variable δ_{s^+}:

$$\begin{aligned} \bar{\boldsymbol{\eta}}_{k+1} &= \bar{\boldsymbol{\eta}}_k \delta_{s^+,k} + \boldsymbol{\eta}_k (1 - \delta_{s^+,k}) \\ &= (\bar{\boldsymbol{\eta}}_k - \boldsymbol{\eta}_k)\delta_{s^+,k} + \boldsymbol{\eta}_k. \end{aligned} \tag{4.24}$$

Here, we replace the bilinear term involving the binary variable with a vector of auxiliary continuous variables $\mathbf{z}_{\text{sh}} \in \mathbb{R}^4$. The following mixed-integer constraints enforce $\mathbf{z}_{\text{sh}} = \left(\bar{\boldsymbol{\eta}} - \boldsymbol{\eta}\right)\delta_{s^+}$:

$$\begin{cases} \mathbf{m}_{\text{sh}}\delta_{s^+} \leq \mathbf{z}_{\text{sh}} \leq \mathbf{M}_{\text{sh}}\delta_{s^+}, \\ (\bar{\boldsymbol{\eta}} - \boldsymbol{\eta}) - \mathbf{M}_{\text{sh}}(1 - \delta_{s^+}) \leq \mathbf{z}_{\text{sh}}, \\ (\bar{\boldsymbol{\eta}} - \boldsymbol{\eta}) - \mathbf{m}_{\text{sh}}(1 - \delta_{s^+}) \geq \mathbf{z}_{\text{sh}}, \end{cases} \tag{4.25}$$

where $\mathbf{m}_{\text{sh}}$, $\mathbf{M}_{\text{sh}} \in \mathbb{R}^4$ are constant vectors satisfying $\mathbf{m}_{\text{sh}} \leq (\bar{\boldsymbol{\eta}} - \boldsymbol{\eta}) \leq \mathbf{M}_{\text{sh}} \; \forall \boldsymbol{\eta}, \bar{\boldsymbol{\eta}} \in \mathcal{Q}^2$. Defining the function $\mathbf{f}_{\text{sh}}(\boldsymbol{\eta}_k, \mathbf{z}_{\text{sh},k}) \triangleq \boldsymbol{\eta}_k + \mathbf{z}_{\text{sh},k}$, the MLD sample-and-hold system is described by:

$$\begin{aligned} &\bar{\boldsymbol{\eta}}_{k+1} = \mathbf{f}_{\text{sh}}(\boldsymbol{\eta}_k, \mathbf{z}_{\text{sh},k}), \\ &\boldsymbol{\gamma}^{\text{sh}}(\boldsymbol{\eta}_k, \bar{\boldsymbol{\eta}}_k, \delta_{s^+,k}, \mathbf{z}_{\text{sh},k}) \leq \mathbf{0}, \end{aligned} \tag{4.26}$$

where $\boldsymbol{\gamma}^{\text{sh}}(\cdot)$ collects constraints (4.25) along with constraints (4.16) for $* = s^+$.

4.5.3 MLD Collision Avoidance Constraints

Let Δ_s^{CA} and Δ_n^{CA} be longitudinal and lateral safety margins, accounting for vehicles dimensions, racing line curvature, and maximum orientation misalignment. Introduce the following scalar functions:

$$h_{s^\pm}^{\text{CA}}(\boldsymbol{\eta}) \triangleq \pm s_{\text{D}} \mp s_{\text{A}} + \Delta_s^{\text{CA}}, \tag{4.27a}$$

$$h_{n^\pm}^{\text{CA}}(\boldsymbol{\eta}) \triangleq \pm n_{\text{D}} \mp n_{\text{A}} + \Delta_n^{\text{CA}}. \tag{4.27b}$$

Then collision-avoidance is ensured if:

$$h_*^{\text{CA}}(\boldsymbol{\eta}) \leq 0 \quad \text{for at least one } * \in \mathcal{I}_{sn}. \tag{4.28}$$

Introducing the vector of binary variables $\boldsymbol{\delta}^{\text{CA}} \triangleq \{\delta_*^{\text{CA}}\}_{*\in\mathcal{I}_{\text{CA}}}$:

$$\begin{cases} h_*^{\text{CA}}(\boldsymbol{\eta}) \leq M_*^{\text{CA}}(1 - \delta_*^{\text{CA}}), \\ \sum_{*\in\mathcal{I}_{sn}} \delta_*^{\text{CA}} \geq 1 \end{cases} \quad \forall * \in \mathcal{I}_{sn}, \tag{4.29}$$

where $M_*^{\text{CA}} \in \mathbb{R} : h_*^{\text{CA}}(\boldsymbol{\eta}) \leq M_*^{\text{CA}} \; \forall \boldsymbol{\eta} \in \mathcal{Q}^2 \quad \forall * \in \mathcal{I}_{sn}$. Finally, we collect the collision avoidance constraints (4.29) in:

$$\gamma^{\text{CA}}(\boldsymbol{\eta}, \boldsymbol{\delta}^{\text{CA}}) \leq 0. \tag{4.30}$$

4.5.4 Regulation-Compliant Model Predictive Control

For notational convenience, we define $(i, -i) \in \{\text{A}, \text{D}\}$ to denote a generic player–opponent pair, where i refers to the player under consideration (either the attacker or the defender), and $-i$ denotes the corresponding opponent. Also for clarity, we replace the joint notation $\boldsymbol{\eta}$ with explicit arguments $(\mathbf{x}^i, \mathbf{x}^{-i})$, representing the individual player states, given that the mapping from the state $\mathbf{x}^i$ to the position $\mathbf{q}^i$ is straightforward.

For each player i, let $\mathcal{X}^i \subseteq \mathbb{R}^{n^i}$ and $\mathcal{U}^i \subseteq \mathbb{R}^{m^i}$ denote the admissible state and control input sets, respectively. We define $\mathbf{r}_x^i : \mathcal{X}^i \to \mathcal{X}^i$ and $\mathbf{r}_u^i : \mathcal{X}^i \to \mathcal{U}^i$ as the state dependent reference state and control input, respectively, and assume that $\mathbf{r}_x^i$, $\mathbf{r}_u^i \in C^2$, i.e., both functions are twice continuously differentiable. We consider the stage and terminal cost functions:

$$\ell^i(\mathbf{x}_k^i, \mathbf{u}_k^i) \triangleq \left\| \mathbf{x}_k^i - \mathbf{r}_x^i(\mathbf{x}_k^i) \right\|_{Q^i}^2 + \left\| \mathbf{u}_k^i - \mathbf{r}_u^i(\mathbf{x}_k^i) \right\|_{R^i}^2, \tag{4.31}$$

$$\ell_f^i(\mathbf{x}_N^i) \triangleq \left\| \mathbf{x}_N^i - \mathbf{r}_x^i(\mathbf{x}_N^i) \right\|_{P^i}^2, \tag{4.32}$$

where $Q^i \succeq 0$, $R^i \succ 0$, and $P^i \succeq 0$ are the weighting matrices for the state, input, and terminal cost, respectively. The total cost over the prediction horizon N is then:

$$J^i(X^i, U^i) \triangleq \sum_{k=0}^{N-1} \ell^i(\mathbf{x}_k^i, \mathbf{u}_k^i) + \ell_f^i(\mathbf{x}_N^i), \tag{4.33}$$

where $X^i \triangleq \{\mathbf{x}_k^i\}_{k=0}^N$ and $U^i \triangleq \{\mathbf{u}_k^i\}_{k=0}^{N-1}$. Let $\boldsymbol{\chi}$ denote the state of a vehicle motion model in the Frenet frame, with dynamics $\boldsymbol{\chi}_{k+1} = \mathbf{f}_\chi(\boldsymbol{\chi}_k, \mathbf{u}_k)$. We define the state, binary, and auxiliary variables for each agent as:

$$\begin{aligned} &\mathbf{x}^{\mathrm{A}} \triangleq \boldsymbol{\chi}^{\mathrm{A}}, && \mathbf{x}^{\mathrm{D}} \triangleq \begin{bmatrix} \boldsymbol{\chi}^{\mathrm{D}} & \bar{\boldsymbol{\eta}} \end{bmatrix}^\top, \\ &\boldsymbol{\delta}^{\mathrm{A}} \triangleq \boldsymbol{\delta}^{\mathrm{CA}}, && \boldsymbol{\delta}^{\mathrm{D}} \triangleq \boldsymbol{\delta}^{\mathrm{row}}, \\ &\mathbf{z}^{\mathrm{A}} \triangleq \varnothing, && \mathbf{z}^{\mathrm{D}} \triangleq \begin{bmatrix} \mathbf{z}^{\mathrm{row}} & \mathbf{z}_{\mathrm{sh}} \end{bmatrix}^\top. \end{aligned} \tag{4.34}$$

The attacker and defender states evolve according to:

$$\begin{aligned} \mathbf{x}_{k+1}^{\mathrm{A}} &= \mathbf{f}^{\mathrm{A}}(\mathbf{x}_k^{\mathrm{A}}, \mathbf{u}_k^{\mathrm{A}}) \triangleq \mathbf{f}_\chi(\mathbf{x}_k^{\mathrm{A}}, \mathbf{u}_k^{\mathrm{A}}), \\ \mathbf{x}_{k+1}^{\mathrm{D}} &= \mathbf{f}^{\mathrm{D}}(\mathbf{x}_k^{\mathrm{D}}, \mathbf{u}_k^{\mathrm{D}}, \mathbf{z}_k^{\mathrm{D}}, \mathbf{x}_k^{\mathrm{A}}) \triangleq \begin{bmatrix} \mathbf{f}_\chi(\mathbf{x}_k^{\mathrm{D}}, \mathbf{u}_k^{\mathrm{D}}) \\ \mathbf{f}_{\mathrm{sh}}(\mathbf{x}_k^{\mathrm{D}}, \mathbf{z}_k^{\mathrm{D}}, \mathbf{x}_k^{\mathrm{A}}) \end{bmatrix}. \end{aligned} \tag{4.35}$$

Each player is subject to stage-wise state-input constraints and a terminal state constraint:

$$\begin{aligned} \mathbf{g}_{x,u}^i(\mathbf{x}_k^i, \mathbf{u}_k^i) &\leq \mathbf{0}, \\ \mathbf{g}_N^i(\mathbf{x}_N^i) &\leq \mathbf{0}. \end{aligned} \tag{4.36}$$

Finally, the overtaking regulations are encoded via *coupling* constraints:

$$\begin{aligned} \boldsymbol{\gamma}^{\mathrm{A}}(\mathbf{x}^{\mathrm{A}}, \boldsymbol{\delta}^{\mathrm{A}}, \mathbf{x}^{\mathrm{D}}) &\triangleq \boldsymbol{\gamma}^{\mathrm{CA}}(\mathbf{x}^{\mathrm{A}}, \boldsymbol{\delta}^{\mathrm{A}}, \mathbf{x}^{\mathrm{D}}), \\ \boldsymbol{\gamma}^{\mathrm{D}}(\mathbf{x}^{\mathrm{D}}, \boldsymbol{\delta}^{\mathrm{D}}, \mathbf{z}^{\mathrm{D}}, \mathbf{x}^{\mathrm{A}}) &\triangleq \begin{bmatrix} \boldsymbol{\gamma}^{\mathrm{sh}}(\mathbf{x}^{\mathrm{D}}, \boldsymbol{\delta}^{\mathrm{D}}, \mathbf{z}^{\mathrm{D}}, \mathbf{x}^{\mathrm{A}}) \\ \boldsymbol{\gamma}^{\mathrm{row}}(\mathbf{x}^{\mathrm{D}}, \boldsymbol{\delta}^{\mathrm{D}}, \mathbf{z}^{\mathrm{D}}) \end{bmatrix}. \end{aligned} \tag{4.37}$$

We now formally define the player-specific MPC problem that incorporates regulation constraints.

Definition 4 (Regulation-Compliant MPC (RC-MPC))**.** The *Regulation-Compliant MPC (RC-MPC)* formulation that enforces overtaking regulation compliance for the player i is defined as[1]:

$$\min_{X^i, U^i, \Delta^i, Z^i} \; J^i(X^i, U^i) \tag{4.38a}$$

$$\text{s.t.} \;\; \mathbf{x}_{k+1}^i = \mathbf{f}^i(\mathbf{x}_k^i, \mathbf{u}_k^i, \mathbf{z}_k^i, \mathbf{x}_k^{-i}), \qquad k \in \mathbb{I}_0^{N-1} \tag{4.38b}$$

$$\mathbf{x}_0^i = \mathbf{x}^i(t), \tag{4.38c}$$

$$\mathbf{g}_N^i(\mathbf{x}_N^i) \leq \mathbf{0}, \tag{4.38d}$$

$$\mathbf{g}_{x,u}^i(\mathbf{x}_k^i, \mathbf{u}_k^i) \leq \mathbf{0}, \qquad k \in \mathbb{I}_0^{N-1} \tag{4.38e}$$

$$\boldsymbol{\gamma}^i(\mathbf{x}_k^i, \boldsymbol{\delta}_k^i, \mathbf{z}_k^i, \mathbf{x}_k^{-i}) \leq \mathbf{0}, \qquad k \in \mathbb{I}_0^N \tag{4.38f}$$

[1]The time instant t is omitted from prediction variables for simplicity, assuming all quantities are defined relative to the current time.

where the optimization variables are $X^i \triangleq \{\mathbf{x}_k^i\}_{k=0}^N$, $U^i \triangleq \{\mathbf{u}_k^i\}_{k=0}^{N-1}$, $\Delta^i \triangleq \{\boldsymbol{\delta}_k^i\}_{k=0}^N$, $Z^i \triangleq \{\mathbf{z}_k^i\}_{k=0}^N$.

Remark 5. Problem (4.38) is a Mixed-Integer Nonlinear Program (MINLP), where (i) the binary variables appear only in the inequality constraints and enter linearly, and (ii) the cost function is assumed to be twice continuously differentiable. In general, solving MINLPs is computationally challenging. To address this, we adopt a Mixed-Integer Sequential Quadratic Programming (MI-SQP) approach [48]. Analogous to classical SQP, the MI-SQP method sequentially computes and solves Mixed-Integer Quadratic Program (MIQP) approximations of the original problem until convergence to a (local) minimizer.

4.6 Regulation-Aware Game-Theoretic Planning

In this section, we leverage the proposed RC-MPC framework to model the interactions of the two vehicles during an overtaking maneuver. Specifically, we focus on the design of a game-theoretic motion planner that brings predictive reasoning capabilities over the regulation constraining the defender to enhance the effectiveness of the attacker's overtaking strategies.

The proposed interaction is modeled as a non-cooperative, two-player, finite-horizon, differential game with player-specific coupling constraints.

Definition 5 (Regulation-Constrained Racing Game)**.** We define the *Regulation-Constrained Racing Game* at time t as:

$$\mathcal{G}(t) \triangleq \left\langle \mathcal{I},\, \mathcal{P}^i(\mathbf{x}^i(t), X^{-i}(t)) \right\rangle_{i\in\mathcal{I}}, \tag{4.39}$$

where:

- $\mathcal{I} \triangleq \{\mathrm{A}, \mathrm{D}\}$ is the set of players (attacker and defender);
- $\mathbf{x}^i(t)$ is the state of agent i at current time t;
- $X^{-i}(t)$ is the predicted state trajectory of the opponent over the horizon N;
- $\mathcal{P}^i(\mathbf{x}^i(t), X^{-i}(t))$ is the RC-MPC problem (4.38) solved by agent i, initialized at $\mathbf{x}^i(t)$, and conditioned on $X^{-i}(t)$.

To characterize the outcome of the game $\mathcal{G}$, we adopt the solution concept of a Generalized Nash Equilibrium, where each player optimizes its own trajectory over a finite horizon, subject to *player-specific* constraints that may depend on the opponent's strategy.

Definition 6 (Generalized Nash Equilibrium (GNE))**.** Let the strategy of player $i \in \mathcal{I}$ be defined as $w^i \triangleq (X^i, U^i, Z^i, \Delta^i)$, and let the feasible set be given by

$$W^i(w^{-i}) \triangleq \left\{ w^i \mid w^i \text{ satisfies (4.38b)–(4.38f)} \right\}. \tag{4.40}$$

A strategy pair $\{w^{i\star}\}_{i\in\mathcal{I}}$ is a *Generalized Nash Equilibrium* of the game $\mathcal{G}$ in (4.39) if, for each player $i \in \mathcal{I}$,

$$w^{i\star} \in \mathrm{BR}^i(w^{-i\star}), \tag{4.41}$$

where the *best-response mapping* is defined as

$$\mathrm{BR}^i(w^{-i}) \triangleq \arg\min_{w^i} J^i(w^i) \quad \text{s.t.} \quad w^i \in W^i(w^{-i}). \tag{4.42}$$

Equivalently, a GNE is a fixed point of the collective best-response mapping.

Remark 6. The regulation-constrained racing game is inherently asymmetric due to the player-specific regulations, which assign different coupling constraints to each player. Although both agents in reality are interested in avoiding collisions, this asymmetry is intended to model a reasoning framework where the regulation assigns greater responsibility for safety to the attacker. From a safety perspective, including collision avoidance constraints in the defender's problem would be redundant: any feasible strategy pair inherently prevents collisions for *both* agents, provided such a strategy pair exists. On the other hand, enforcing collision avoidance also on the defender could admit GNEs in which the attacker exploits the defender's reactive behavior, for example by forcing it off the racing line. This may promote overly aggressive or unfair behavior, which motivates our choice to assign collision avoidance constraints only to the attacker. Finally, we highlight that ensuring the existence of a GNE to this asymmetric and non-convex game is nontrivial and depends critically on the proper design of the regulatory framework.

On "asymmetric" GNEP. To formally characterize the implications of assigning player-specific coupling constraints, we analyze the relation between an "asymmetric" game, in which each coupling constraint is private to its corresponding player, and the corresponding "symmetric" game, where all coupling constraints are shared among all players.

Proposition 1 (Inclusion of GNE solutions: asymmetric ⇒ symmetric)**.** *Let $\mathcal{G}_{\mathrm{asym}}$ be a GNEP in which each player $i \in \mathcal{I}$ enforces only its own coupling constraint $\boldsymbol{\gamma}^i(w^i, w^{-i}) \leq \mathbf{0}$, with feasible set*

$$W_i^{\mathrm{asym}}(w^{-i}) \triangleq \{\, w^i \mid \boldsymbol{\gamma}^i(w^i, w^{-i}) \leq \mathbf{0} \,\}. \tag{4.43}$$

Let $\mathcal{G}_{\mathrm{sym}}$ be the corresponding symmetric GNEP, where each player enforces all coupling constraints, i.e.,

$$W_i^{\mathrm{sym}}(w^{-i}) \triangleq \bigcap_{j\in\mathcal{I}} \{\, w^i \mid \boldsymbol{\gamma}^j(w^i, w^{-i}) \leq \mathbf{0} \,\}. \tag{4.44}$$

Then any GNE $w^\star$ of $\mathcal{G}_{\mathrm{asym}}$ is also a GNE of $\mathcal{G}_{\mathrm{sym}}$. However, the converse does not hold in general: a GNE of $\mathcal{G}_{\mathrm{sym}}$ is not necessarily a GNE of $\mathcal{G}_{\mathrm{asym}}$, as it may rely on constraints not enforced by all players in the asymmetric formulation.

Proof. Let $w^\star = \{w^{i\star}\}_{i\in\mathcal{I}}$ be a GNE of the "asymmetric" game $\mathcal{G}_{\mathrm{asym}}$. By definition, for every player $i \in \mathcal{I}$,

$$w^{i\star} \in \arg\min_{w^i \in W_i^{\mathrm{asym}}(w^{-i\star})} J_i(w^i, w^{-i\star}). \tag{4.45}$$

Because $w^{i\star}$ satisfies its private constraint for every i, we have $\boldsymbol{\gamma}^i(w^{i\star}, w^{-i\star}) \leq \mathbf{0}$ for all i. Evaluating these inequalities at the joint profile $w^\star$ shows that all coupling constraints are satisfied: $\boldsymbol{\gamma}^j(w^\star) \leq \mathbf{0}$ for every $j \in \mathcal{I}$. Hence $w^{i\star}$ is also feasible for the symmetric game, that is, $w^{i\star} \in W_i^{\text{sym}}(w^{-i\star})$ for all i.

Now, for any fixed w^{-i}, the feasible sets satisfy $W_i^{\text{sym}}(w^{-i}) \subseteq W_i^{\text{asym}}(w^{-i})$, since the symmetric game adds additional coupling constraints for $j \neq i$. Because $w^{i\star}$ minimizes $J_i(\cdot, w^{-i\star})$ over the larger set $W_i^{\text{asym}}(w^{-i\star})$, it remains optimal over the subset $W_i^{\text{sym}}(w^{-i\star})$ that still contains $w^{i\star}$. Each player's strategy $w^{i\star}$ is therefore a best response within the feasible set of the symmetric game. Consequently, the joint profile $w^\star$ is a GNE of $\mathcal{G}_{\text{sym}}$.

The reverse implication does not hold. If $\tilde{w}$ is a GNE of the symmetric game, relaxing a player's feasible set from $W_i^{\text{sym}}(w^{-i})$ to $W_i^{\text{asym}}(w^{-i})$ removes some coupling constraints that were previously enforced. As a result, player i may find a new strategy $\hat{w}^i \in W_i^{\text{asym}}(\tilde{w}^{-i})$ such that $J_i(\hat{w}^i, \tilde{w}^{-i}) < J_i(\tilde{w}^i, \tilde{w}^{-i})$, violating equilibrium optimality. Hence, a symmetric GNE is not necessarily a GNE of the asymmetric game. □

Proposition 1 confirms that the asymmetric formulation preserves all mutually feasible and regulation-compliant equilibria, while avoiding redundant shared responsibilities. It therefore provides the theoretical justification for assigning collision-avoidance obligations only to the attacker, as implemented in the planner design described next.

To compute an approximate solution to the GNE, we adopt an Iterative Best Response scheme [42], wherein each agent $i \in \mathcal{I}$ sequentially updates its strategy as $w^{i,(\ell+1)} \in \mathrm{BR}^i(w^{-i,(\ell)})$, assuming the opponent's strategy $w^{-i,(\ell)}$ is fixed. The process is repeated until convergence to a fixed point of the best-response mapping (i.e., a GNE) or until a maximum number of iterations is reached. Despite the difficulty of analyzing existence and convergence properties [78], these methods remain widely used in practice to approximate GNEs [51], [52], [64].

We now formalize the strategy computed by the attacker.

Definition 7 (Regulation-Aware Game-Theoretic Planner (RA-GTP))**.** We define the *Regulation-Aware Game-Theoretic Planner (RA-GTP)* for the attacker as the strategy $w^{\mathrm{A}\star}(t)$ obtained from the IBR approximation of the GNE of the game $\mathcal{G}(t)$, applied in a receding horizon fashion.

Although the RA-GTP is here defined for the attacker, the formulation extends naturally to any player.

4.7 Simulation Studies

The simulation studies are conducted on the Autodromo di Modena racetrack layout. Both the attacker and the defender are assumed to follow the same racing line. Each vehicle follows a time-optimal speed profile subject to bounds on longitudinal and lateral accelerations. To encourage overtaking maneuvers, the attacker is assigned a reference speed profile with higher longitudinal and lateral accelerations compared to the defender.

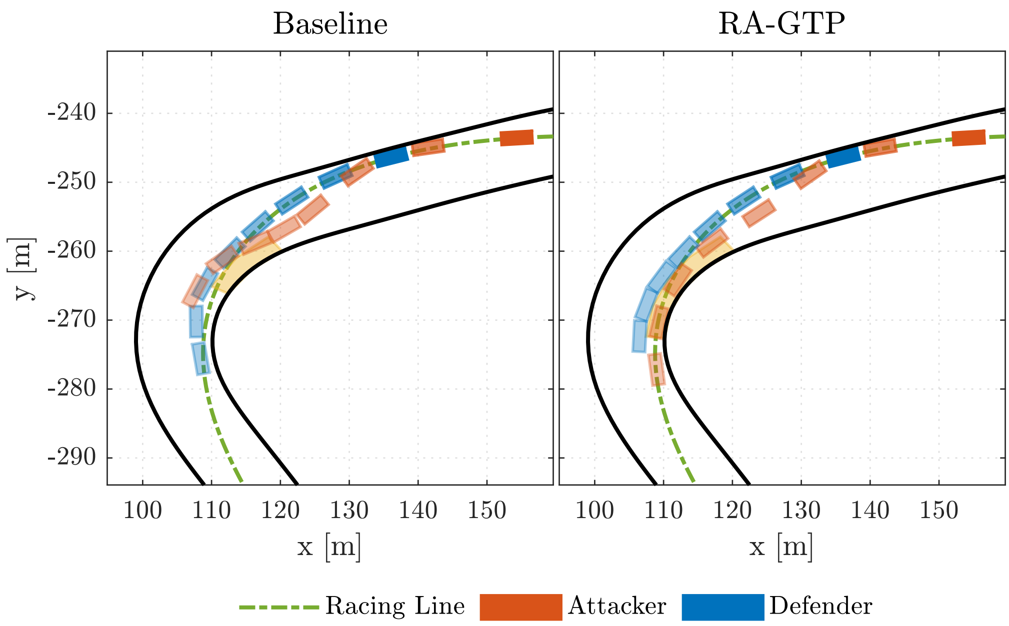


Figure 4.2. Comparison of regulation-agnostic (baseline) and regulation-aware (RA-GTP) planning during an inner-pass attempt. Overlay of attacker and defender positions over time. The yellow shaded region denotes the lateral space the defender is required to yield once the attacker acquires right-of-way. The baseline attacker aborts the maneuver due to excessive conservatism. RA-GTP attacker, reasoning on regulation-constrained opponent behavior, completes the inner pass by exploiting the granted space.

The motion planning problems for both the attacker and the defender are formulated over a prediction horizon of $N = 20$ steps, with a sampling time of $T_s = 0.05\,\mathrm{s}$. Both planning schemes rely on the discretized kinematic bicycle model in Appendix A.2. The right-of-way and collision avoidance thresholds are summarized in Table 4.1.

Table 4.1. Right-of-way and collision-avoidance thresholds expressed as multiples of the vehicle dimensions.

Parameter	**Value**	**Parameter**	**Value**
Δ_s^{row}	2.0× car length	Δ_s^{CA}	1.5× car length
Δ_n^{row}	0.5× car width	Δ_n^{CA}	1.5× car width
Δ_g^{row}	1.5× car width		

The MI-SQP algorithm is implemented in Python, using `CasADi` [80] for the computation of the quadratic approximations of the original MINLP, and `Gurobi` [81] as the MIQP solver. The IBR iterations are initialized using the solution from the previous planning step.

The simulation studies compare two planning schemes for the attacker: a baseline method and the proposed RA-GTP. In both cases, the defender executes a RC-MPC strategy, meaning it adheres to the race regulations. To this end, the defender is provided with the attacker's predicted trajectory over the entire horizon, while the attacker observes only the defender's current state. The baseline method solves the

IBR problem using a defender model that does not account for regulation compliance; equivalently, the baseline assumes a fixed prediction of the opponent's trajectory.

To illustrate both the regulation enforcement within the RC-MPC and the regulation-aware reasoning mechanisms of the proposed RA-GTP, we first analyze in detail a representative *inner-pass* maneuver between an attacking and a defending vehicle. This example provides a step-by-step view of how right-of-way constraints are evaluated within the players' RC-MPC problem. The subsequent figures describe the evolution of binary indicators and constraint activations throughout this maneuver. We then conclude the section with a statistical comparison over a larger set of randomized simulations, highlighting the overall performance gains achieved by the RA-GTP relative to the baseline planner.

Baseline vs Regulation-Aware Planning

The considered scenario reproduces an inner pass attempt, where the attacking vehicle must decide whether to commit to the overtaking maneuver while anticipating the opponent's response under race regulations. Fig. 4.2 illustrates the resulting behavior for the two planners. In the baseline, the attacker assumes a regulation-agnostic model of the opponent, enforcing collision avoidance but disregarding the defender's yielding obligation once right-of-way is acquired; thus, the resulting behavior remains overly conservative, leading to the abort of the overtaking maneuver. In contrast, the RA-GTP explicitly reasons on the regulatory asymmetry: when the attacker anticipates acquiring right-of-way, it predicts that the defender will yield the corresponding lateral space. This reasoning allows the attacker to complete the inner pass successfully, without compromising safety, which is enforced through the collision-avoidance constraints.

Defender Behavior and Regulation Activation

Fig. 4.3 shows the same inner-pass scenario introduced above, now represented in the Frenet frame. Both the attacker and defender trajectories, denoted by $\mathbf{q}^{\mathrm{A}}$ and $\mathbf{q}^{\mathrm{D}}$, are plotted together with the corresponding crossing-position trajectories $\bar{\mathbf{q}}^{\mathrm{A}}$ and $\bar{\mathbf{q}}^{\mathrm{D}}$, extracted from the defender extended state $\mathbf{x}^{\mathrm{D}}$ as defined in (4.34). It can be observed that the sample-and-hold dynamics (4.26) maintain the crossing positions fixed whenever the longitudinal gap between the vehicles falls below the right-of-way threshold Δ_s^{row}. Here, the crossing points represent the spatial locations used by the defender to evaluate its regulation constraints, according to Definition 3.

Fig. 4.4 details the evolution of the defender's binary variables associated to the relative positioning of the vehicles. When the relative longitudinal distance $(s^{\mathrm{D}} - s^{\mathrm{A}})$ falls below the threshold Δ_s^{row} (top panel), the binary variable $\delta_{s^+}^{\mathrm{row}}$ is activated (middle panel), indicating that the attacker has closed the longitudinal gap. Similarly, when the relative crossing lateral position $(\bar{n}^{\mathrm{D}} - \bar{n}^{\mathrm{A}})$ becomes smaller than $-\Delta_n^{\mathrm{row}}$, the binary variable $\delta_{n^+}^{\mathrm{row}}$ equals 1 (bottom panel), signaling that the attacker is performing an overtaking maneuver on the left side. These indicators jointly determine the activation of the yielding constraints within the defender's RC-MPC problem.

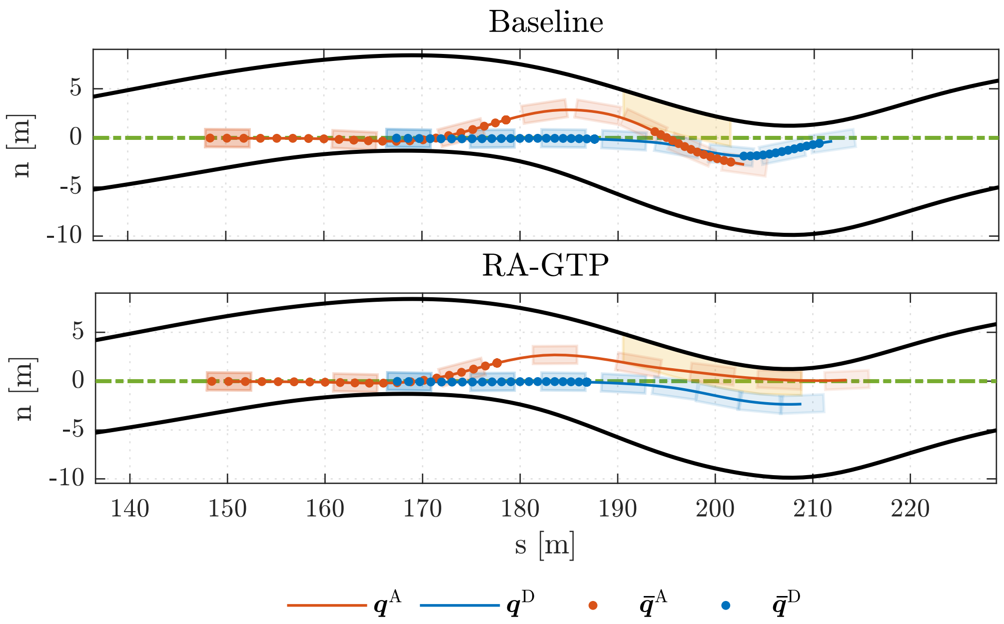


Figure 4.3. Comparison of the inner-pass maneuver under the baseline and RA-GTP, represented in the Frenet frame. The trajectories of the attacker ($\mathbf{q}^{\mathrm{A}}$) and defender ($\mathbf{q}^{\mathrm{D}}$) are shown together with the corresponding crossing positions ($\bar{\mathbf{q}}^{\mathrm{A}}$, $\bar{\mathbf{q}}^{\mathrm{D}}$). The sample-and-hold dynamics preserve the crossing positions whenever the longitudinal gap is below the right-of-way threshold Δ_s^{row}, which are used by the defender to evaluate the regulation constraints.

Fig. 4.5 illustrates the activation of the defender's lateral-position constraints. The top panel shows the defender's lateral position n^{D}, together with the nominal track boundaries n_ℓ and n_r, and the yielding boundaries $n_\ell - z_{g\ell}$ and $n_r + z_{gr}$. The shaded area represents the range of lateral positions n^{D} that are compliant with the right-of-way constraint $\boldsymbol{\gamma}^{\mathrm{row}}$ in (4.21). The bottom panel reports the binary indicators δ_ℓ and δ_r, corresponding to the defender's obligation to yield to the left or to the right. It can be observed that when δ_ℓ transitions to one, the admissible region shrinks accordingly and the active lateral boundary switches from the nominal left track limit n_ℓ to the yielding boundary $n_\ell - z_{g\ell}$, enforcing the spatial clearance required by the attacker's right-of-way.

Attacker Constraints and Collision Avoidance

Fig. 4.6 shows the evolution of the attacker's relative positions and binary indicators. The top panel shows the relative longitudinal and lateral distances ($s^{\mathrm{A}} - s^{\mathrm{D}}$) and ($n^{\mathrm{A}} - n^{\mathrm{D}}$), respectively. The middle and bottom panels report the corresponding longitudinal and lateral binary indicators: $\delta_{s+}^{\mathrm{row}}$, $\delta_{s-}^{\mathrm{row}}$ for the longitudinal direction, and $\delta_{n+}^{\mathrm{row}}$, $\delta_{n-}^{\mathrm{row}}$ for the lateral direction.

It can be observed that as the longitudinal gap ($s^{\mathrm{A}} - s^{\mathrm{D}}$) falls below the collision-avoidance threshold Δ_s^{CA}, the associated indicator $\delta_{s+}^{\mathrm{row}}$ deactivates, while the lateral indicator $\delta_{n+}^{\mathrm{row}}$ simultaneously activates, corresponding to the increase of lateral separation above the threshold Δ_n^{CA}. Thus, at any time, at least one binary variable remains active, ensuring that a separation constraint is always enforced, as imposed by the MLD formulation in (4.29).

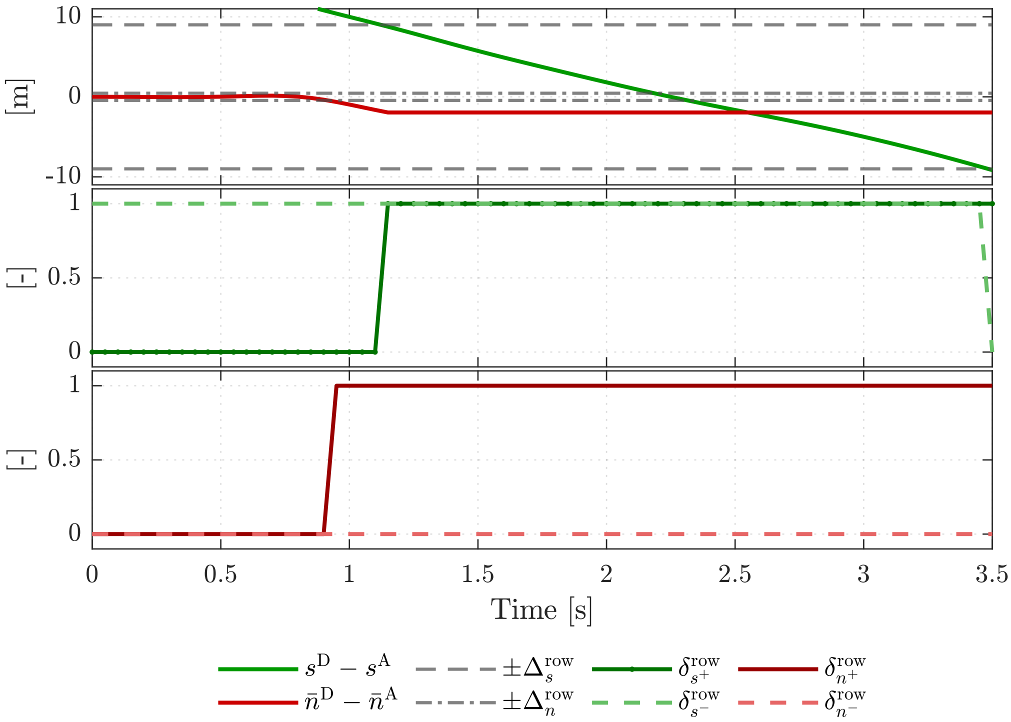


Figure 4.4. Evolution of the defender's binary indicators for the relative vehicles positioning. Top panel shows the relative longitudinal position and relative crossing lateral position. Middle and bottom panels show how the binary indicators $\delta_{s+}^{\mathrm{row}}$ and $\delta_{n+}^{\mathrm{row}}$ detect when the longitudinal gap and lateral offset fall below Δ_s^{row} and rise above Δ_n^{row}, respectively. The activation of these binaries identifies the onset of the attacker's left overtaking maneuver.

Overtaking Performance Comparison

The performance of the two planning schemes, i.e, baseline and RA-GTP, are summarized across multiple randomized test cases (Fig. 4.7). Each marker represents the track location where the attacker either successfully completed an overtaking maneuver (*success*) or failed the attempt (*abort*). The analysis is based on 100 simulation runs, where the attacker is initialized at uniformly sampled positions along the racing line, and the defender is placed at a fixed longitudinal offset ahead. Both vehicles start at their respective reference speed profiles. Overtaking points are recorded when the longitudinal gap between the vehicles drops below the right-of-way threshold Δ_s^{row}. A case is labeled as a *success* if the attacker subsequently moves ahead by at least Δ_s^{row}, or as an *abort* if the gap increases again beyond this threshold without completing the maneuver.

Overall, the comparison highlights that RA-GTP achieves a higher success rate (96 %) and more consistent overtaking maneuvers compared to the baseline approach (29 %). Specifically, the baseline is prone to abort overtaking attempts in those regions of the track where the racing line moves laterally across the track width, thus (erroneously) predicting the defender to shift sides and preventing the attacker from executing a safe pass. In contrast, RA-GTP correctly predicts when lateral space will be granted by reasoning over the defender's right-of-way constraints, thereby enabling successful overtaking maneuvers in regions where the baseline would abort. A video recording of selected maneuvers can be found in the accompanying video [82].

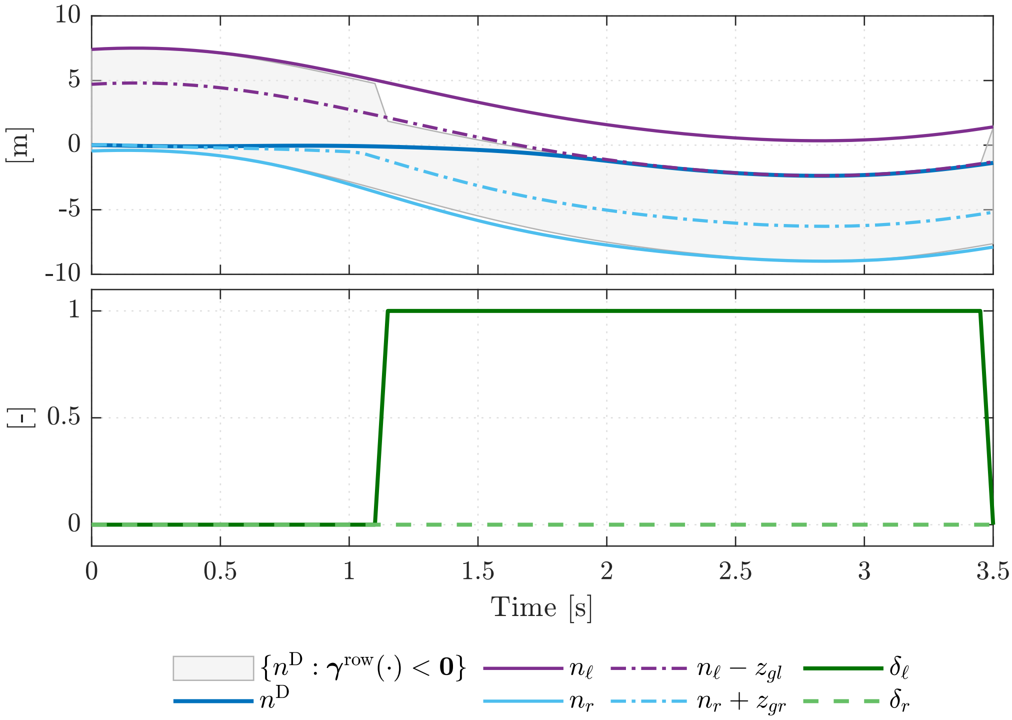


Figure 4.5. Defender lateral-position constraints and yielding indicators during the inner-pass maneuver. On the top panel, the defender lateral position n^{D} with nominal track boundaries n_ℓ and n_r, yielding boundaries $n_\ell - z_{g\ell}$ and $n_r + z_{gr}$, and shaded region denoting values of n^{D} compliant with the right-of-way constraint $\boldsymbol{\gamma}^{\mathrm{row}}$. On the bottom, the binary indicators δ_ℓ and δ_r, enforcing left and right yielding obligations. When δ_ℓ becomes active, the admissible region switches to the left-yielding bound $n_\ell - z_{g\ell}$.

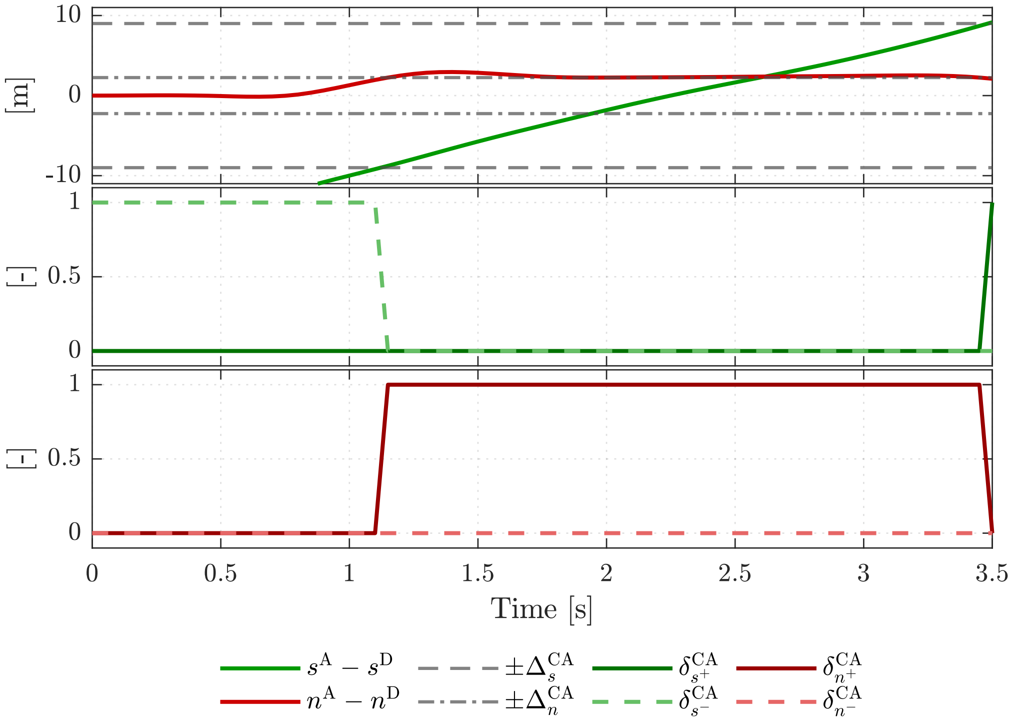


Figure 4.6. Attacker relative distances and binary indicators during the inner-pass maneuver. When the longitudinal gap falls below the collision-avoidance threshold Δ_s^{CA} (top panel), the corresponding longitudinal indicator deactivates (middle panel), while the lateral indicator δ_{n+}^{CA} activates (bottom panel) as the lateral distance increases beyond Δ_n^{CA}. At least one separation constraint remains active at all times, in accordance with the MLD collision-avoidance constraints (4.29).

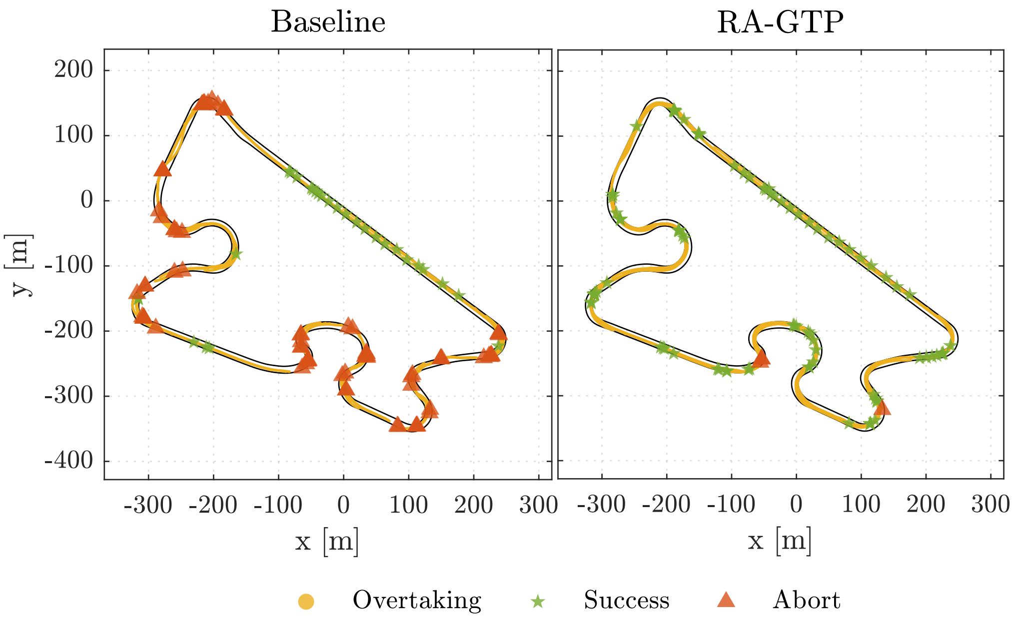


Figure 4.7. Overtaking outcomes across 100 randomized simulations, showing ongoing, aborted, and completed maneuvers. The baseline achieves a 29 % success rate, while the proposed RA-GTP attains 96 %, demonstrating the advantage of regulation-aware reasoning.

4.8 Summary and Discussion

This chapter addressed the problems of regulation compliance and strategic overtaking in motion planning for autonomous racing. The proposed a Regulation-Compliant Model Predictive Control (RC-MPC) formulation enforces right-of-way rules and collision avoidance using Mixed Logical Dynamical constraints. Building on this, the Regulation-Aware Game-Theoretic Planner (RA-GTP) enables the attacker to reason over the defender's regulation-constrained behavior via an Iterative Best Response approximation of a Generalized Nash Equilibrium.

Simulation results demonstrate that RA-GTP achieves a higher overtaking success rate compared to baseline methods based on non-interactive opponent models. In particular, the proposed planner successfully exploits rule-compliant behavior to execute overtakes in scenarios where fixed-prediction approaches are prone to unnecessary aborts. These results highlight the importance of integrating regulatory awareness into motion planning to reduce conservatism and enable strategic reasoning.

In contrast to prior works that model multi-agent interactions through symmetric collision-avoidance constraints shared among all players, the proposed formulation assigns *asymmetric responsibility* to each agent. This structure leads to interactions of higher realism, reflecting how regulations serve as *tie-breaking mechanisms*, preventing deadlocks and enhancing overtaking opportunities, besides discouraging unfair or unsafe maneuvers.

As a drawback, the proposed approach introduces significant computational complexity. Both the mixed-integer formulation of the RC-MPC and the game-theoretic layer contribute to this aspect, as the latter requires iterative evaluation of each agent's optimization problem. Moreover, studying the convergence properties of the IBR scheme and the existence of a GNE remains an open problem in the presence of nonconvexities, including those introduced by mixed-integer regulatory constraints.

The validity of the underlying modeling assumptions also deserves discussion. First, the opponent's nominal racing line is assumed to be known, which is a fair approximation in competitive racing, where all vehicles typically converge toward a time optimal racing line, or to a few variants. Similarly, the opponent's vehicle model is assumed known, which is reasonable as a first approximation given comparable vehicle dynamics among players. Finally, the approach relies on the assumptions that the opponent behaves rationally and complies with racing regulations, which are implicitly embedded in its cost function and constraint set. While these assumptions may be acceptable in structured racing contexts, relaxing them represents an important direction for future research toward more general and uncertain interaction scenarios.

Chapter 5

Real-Time Regulation-Aware Game-Theoretic Planning

5.1 Introduction

This chapter addresses Research Question 3, which investigates how real-time feasibility can be achieved for regulation-based predictive motion planning in competitive multi-agent racing scenarios. The objective is to develop computational strategies that preserve the predictive and strategic structure of regulation-aware planning while enabling reliable real-time operation.

As discussed in Chapter 4, in head-to-head racing, trajectory planning must balance competitive performance with adherence to race regulations, while reacting strategically to the opponent's behavior. Traditional predict-then-plan approaches often yield conservative maneuvers, as they fail to anticipate how opponents might adapt to the ego vehicle's planned motion. Game-theoretic formulations overcome this limitation by explicitly modeling mutual interactions between agents, enabling competitive yet safe strategies [52]–[54].

To this aim, Chapter 4 introduced a *Regulation-Compliant Model Predictive Control* (RC-MPC) scheme that encoded racing regulations, such as *right-of-way*, using the Mixed Logical Dynamical (MLD) framework [47]. This integration of logical regulations and continuous vehicle dynamics within a unified predictive control scheme enabled rule-compliant trajectory planning. Furthermore, a *Regulation-Aware Game-Theoretic Planner* (RA-GTP) was developed to compute Generalized Nash Equilibria (GNE) [42] of a regulation-compliant racing game, enabling the ego vehicle to anticipate the opponent's constrained behavior and exploit regulatory asymmetries strategically. By explicitly embedding rule-based responsibilities into the game formulation, the RA-GTP mitigates the excessive conservatism of baseline planners that assume regulation-agnostic opponents, leading to a significant higher rate of successful overtaking maneuvers. However, despite its demonstrated effectiveness, the original RA-GTP formulation presents substantial computational challenges, which limit its real-time applicability and, consequently, its deployment in real-world racing scenarios.

This chapter presents a computationally efficient reformulation, referred to as *Fast RA-GTP*, which achieves real-time performance while preserving the essential game-theoretic and regulation-aware structure. The proposed framework introduces three main advancements:

- adoption of a computational strategy inspired by the Real-Time Iteration (RTI) scheme for nonlinear MPC [83], applying its single-iteration philosophy within a Mixed-Integer Sequential Quadratic Programming (MI-SQP) solver [48] to efficiently handle the Mixed-Integer Nonlinear Program (MINLP) underlying the RC-MPC;

- approximation of the Generalized Nash Equilibrium Problem (GNEP) solution through a single-pass Iterative Best-Response (IBR) update at each control step, enabling rapid adaptation to the opponent's latest predicted motion;

- simplification of the defender's MLD formulation by precomputing a subset of binary variables based on the latest predicted trajectories, reducing the problem size while retaining the essential interaction logic.

Simulation and experimental validation on the 1:10-scale *Berkeley Autonomous Race Car* (BARC) platform demonstrate that the proposed Fast RA-GTP achieves real-time feasibility and executes regulation-compliant overtaking maneuvers that are unattainable for a regulation-agnostic baseline, which remains overly conservative and fails to initiate passing attempts.

The remainder of this chapter is organized as follows. Section 5.2 recall the Regulation-Compliant MPC and Regulation-Aware Game-Theoretic Planner. Section 5.3 introduces the MI-SQP framework and its adaptation for mixed-integer predictive planning. Section 5.4 describes the single-pass IBR approximation for real-time GNE computation. Section 5.5 presents the model simplifications and precomputation strategy. Simulation and experimental results are discussed in Section 5.6, followed by conclusions in Section 5.7.

The material presented in this chapter builds upon the work

- *"Real-Time Regulation-Aware Game-Theoretic Motion Planning for Head-to-Head Autonomous Racing,"* by F. Prignoli, S. Cao, P. Falcone, and F. Borrelli, under review at the *IEEE Transactions on Control Systems Technology* (2025).

5.2 Background

The interaction during an overtake maneuver is subject to a set of constraints introduced Section 4.4, which define the conditions under which a vehicle (the *attacker*) gains the right-of-way and the corresponding obligations of the opponent (the *defender*). According to these regulations, the attacker gains the right-of-way when, at a longitudinal distance of Δ_s^{row} from the defender, its relative lateral displacement with respect to the defender exceeds a prescribed margin Δ_n^{row}. The sign of this relative displacement indicates the overtaking side. Once this condition is met, the defender is required to yield by maintaining at least Δ_g^{row} of lateral clearance on the overtaking side, provided such space was already available when the attacker obtained the right-of-way. Throughout the maneuver, the attacker remains responsible for avoiding collisions.

To enforce these regulations within a predictive control framework, Chapter 4 formulated a *Regulation-Compliant Model Predictive Control* (RC-MPC) scheme based on the MLD framework. The MLD formulation integrates continuous vehicle dynamics with logical constraints defining the overtaking rules. The logical constraints are expressed as functions of binary variables, indicating the acquisition of the right-of-way and the obligation to yield, together with continuous auxiliary variables representing the available lateral space. These logical and continuous constraints are coupled with the vehicle's dynamic model in a unified Mixed-Integer Nonlinear Program (MINLP). By design, the RC-MPC produces trajectories that satisfy dynamic feasibility, safety constraints, and overtaking rules, thereby enabling regulation-compliant trajectories in head-to-head racing scenarios. For completeness, we recall the formulation of the RC-MPC for the player i as the following optimal control problem.

Problem 2 (Regulation-Compliant MPC (RC-MPC))**.**

$$
\begin{aligned}
\min_{X^i,U^i,\Delta^i,Z^i} \quad & J^i(X^i,U^i) && && (5.1a)\\
\text{s.t.} \quad & \mathbf{x}^i_{k+1} = \mathbf{f}^i(\mathbf{x}^i_k,\mathbf{u}^i_k,\mathbf{z}^i_k,\mathbf{x}^{-i}_k), && k \in \mathbb{I}_0^{N-1} && (5.1b)\\
& \mathbf{x}^i_0 = \mathbf{x}^i(t), && && (5.1c)\\
& \mathbf{g}^i_N(\mathbf{x}^i_N) \leq \mathbf{0}, && && (5.1d)\\
& \mathbf{g}^i_{x,u}(\mathbf{x}^i_k,\mathbf{u}^i_k) \leq \mathbf{0}, && k \in \mathbb{I}_0^{N-1} && (5.1e)\\
& \boldsymbol{\gamma}^i(\mathbf{x}^i_k,\boldsymbol{\delta}^i_k,\mathbf{z}^i_k,\mathbf{x}^{-i}_k) \leq \mathbf{0}, && k \in \mathbb{I}_0^{N} && (5.1f)
\end{aligned}
$$

The optimization variables include the predicted states $X^i \triangleq \{\mathbf{x}^i_k\}_{k=0}^N$, inputs $U^i \triangleq \{\mathbf{u}^i_k\}_{k=0}^{N-1}$, regulation-encoding binaries $\Delta^i \triangleq \{\boldsymbol{\delta}^i_k\}_{k=0}^N$, and auxiliary variables $Z^i \triangleq \{\mathbf{z}^i_k\}_{k=0}^N$. The cost function (5.1a) penalizes deviations from the racing line, speed profile tracking error, and control effort. Constraints (5.1b) enforces the vehicle dynamics, (5.1c) the measured initial condition, (5.1d) the terminal condition, and (5.1e) state and input bounds. Finally, the regulation-compliance constraints (5.1f) enforce the player-specific overtaking rules, i.e., collision avoidance for the attacker and yielding obligations for the defender, and depend explicitly on the predicted state trajectory of the opponent, $\mathbf{x}^{-i}_k$.

Building upon the RC-MPC (5.1), we model the interaction between the two vehicles as a noncooperative dynamic game. In this setting, each player solves its own RC-MPC problem over a finite horizon, taking into account the predicted trajectory of the opponent. The resulting optimization problems are coupled through regulation-compliance constraints (5.1f), which depend on the opponent's predicted states. Due to this coupling, each player's feasible set depends on the other player's strategy. For completeness, the formal definition of the *Regulation-Constrained Racing Game* introduced in Definition 5 is recalled below.

Definition 8 (Regulation-Constrained Racing Game)**.** We define the *Regulation-Constrained Racing Game* at time t as:

$$
\mathcal{G}(t) \triangleq \left\langle \mathcal{I},\, \mathcal{P}^i(\mathbf{x}^i(t), X^{-i}(t)) \right\rangle_{i\in\mathcal{I}}, \tag{5.2}
$$

where:

- $\mathcal{I} \triangleq \{\mathrm{A}, \mathrm{D}\}$ is the set of players (attacker and defender);
- $\mathbf{x}^i(t)$ is the state of player $i \in \mathcal{I}$ at current time t;
- $X^{-i}(t)$ is the predicted state trajectory of the opponent over the horizon N;
- $\mathcal{P}^i(\mathbf{x}^i(t), X^{-i}(t))$ is the RC-MPC problem (5.1) solved by agent i, initialized at $\mathbf{x}^i(t)$, and conditioned on $X^{-i}(t)$.

In Chapter 4, the game was approximately solved through an IBR scheme, in which players alternately solve their optimization problems using the most recent prediction of the opponent's trajectory. This iterative process converges to an approximate GNE for the game $\mathcal{G}(t)$, which is then executed in a receding-horizon fashion. We now extend the *Regulation-Aware Game-Theoretic Planner* (RA-GTP), originally defined for the attacker in Definition 7 to a general player.

Definition 9 (Regulation-Aware Game-Theoretic Planner (RA-GTP))**.** Let $w^i \triangleq (X^i, U^i, Z^i, \Delta^i)$ denote the strategy of player $i \in \mathcal{I}$. The *Regulation-Aware Game-Theoretic Planner (RA-GTP)* is a receding-horizon planning algorithm that, at each time instant t, runs an IBR scheme to approximate a GNE for the racing game in (5.2), and outputs for player i its equilibrium strategy $w^{i\star}(t)$.

5.3 RTI-MISQP for Regulation-Compliant MPC

The RC-MPC formulation in (5.1) results in a MINLP, due to the nonlinear vehicle dynamics and the presence of binary variables encoding overtaking regulations. This problem can be written in compact form as

$$\begin{aligned} \min_{y,\,\delta} \quad & f(y) && \text{(5.3a)}\\ \text{s.t.} \quad & g(y) + G\delta \leq 0, && \text{(5.3b)}\\ & h(y) + H\delta = 0, && \text{(5.3c)}\\ & \delta \in \{0,1\}^m, && \text{(5.3d)} \end{aligned}$$

where $y \in \mathbb{R}^n$ collects the continuous decision variables (predicted states, inputs, and auxiliary variables), while δ denotes the binary variables encoding the logical conditions. The matrices G, H capture the affine dependence of the constraints on δ. Although the binary variables enter linearly in the constraints, solving the resulting MINLP to optimality is generally NP-hard and therefore unsuitable for real-time racing applications.

We first consider a Mixed-Integer Sequential Quadratic Programming (MISQP) approach [48], which extends the Sequential Quadratic Programming paradigm to handle mixed-integer variables. At each MISQP iteration, the nonlinear dynamics and constraints are linearized around the current iterate, yielding a Mixed-Integer Quadratic Program (MIQP). Solving this MIQP provides simultaneous updates for both continuous and discrete decision variables, resulting in a tractable approximation of the original MINLP. At each MISQP iteration ℓ, the nonlinear inequality

Algorithm 1 RTI-MISQP for RC-MPC

Input: Current state x_{meas}, previous solution $(y^{\mathrm{prev}}, \delta^{\mathrm{prev}})$
Output: Control input $u_0^\star$

1: **Warm-start:** Shift the previous solution to obtain initial guess (y^0, δ^0), with $y_0^0 \leftarrow x_{\mathrm{meas}}$.
2: **Linearization:** Around y^0, compute Jacobians $\nabla g(y^0), \nabla h(y^0)$ and Hessian approximation B^0.
3: **MIQP subproblem:** Formulate and solve the MIQP in (5.4) obtaining solution $(\Delta y^\star, \delta^\star)$.
4: **Update:** Set $y^1 \leftarrow y^0 + \Delta y^\star, \quad \delta^1 \leftarrow \delta^\star$.
5: **Return:** First control input $u_0^\star$ extracted from y^1.

and equality constraints (5.3b)-(5.3c) are linearized with respect to the continuous variables y around the current iterate y^ℓ, while the binary dependence is preserved exactly. This yields the MIQP subproblem

$$\begin{aligned} \min_{\Delta y,\, \delta} \quad & \nabla f(y^\ell)^\top \Delta y + \tfrac{1}{2} \Delta y^\top B^\ell \Delta y \\ \text{s.t.} \quad & g(y^\ell) + \nabla g(y^\ell)\, \Delta y + G\delta \leq 0, \\ & h(y^\ell) + \nabla h(y^\ell)\, \Delta y + H\delta = 0, \\ & \delta \in \{0,1\}^m, \end{aligned} \tag{5.4}$$

where B^ℓ is a (possibly regularized) approximation of the Lagrangian Hessian. The solution $(\Delta y^\star, \delta^\star)$ defines the next iterate as

$$y^{\ell+1} \leftarrow y^\ell + \alpha^\ell \Delta y^\star, \quad \delta^{\ell+1} \leftarrow \delta^\star, \tag{5.5}$$

with $\alpha^\ell \in (0,1]$ denoting the step length, typically chosen via a merit-function line search [48]. The MISQP algorithm iterates until convergence criteria such as small KKT residuals, a sufficient decrease of the merit function, or a maximum number of iterations are met. In practice, limiting the number of iterations can improve tractability but does not yet systematically guarantee real-time feasibility.

Inspired by the RTI scheme for SQP [83], we adopt an analogous approach in the mixed-integer setting. Instead of iterating to convergence, our RTI-MISQP implementation executes exactly one MISQP step per control cycle, warm-started from the shifted solution of the previous cycle. This single-step strategy significantly reduces computation time while retaining the ability to adapt to the most recent state and parameter measurements. The RTI-MISQP procedure is summarized in Algorithm 1.

It is important to note that, due to the nonconvexity of the integrality constraints (5.3d) the step direction obtained from the MIQP at each MISQP iteration is not, in general, a descent direction for the merit function of the original MINLP. This occurs because small changes in the binary variables can lead to large and discontinuous variations in the constraint violation terms. However, under the assumption that the binary variables remain fixed at their updated value, the continuous step direction can be shown to be a descent direction for the associated continuous NLP [48]. This property enables the use of standard globalization strategies, such as line search, to enforce progress and stability in full MISQP iterations. In contrast, in the proposed RTI-style implementation, only a single MIQP solve is performed per control cycle, and no explicit merit-function descent is enforced. Instead, we rely on warm-starting from the previous cycle and the temporal consistency of successive MPC problems to obtain feasible and practically effective solutions in real time.

Key practical aspects of the RTI MI-SQP implementation are discussed below:

(i) *Feasibility.* At each time instant t, the RTI scheme is warm-started from the previously computed solution at $t-1$, shifted forward in time and updated using the current measured state. Since this shifted solution may violate some constraints due to disturbances or tightened bounds at time t, slack variables are introduced to soften selected constraints. This ensures that the resulting MIQP subproblem remains feasible and solvable at every control step.

(ii) *Binary stability.* The performance of the warm-started MISQP is enhanced when binary decisions remain consistent across consecutive cycles, which is typically the case in the considered racing scenario due to gradual changes in the relative vehicle states. Large binary changes, such as switching the overtaking side, are rare with a sufficiently long prediction horizon but may cause transient suboptimality.

(iii) *Optimality gap.* The RTI-MISQP sacrifices per-cycle optimality for computational speed, but successive re-optimizations in closed loop keep the system close to optimal trajectories, as confirmed in our experiments.

The RTI-MISQP formulation thus enables efficient real-time solutions of the regulation-compliant MPCs for each player. In the following, this solver is integrated within a single-pass IBR framework to approximate the GNE of the racing game in real time.

5.4 Single-pass IBR for RA-GTP

To approximate the GNE of the regulation-constrained racing game (5.2), we adopt an IBR scheme (Algorithm 2). In this approach, players sequentially solve their own RC-MPC problems, formulated as MINLPs and solved via the RTI-MISQP described in Algorithm 1, using the most recent prediction of the opponent's trajectory as a fixed parameter. The IBR loop follows a Gauss-Seidel update order, where the opponent is updated first and the ego computes its best response using the opponent's most recent plan. Convergence is checked twice per iteration, i.e., after the opponent update and after the ego update, to allow early termination as one of the trajectories stabilizes, since further updates would be redundant in this case. The opponent convergence check is skipped at the first iteration ($\ell = 0$) to ensure that the ego is always updated at least once per planning cycle.

In the present work, motivated by real-time constraints, we restrict the IBR scheme of Algorithm 2 to a *single pass* per control cycle, implemented by setting $N_{\mathrm{IBR}} = 1$. This corresponds to one opponent update followed by one ego update at each control step:

1. Opponent's Best Response: The opponent solves its optimal control problem, assuming the ego agent's trajectory from the previous step is fixed, yielding a

Algorithm 2 Iterative Best Response (IBR) for the Racing Game

Input: Measured states $\mathbf{x}^{i}_{\text{meas}}$, $\mathbf{x}^{-i}_{\text{meas}}$; previous ego plan $(\hat{X}^i, \hat{U}^i)$; number of passes $N_{\text{IBR}} \geq 1$; tolerances $\varepsilon_x, \varepsilon_u$

Output: Ego first control input u^i_0

Warm-start ego: $(X^{i,0}, U^{i,0}) \leftarrow$ shifted $(\hat{X}^i, \hat{U}^i)$
for $\ell = 0$ **to** $N_{\text{IBR}} - 1$ **do**
  Opponent update: $(X^{-i,\ell+1}, U^{-i,\ell+1}) \leftarrow \mathcal{P}^{-i}(\mathbf{x}^{-i}_{\text{meas}}, X^{i,\ell})$
  if $\ell > 0$ **and** $\|X^{-i,\ell+1} - X^{-i,\ell}\|_\infty \leq \varepsilon_x$ **and** $\|U^{-i,\ell+1} - U^{-i,\ell}\|_\infty \leq \varepsilon_u$ **then**
    $U^{i,\ell+1} \leftarrow U^{i,\ell}$
    break
  Ego update: $(X^{i,\ell+1}, U^{i,\ell+1}) \leftarrow \mathcal{P}^{i}(\mathbf{x}^{i}_{\text{meas}}, X^{-i,\ell+1})$
  if $\|X^{i,\ell+1} - X^{i,\ell}\|_\infty \leq \varepsilon_x$ **and** $\|U^{i,\ell+1} - U^{i,\ell}\|_\infty \leq \varepsilon_u$ **then**
    break
Return $u^i_0 \leftarrow U^{i,\ell+1}_0$

predicted trajectory over the planning horizon.

2. Ego's Best Response: The ego agent then solves its own optimal control problem, using the opponent's newly predicted trajectory as fixed.

While increasing the number of IBR passes per control cycle ($N_{\text{IBR}} > 1$) can improve the per-cycle equilibrium approximation, it also increases computational cost. In the proposed single-pass IBR scheme, the resulting solutions may be suboptimal at each cycle but can still approach equilibrium behavior over successive cycles.

The following observations summarize practical aspects of the proposed single-pass IBR scheme:

(i) *Convergence.* The IBR scheme corresponds to a Gauss–Seidel method applied to the coupled optimality conditions of the players' decision problems [78]. Convergence to a GNE is only guaranteed for certain classes of convex and monotone games [42]. In the regulation-constrained racing game considered here, each player solves a nonconvex MINLP, and therefore such convergence guarantees do not hold. Nevertheless, temporal coherence and warm-starting empirically yield stable closed-loop behavior.

(ii) *Initialization.* For nonconvex games, the final equilibrium may depend on the initial strategies. In the MPC context, initialization is naturally provided by the shifted solution from the previous control cycle, promoting convergence to consistent trajectories across adjacent cycles.

(iii) *Asymmetry.* The regulation constraints introduce an inherent asymmetry between the players' feasible sets, which reduces the likelihood of cycling between multiple equilibria. This asymmetry introduce a "stabilizing effect" promoting a more consistent convergence behavior in practice.

(iv) *Practical stability.* Despite the lack of theoretical convergence guarantees in this nonconvex setting, closed-loop stability is supported by warm-starting, short sampling times, gradual state evolution, and the “stabilizing" effect of asymmetric regulation constraints.

In summary, the proposed single-pass IBR settings provides a computationally efficient means of approximating the GNE of the regulation-constrained racing game in real time, leveraging the RTI-MISQP solver for each player's RC-MPC problem. While the method sacrifices per-cycle optimality by limiting the number of iterations to a single Gauss–Seidel sweep, closed-loop execution remains effective owing to warm-started trajectories that lie close to the optimal solution, thereby preserving strategic interaction and regulation compliance under stringent timing constraints.

5.5 Defender Model Simplification

To further reduce computational requirements, we introduce a simplification of the defender's RC-MPC formulation presented in Chapter 4 by pre-computing a subset of its regulation-encoding binary variables before solving the optimization problem. In particular, the binaries that determine the lateral space to be granted to the attacker are fixed at each planning step based on the most recent predicted trajectories of both players. By removing these variables from the defender's optimization problem, the number of free binary decision variables is reduced, decreasing the complexity of the resulting MIQP subproblems and improving solver runtimes.

The remaining binary variables $\delta_{s^-}, \delta_{s^+},$ and δ_s represent indicators of the longitudinal interaction, implying that the defender optimizes its longitudinal motion to delay or anticipate the occurrence of such interaction. These indicators are defined as $h_{s^\pm} : \mathbb{R}^2 \to \mathbb{R}$:

$$h_{s^\pm}(s_\mathrm{D}, s_\mathrm{A}) \triangleq \pm s_\mathrm{D} \mp s_\mathrm{A} - \Delta_s^\mathrm{row}, \tag{5.6}$$

with the logical relations

$$[\delta_{s^\pm} = 1] \iff [h_{s^\pm}(s_\mathrm{D}, s_\mathrm{A}) \leq 0], \tag{5.7a}$$

$$\delta_s = \delta_{s^+} \wedge \delta_{s^-}, \tag{5.7b}$$

where $s_\mathrm{D}, s_\mathrm{A}$ denote the arc-length coordinates of the defender and the attacker in the Frenet frame, respectively.

As a result of this simplification, the defender model no longer includes the sample-and-hold dynamics for the joint crossing position $\bar{\boldsymbol{\eta}}$ included in (4.26), nor the associated auxiliary and binary variables. This reduction removes four states, six auxiliary variables, and six binary variables from the defender's RC-MPC formulation, resulting in a smaller and faster MINLP.

This pre-computation, however, removes the defender's ability to adjust its planned trajectory in order to influence the lateral space it must yield. In particular, by fixing the granted space beforehand, the defender cannot adopt blocking maneuvers, such as deviating from the racing line to reduce the available overtaking gap. For the autonomous racing case study considered here, this limitation is acceptable, as abrupt blocking maneuvers are usually prohibited by competition rules. Moreover, in practice, blocking behavior can often be neglected if the planner tuning prioritizes path-tracking accuracy over speed optimization, or, as in our case, when the objective does not include an explicitly competitive term (typical of zero-sum games) that rewards hindering the opponent's progress. Importantly, this simplification only reduces the defender's strategic options but does not compromise regulation compliance, as the required overtaking space is still guaranteed by the pre-computed constraints.

5.6 Experimental Results

This section presents simulation and experimental validation of the proposed RA-GTP framework on the Berkeley Autonomous Race Car (BARC), a 1:10-scale research platform for autonomous racing. We consider two implementations of RA-GTP: *Full RA-GTP*, a more accurate but computationally demanding variant that solves the complete line-search MI-SQP formulation [48] in combination with a standard IBR scheme[1], and *Fast RA-GTP*, a real-time oriented variant that combines the RTI-MISQP with the single-pass IBR, and uses the simplified defender RC-MPC problem described in Section 5.5. The results highlight three main aspects: (i) simulation studies comparing the Full and Fast RA-GTP, (ii) computational timing analysis across solver variants, and (iii) real-world experiments on the BARC platform against a baseline approach.

5.6.1 Simulation Study: Full vs. Fast RA-GTP

We compare closed-loop trajectories obtained from 50 overtaking simulations of Full and Fast RA-GTP under identical initial conditions.

A trajectory pair is classified as *matching* if the maximum lateral deviation between them does not exceed the vehicle width. An example of matching and non-matching outcomes is shown in Fig. 5.1. For matching cases, the root-mean-square (RMS) values of the longitudinal and lateral deviations are reported to quantify trajectory similarity.

To evaluate overtaking performance, we define the time-to-overtake (TTO) as the duration required for the ego vehicle to complete the overtaking maneuver. We then compute the average difference in TTO for both the matching subset and the full set of simulations:

$$\Delta\mathrm{TTO} \triangleq \mathrm{TTO}^{\mathrm{Full}} - \mathrm{TTO}^{\mathrm{Fast}}. \tag{5.8}$$

The results in Table 5.1 show that the deviations in matching trajectories remain small (centimeter-level laterally, decimeter-level longitudinally), and that the average difference in TTO is below two control steps. Although the Full variant tends to produce slightly earlier overtakes in the matching subset, the overall average ΔTTO is positive, indicating that the Fast variant does not degrade overtaking performance, even in the presence of trajectory mismatches.

It is worth noting that non-matching outcomes may arise from small variations in the vehicle state just before the overtake commitment, which can lead to different maneuver realizations, such as selecting the opposite overtaking side or postponing the maneuver itself. This sensitivity is inherent to the presence of logical constraints in the formulation, whose activation depends on discrete conditions that can change abruptly near the decision boundaries. Thus, small variations in the control inputs

[1] The maximum number of iterations is set to 5 in both algorithms.

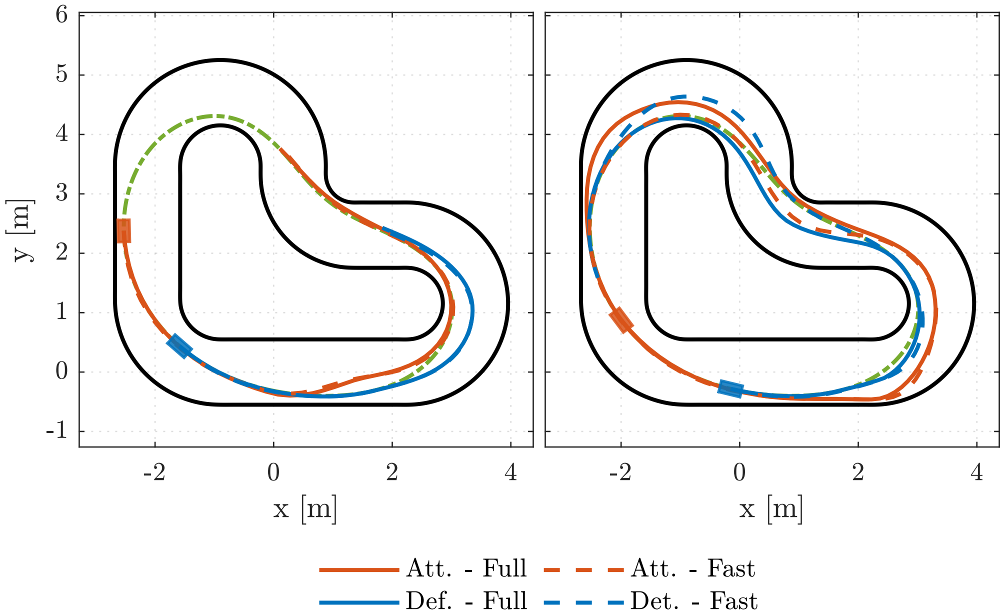


Figure 5.1. Closed-loop trajectories obtained with the Full and Fast RA-GTP variants in a matching case (left) and a non-matching case (right). In the matching case, both methods produce similar overtaking trajectories with comparable longitudinal and lateral evolution. In the non-matching case, small differences in the vehicle state near the overtake commitment lead to opposite maneuver realizations, with the Fast RA-GTP executing a left-side overtake and the Full RA-GTP overtaking on the right.

induced by suboptimality may lead to different outcomes. However, in practice, this sensitivity is comparable to other sources of prediction error, such as noise, modeling inaccuracies, and external disturbances. In this sense, such outcome differences would likely be observed regardless of the approximations introduced by the Fast RA-GTP variant.

Table 5.1. Closed-loop comparison of Full vs. Fast RA-GTP over 50 overtaking simulations with identical initial conditions.

Metric	**Value**	
Matching rate	42 / 50 (84%)	
RMS deviation (s) [cm]	Attacker: 15.7	Defender: 9.2
RMS deviation (n) [cm]	Attacker: 2.6	Defender: 2.1
Average ΔTTO (matching) [s]	-0.15	
Average ΔTTO (all) [s]	0.04	

5.6.2 Computational Performance Analysis

The computational feasibility of four solver configurations is evaluated: Full RA-GTP, RTI-MISQP only, RTI-MISQP with single-pass IBR, and Fast RA-GTP. The implementation is developed in Python, using `CasADi` [80] for automatic differentiation and model linearization, and `Gurobi` [81] for the solution of the MIQP subproblems.

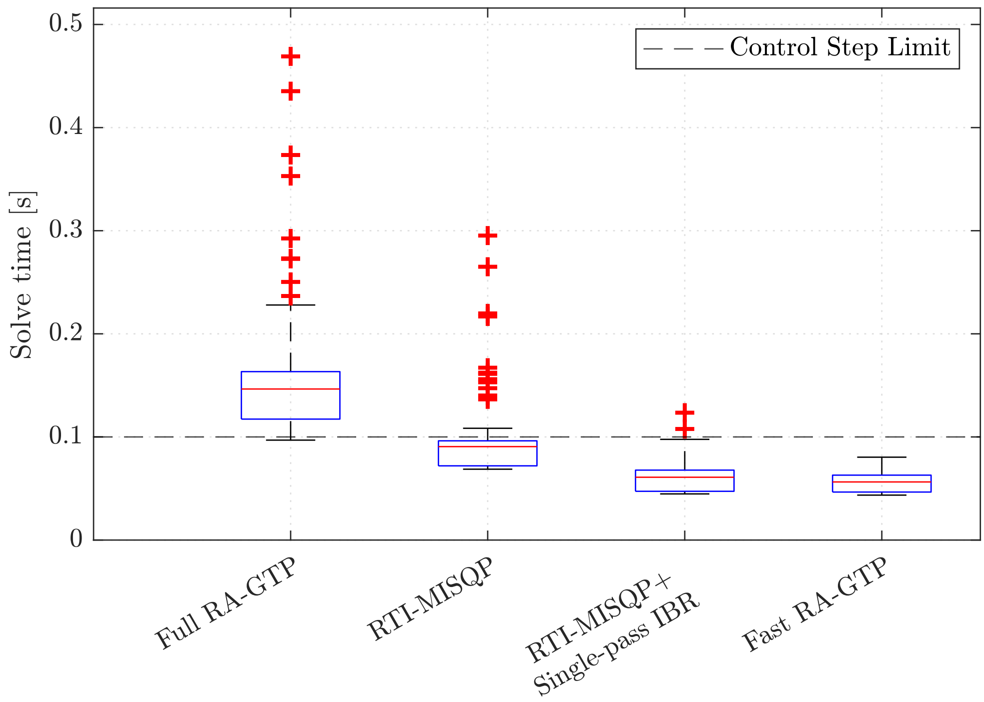


Figure 5.2. Solve time comparison across different RA-GTP implementations during an overtake. Box plots show the median (line), interquartile range (box), and whiskers up to 1.5×IQR, with outliers plotted individually. Outliers typically occurred at the onset of interaction between the two players.

The control sampling time was set to $T_s = 100$ ms. The RC-MPC problems used the kinematic bicycle model expressed in the Frenet frame (Appendix A.2) with a prediction horizon of $N = 10$ steps. Timing data were collected from the simulated overtaking maneuvers shown in Fig. 5.1, executed on a laptop computer equipped with a 12th Gen Intel Core i7-12700H processor (4.0 GHz) and 16 GB RAM. The boxplot in Fig. 5.2 reports the distribution of computation times, demonstrating that only the Fast RA-GTP consistently remains below the control step threshold, thereby confirming its suitability for real-time application.

Despite the intrinsic computational complexity of mixed-integer formulations, the solver times reported in Table 5.2 demonstrate that each MIQP subproblem within the MI-SQP framework can be solved efficiently in real time. A considerable portion of the total computation time arises from the *update phase*, which includes the construction of the MIQP approximation of the MINLP and the update of mode-dependent parameters through the Gurobi interface. Further speed-ups could be achieved by optimizing this step, potentially allowing for more detailed formulations, such as the full defender model, or for an increased number of IBR iterations within the same real-time constraints.

5.6.3 Experimental Validation

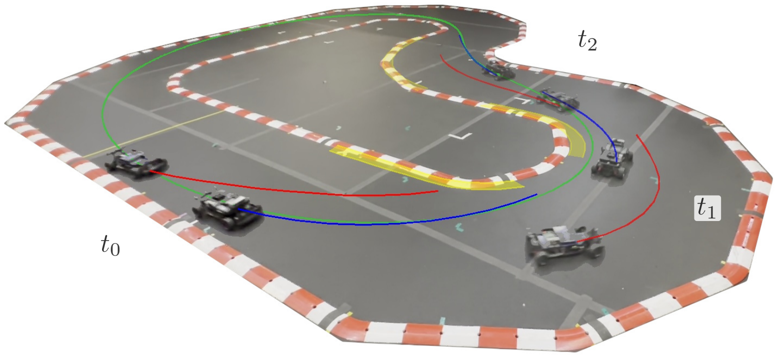
The experiments were conducted on the BARC platform, in a head-to-head racing scenario, on a L-shaped track. State estimation was provided by an external motion

Table 5.2. Computation time statistics for the Regulation-Compliant MPC (RC-MPC). The *Solve* times (average, minimum, and maximum) refer to the MIQP optimization phase, while the *Update* time includes the MIQP construction and parameter-update steps through the Gurobi interface.

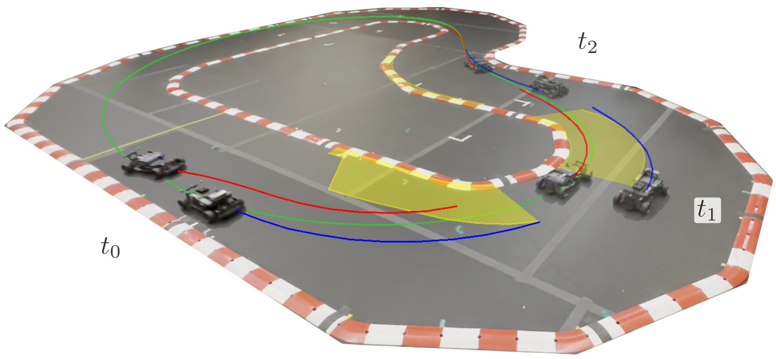
Metric	Defender RC-MPC	Attacker RC-MPC
Average Solve [ms]	4.48	8.92
Maximum Solve [ms]	8.86	25.93
Minimum Solve [ms]	2.15	2.20
Average Update [ms]	19.50	19.29

capture system with millimeter accuracy, while low-level acceleration and steering control were executed by onboard microcontrollers. All optimization problems were computed offboard on an external desktop computer. The attacker and defender motion planners were implemented as two separate ROS 2 nodes, each solving its own Fast RA-GTP planner. At each control cycle, both nodes received as input the measured state of their own vehicle and the opponent's vehicle from the motion capture system, and produced the corresponding control input to be applied on the car. Both players followed the same reference racing line, but with different speed profiles reflecting distinct performance limits: the attacker was assigned higher longitudinal and lateral acceleration bounds, promoting overtaking opportunities against the slower defender. For comparison, a *baseline* variant is considered, where the attacker maintains its collision-avoidance responsibility but models the defender as regulation-agnostic. Fig. 5.3 compares two BARC overtaking experiments. In the baseline (Fig. 5.3a), the attacker neglects the defender's yielding obligation, thus predicting the defender to remain on the racing line and causing repeated aborts, without gaining the right-of-way. In contrast, the Fast RA-GTP (Fig. 5.3b) correctly reason about the defender yielding the granted lateral space (yellow area) once the attacker gains the right-of-way, and successfully completes the maneuver.

In Fig. 5.4, overtaking outcomes are classified based on the longitudinal gap: once the attacker reduces the gap below three car lengths, the maneuver is deemed a *success* if the attacker subsequently passes the defender, and an *abort* if the gap increases again. In the experiments, the baseline planner was never able to complete an overtake and repeatedly aborted the maneuver before reaching the side-by-side phase. In contrast, the Fast RA-GTP successfully executed several inner and outer overtakes, demonstrating its ability to reason strategically within the same real-time constraints. In both cases, no collisions occurred. This is explained by the fact that the baseline attacker, which employs a rule-agnostic defender model, exhibits overly conservative behavior and tends to yield prematurely. In contrast, the RA-GTP attacker initiates more aggressive yet regulation-compliant maneuvers that remain safe as long as both agents adhere to the rules. These results confirm that the RC-MPC solved via RTI-MISQP preserved constraints satisfaction despite the suboptimality introduced by the solver. Further details are provided in the accompanying video [84], which showcases multiple overtaking maneuvers and the evolution of the planner's internal variables, illustrating the attacker's decision-making process.



(a) Baseline.



(b) Fast RA-GTP.

Figure 5.3. Overlay of three video frames from BARC experiments comparing the baseline (a) and Fast RA-GTP (b). Attacker (red), defender (blue), racing line (green), and yielded space (yellow) are shown. The attacker initiates an inner pass, which succeeds under RA-GTP but is aborted by the baseline.

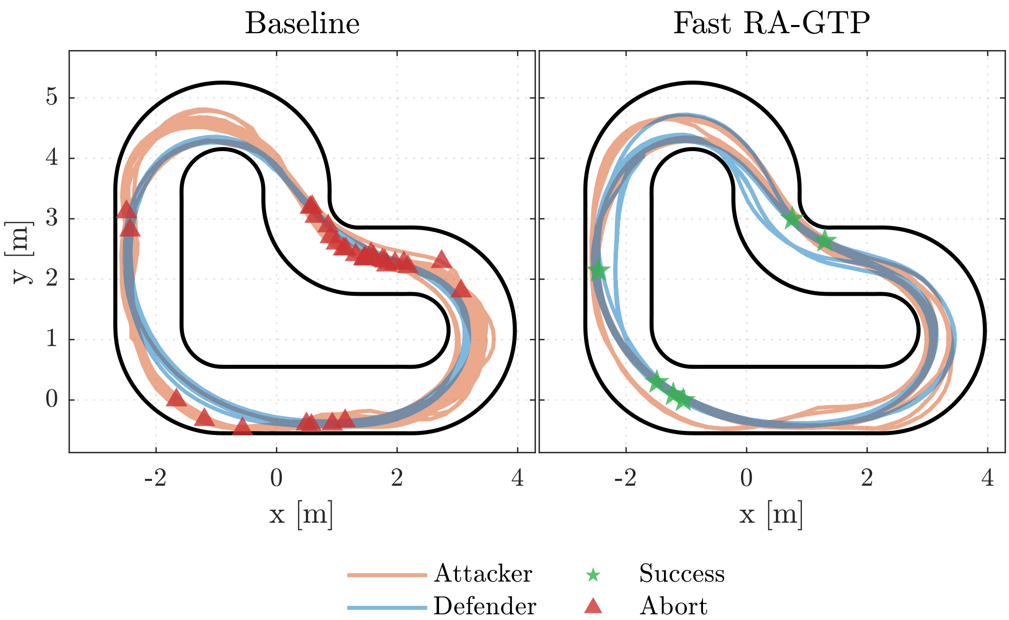


Figure 5.4. Closed-loop trajectories of attacker and defender during BARC experiments across 21 defender laps. The attacker and defender trajectories are shown in red and and blue, respectively. Successful overtakes and aborted attempts are indicated by green and red markers.

5.7 Summary and Discussion

This chapter presented the real-time implementation and experimental validation of the Regulation-Aware Game-Theoretic Planner (RA-GTP) for head-to-head autonomous racing presented in Chapter 4. The framework builds on a regulation-compliant MPC formulation that encodes overtaking rules within an MLD framework. Each player's resulting MINLP is solved efficiently using an RTI-inspired MI-SQP scheme, while the strategic reasoning is achieved by computing an approximate solution of the GNEP through a single-pass IBR update. To further improve efficiency, the defender's RC-MPC was simplified by pre-computing part of the regulation logic, yielding the Fast RA-GTP implementation.

Simulation studies showed that the Fast RA-GTP preserve the overtaking performance of Full RA-GTP, while a timing analysis demonstrated that only the Fast variant consistently remains below the control step threshold. Experiments on the 1:10-scale BARC platform further validated the approach: Fast RA-GTP achieved successful regulation-compliant overtakes in real time, whereas a regulation-agnostic baseline failed due to excessive conservatism. These results highlight the contribution of integrating RTI-MISQP and single-pass IBR as an effective strategy to bring game-theoretic planning with regulation constraints into the real-time domain.

A current limitation of the proposed framework lies in the simplified formulation adopted for the defender's RC-MPC problem, where only the longitudinal motion is optimized to anticipate or delay the interaction, while the right-of-way constraints are fixed a priori at each optimization step. This simplification removes the defender's ability to actively block the attacker's maneuver. Nonetheless, additional computational speed-ups could be obtained by optimizing the update of the problem matrices through the Gurobi interface, which is currently less efficient than the actual solver execution. Improving this step could make it feasible to reintroduce the full defender model, restoring its blocking capability while preserving real-time performance.

Chapter 6

Conclusion

6.1 Summary of Contributions

The research presented in this thesis addressed optimization-based predictive motion planning and control for autonomous racing, with the overarching goal of achieving dynamically feasible, regulation-aware, and computationally efficient behavior in full-scale autonomous vehicles. The three main contributions introduced in Chapter 1 are here revisited and discussed in light of the obtained results.

C1 concerned the design of a hierarchical motion planning and control framework that brings optimization into the planning layer of full-scale autonomous racing. The proposed architecture effectively bridged optimization-based predictive motion planning with real-world autonomous racing, jointly handling vehicle dynamics limits and interactions constraints within a unified planning layer. This design ensured consistency of the generated trajectories with respect to multiple domain constraints, such as dynamic feasibility, actuation limits, and interaction safety, thereby aligning the planner's output with the physical and operational limits of the vehicle. Experimental validation in international autonomous racing competitions confirmed the effectiveness of the approach, achieving high-performance behavior and podium results in both single-vehicle and head-to-head racing formats.

C2 addressed the problem of predictive, optimal trajectory planning under racing regulations. The proposed Regulation-Compliant Model Predictive Control formulation encoded racing regulations through Mixed Logical Dynamical constraints, resulting in a Mixed-Integer MPC problem capable of computing trajectories that are both optimal and compliant with rules. Building on this foundation, the competitive interaction between vehicles was modeled as a racing game that explicitly accounts for asymmetric player responsibilities, yielding a Generalized Nash Equilibrium Problem in which coupling constraints differ among agents. The presented Regulation-Aware Game-Theoretic Planner aimed to approximate the solution of this GNEP through an Iterative Best Response scheme, thereby enabling the ego vehicle to reason about the opponent's regulation-constrained motion. Simulation studies confirmed the high realism of the resulting interactions in head-to-head racing scenarios, where the attacker successfully exploited the defender's yielding obligations to complete overtaking maneuvers, which could not be achieved by a rule-agnostic baseline.

C3 addressed the challenge of achieving real-time implementability of regulation-compliant and regulation-aware planning. To this aim, computational strategies inspired by Real-Time Iteration SQP were extended to a Mixed-Integer SQP solver, enabling fast execution of the RC-MPC formulation. In addition, a single-pass Iterative Best Response scheme was developed to approximate the multi-agent interaction while preserving the essential strategic structure of the game. Experimental validation on scaled autonomous racing platforms demonstrated the real-time feasibility of the proposed Fast RA-GTP implementation and its effectiveness in reproducing competitive, regulation-aware behavior in head-to-head racing scenarios.

Together, these contributions demonstrate that optimization-based predictive planning and control can serve as a unifying paradigm for autonomous driving across complex and diverse scenarios, enabling vehicles to reason about dynamic feasibility, regulatory compliance, and strategic interaction within a single motion planning and control framework.

6.2 Future Work

Building on the results presented in this thesis, several directions emerge for future research aimed at extending the theoretical foundations, robustness, and scalability of optimization-based predictive planning and control for autonomous racing and driving.

A first line of investigation concerns the effect of uncertainty in the opponent model on decision making and safety in head-to-head racing. The current formulation assumes rational agents with known vehicle dynamics, cost functions, and full compliance with racing regulations. These assumptions are reasonable within structured racing environments, where vehicles exhibit comparable dynamic performance and are expected to adhere strictly to the rules. Nevertheless, uncertainties may arise from imperfect perception, modeling errors, or deviations from rational or rule-compliant behavior. Studying how such uncertainties affect the closed-loop behavior of the racing game, and whether their impact differs across scenarios, represents an important direction for future research. For instance, in urban driving contexts, lower speeds and greater maneuverability may allow partial correction of mispredicted interactions, whereas uncertainty in other agents' intent is typically higher and can dominate the decision-making process. Distinguishing whether the resulting behavior under uncertainty corresponds to a merely suboptimal maneuver or an unsafe outcome is essential to evaluate the reliability and robustness of the planner. Incorporating these aspects within a regulation-aware planning framework, through robust, stochastic, or learning-based formulations, would enhance both the realism and safety of the proposed method.

A second line of research concerns the theoretical analysis of the constrained racing game. The existence and characterization of Generalized Nash Equilibria in the presence of nonconvex or mixed-integer constraints remain open problems, and studying these properties would provide stronger guarantees on the validity and stability of the Regulation-Aware Game-Theoretic Planner. Despite the analytical challenges, intuitive reasoning already suggests that a pure-strategy GNE may not exist in certain situations. Consider, for example, a two-player maneuver game in which the attacker can either attempt an overtake or remain behind, and the defender can either yield or block. If blocking is allowed and both agents act rationally, no pair of pure strategies simultaneously satisfies mutual optimality and feasibility: when the attacker commits to the overtake, the defender's best response is to block, making the joint outcome infeasible; conversely, when the attacker remains behind, the defender's best response is to continue, which leaves the attacker's choice suboptimal. This simplified example illustrates how conflicts in responsibility or admissibility can lead to the nonexistence of pure equilibria. In contrast, under mixed strategies, where each player randomizes over maneuvers, the game admits an equilibrium in expectation, reflecting how the attacker's commitment probability depends on the anticipated likelihood of the defender blocking, and vice versa. A formal study of the existence and properties of GNE in the regulation-constrained racing game under mixed strategies would therefore represent an important step toward a theoretical understanding of equilibrium behavior in interactive autonomous driving and competitive racing [42], [85].

A final research direction concerns the extension of the proposed framework to multi-vehicle scenarios. While the current formulation focuses on head-to-head racing, extending it to games involving multiple opponents would enable the study of richer competitive and cooperative behaviors such as drafting, blocking alliances, and multi-car overtakes. From a theoretical standpoint, this corresponds to transitioning from a two-player Generalized Nash Equilibrium Problem to an N-player game with shared and pairwise coupling constraints, where the dimensionality of the joint strategy space and the number of possible maneuver combinations grow combinatorially with N. Such scaling poses both computational and conceptual challenges, as interactions among vehicles can no longer be decomposed into pairwise subgames without loss of optimality. Developing tractable approximations that preserve essential strategic and regulatory features will therefore be crucial. Possible research paths include hierarchical or graph-based formulations that restrict attention to a subset of relevant opponents, learning-based pre-selection of maneuver hypotheses, or distributed solution schemes exploiting the sparsity structure of the interaction graph. These approaches could make multi-vehicle optimization-based planning feasible in real time while maintaining regulation consistency and safety guarantees across all interacting agents.

Appendix A

Vehicle Models

A.1 Vehicle Lateral Dynamics Matrices

The continuous-time lateral dynamics (3.13) are derived from a linearized bicycle model with small slip angles and a first-order steering actuator. The longitudinal velocity v_x is treated as a scheduling parameter. The model parameters are: mass m, yaw inertia I_z, front and rear axle distances l_f, l_r, cornering stiffness coefficients C_f, C_r, and actuator time constant τ_δ. While constant cornering stiffnesses are typically assumed in linear models, here $C_f(v_x)$ and $C_r(v_x)$ are modeled as functions of v_x to capture the influence of aerodynamic downforce on the vertical tire loads, thereby reflecting the resulting variation in effective cornering stiffness (the dependence on v_x is omitted in the following for compactness).

The matrices in (3.13) are given by

$$\mathcal{A}(v_x) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -\dfrac{C_f + C_r}{m v_x} & \dfrac{C_f + C_r}{m} & -\dfrac{l_f C_f - l_r C_r}{m v_x} & \dfrac{C_f}{m} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & \dfrac{-l_f C_f + l_r C_r}{I_z v_x} & \dfrac{l_f C_f - l_r C_r}{I_z} & -\dfrac{l_f^2 C_f + l_r^2 C_r}{I_z v_x} & \dfrac{l_f C_f}{I_z} \\ 0 & 0 & 0 & 0 & -\dfrac{1}{\tau_\delta} \end{bmatrix}, \tag{A.1}$$

$$\mathcal{B}_u = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \dfrac{1}{\tau_\delta} \end{bmatrix}, \tag{A.2}$$

$$\mathcal{B}_d(v_x) = \begin{bmatrix} 0 & 0 \\ -\dfrac{l_f C_f - l_r C_r}{m} - v_x^2 & 1 \\ 0 & 0 \\ \dfrac{-l_f^2 C_f + l_r^2 C_r}{I_z} & 0 \\ 0 & 0 \end{bmatrix}. \tag{A.3}$$

The first disturbance component corresponds to the path curvature κ, while the second component accounts for the lateral acceleration induced by road banking $g \sin\theta$. The steering actuator dynamics are modeled as

$$\dot{\delta} = -\tfrac{1}{\tau_\delta}\delta + \tfrac{1}{\tau_\delta}\delta_{\text{com}}. \tag{A.4}$$

This formulation yields a velocity-dependent (LPV) model suitable for discretization and use in the lateral MPC-based planner.

A.2 Kinematic bicycle model in Frenet coordinates

The kinematic bicycle model, expressed with respect to the curvilinear reference frame aligned to the reference path, is given by

$$\begin{aligned} \dot{s} &= \frac{v\cos\big(e_\psi + \beta\big)}{1 - \kappa(s)\,n}, \\ \dot{n} &= v\sin\big(e_\psi + \beta\big), \\ \dot{e}_\psi &= \frac{v}{L}\cos\beta\,\tan\delta - \kappa(s)\dot{s}, \\ \dot{v} &= a, \\ \dot{\delta} &= \omega, \end{aligned} \tag{A.5}$$

where s is the path arc-length coordinate, n the lateral deviation, e_ψ the heading error, v the vehicle speed, δ the steering angle, and $L = l_f + l_r$ the wheelbase. The control inputs are the longitudinal acceleration a and the steering rate ω. The sideslip angle at the vehicle's center of gravity is approximated kinematically as

$$\beta = \arctan\left(\frac{l_r}{L}\tan\delta\right). \tag{A.6}$$

Bibliography

[1] C. Urmson *et al.*, "Autonomous driving in urban environments: Boss and the Urban Challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008. DOI: 10.1002/rob.20255.

[2] D. Gonzalez, J. Perez, V. Milanes, and F. Nashashibi, "A Review of Motion Planning Techniques for Automated Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1135–1145, 2016. DOI: 10.1109/TITS.2015.2498841.

[3] C. Badue *et al.*, "Self-driving cars: A survey," *Expert Systems with Applications*, vol. 165, p. 113 816, 2021. DOI: 10.1016/j.eswa.2020.113816.

[4] P. Sun *et al.*, "Scalability in Perception for Autonomous Driving: Waymo Open Dataset," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA: IEEE, 2020, pp. 2443–2451. DOI: 10.1109/CVPR42600.2020.00252.

[5] R. Rajamani, *Vehicle Dynamics and Control*. Boston, MA: Springer US, 2012. DOI: 10.1007/978-1-4614-1433-9.

[6] 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. DOI: 10.1109/TCST.2007.894653.

[7] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli, "A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016. DOI: 10.1109/TIV.2016.2578706.

[8] 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. DOI: 10.1146/annurev-control-060117-105157.

[9] J. Betz *et al.*, "Autonomous Vehicles on the Edge: A Survey on Autonomous Vehicle Racing," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 3, pp. 458–488, 2022. DOI: 10.1109/OJITS.2022.3181510.

[10] M. O'Kelly, H. Zheng, D. Karthik, and R. Mangharam, "F1TENTH: An Open-source Evaluation Environment for Continuous Control and Reinforcement Learning," in *Proceedings of the NeurIPS 2019 Competition and Demonstration Track*, PMLR, 2020, pp. 77–89. [Online]. Available: `https://proceedings.mlr.press/v123/o-kelly20a.html`.

[11] "Indy Autonomous Challenge," Indy Autonomous Challenge Official Website. (2025), [Online]. Available: `https://www.indyautonomouschallenge.com/`.

[12] "Abu Dhabi Autonomous Racing League," Abu Dhabi Autonomous Racing League Official Website. (2025), [Online]. Available: `https://a2rl.io/autonomous-car-race`.

[13] C. Katrakazas, M. Quddus, W.-H. Chen, and L. Deka, "Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 416–442, 2015. DOI: `10.1016/j.trc.2015.09.011`.

[14] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd edition. Madison, Wisconsin: Nob Hill Publishing, 2017, 623 pp.

[15] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*, 1st ed. Cambridge University Press, 2017. DOI: `10.1017/9781139061759`.

[16] S. Di Cairano, H. E. Tseng, D. Bernardini, and A. Bemporad, "Vehicle Yaw Stability Control by Coordinated Active Front Steering and Differential Braking in the Tire Sideslip Angles Domain," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1236–1248, 2013. DOI: `10.1109/TCST.2012.2198886`.

[17] X. Zhang, A. Liniger, and F. Borrelli, "Optimization-Based Collision Avoidance," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 3, pp. 972–983, 2021. DOI: `10.1109/TCST.2019.2949540`.

[18] U. Rosolia, S. De Bruyne, and A. G. Alleyne, "Autonomous Vehicle Control: A Nonconvex Approach for Obstacle Avoidance," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 2, pp. 469–484, 2017. DOI: `10.1109/TCST.2016.2569468`.

[19] I. Batkovic, M. Ali, P. Falcone, and M. Zanon, "Safe Trajectory Tracking in Uncertain Environments," *IEEE Transactions on Automatic Control*, pp. 1–14, 2022. DOI: `10.1109/TAC.2022.3207875`.

[20] E. Joa, H. Lee, E. Y. Choi, and F. Borrelli, "Energy-Efficient Lane Changes Planning and Control for Connected Autonomous Vehicles on Urban Roads," in *2023 IEEE Intelligent Vehicles Symposium (IV)*, Anchorage, AK, USA: IEEE, 2023, pp. 1–6. DOI: `10.1109/IV55152.2023.10186574`.

[21] 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. DOI: `10.1109/LCSYS.2020.3006819`.

[22] N. Murgovski and J. Sjoberg, "Predictive cruise control with autonomous overtaking," in *2015 54th IEEE Conference on Decision and Control (CDC)*, Osaka: IEEE, 2015, pp. 644–649. DOI: 10.1109/CDC.2015.7402302.

[23] L. Riegger, M. Carlander, N. Lidander, N. Murgovski, and J. Sjoberg, "Centralized MPC for autonomous intersection crossing," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Rio de Janeiro, Brazil: IEEE, 2016, pp. 1372–1377. DOI: 10.1109/ITSC.2016.7795736.

[24] U. Rosolia and F. Borrelli, "Learning How to Autonomously Race a Car: A Predictive Control Approach," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 6, pp. 2713–2719, 2020. DOI: 10.1109/TCST.2019.2948135.

[25] E. Alcalá, V. Puig, J. Quevedo, and U. Rosolia, "Autonomous racing using Linear Parameter Varying-Model Predictive Control (LPV-MPC)," *Control Engineering Practice*, vol. 95, p. 104 270, 2020. DOI: 10.1016/j.conengprac.2019.104270.

[26] A. Liniger, A. Domahidi, and M. Morari, "Optimization-based autonomous racing of 1:43 scale RC cars," *Optimal Control Applications and Methods*, vol. 36, no. 5, pp. 628–647, 2015. DOI: 10.1002/oca.2123.

[27] N. R. Kapania and J. C. Gerdes, "Design of a feedback-feedforward steering controller for accurate path tracking and stability at the limits of handling," *Vehicle System Dynamics*, vol. 53, no. 12, pp. 1687–1704, 2015. DOI: 10.1080/00423114.2015.1055279.

[28] J. L. Vazquez, M. Bruhlmeier, A. Liniger, A. Rupenyan, and J. Lygeros, "Optimization-Based Hierarchical Motion Planning for Autonomous Racing," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA: IEEE, 2020, pp. 2397–2403. DOI: 10.1109/IROS45743.2020.9341731.

[29] A. Raji *et al.*, "A Tricycle Model to Accurately Control an Autonomous Racecar with Locked Differential," in *2023 IEEE 11th International Conference on Systems and Control (ICSC)*, Sousse, Tunisia: IEEE, 2023, pp. 782–789. DOI: 10.1109/ICSC58660.2023.10449744.

[30] A. Wischnewski, M. Euler, S. Gümüs, and B. Lohmann, "Tube model predictive control for an autonomous race car," *Vehicle System Dynamics*, vol. 60, no. 9, pp. 3151–3173, 2022. DOI: 10.1080/00423114.2021.1943461.

[31] A. Raji *et al.*, "Motion Planning and Control for Multi Vehicle Autonomous Racing at High Speeds," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, Macau, China: IEEE, 2022, pp. 2775–2782. DOI: 10.1109/ITSC55140.2022.9922239.

[32] C. C. Chung *et al.*, "Autonomous System for Head-to-Head Race: Design, Implementation and Analysis; Team KAIST at the Indy Autonomous Challenge," *IEEE Transactions on Field Robotics*, pp. 1–1, 2025. DOI: 10.1109/TFR.2024.3497922.

[33] T. Stahl, A. Wischnewski, J. Betz, and M. Lienkamp, "Multilayer Graph-Based Trajectory Planning for Race Vehicles in Dynamic Scenarios," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, Auckland, New Zealand: IEEE, 2019, pp. 3149–3154. DOI: `10.1109/ITSC.2019.8917032`.

[34] L. Ögretmen*, M. Rowold*, T. Betz, A. Langmann, and B. Lohmann, "A Hybrid Trajectory Planning Approach for Autonomous Rule-Compliant Multi-Vehicle Oval Racing," *SAE International Journal of Connected and Automated Vehicles*, vol. 7, no. 1, pp. 12-07-01–0007, 2023. DOI: `10.4271/12-07-01-0007`.

[35] C. Jung *et al.*, "An Autonomous Racing System: Design, Implementation, and Analysis; Team KAIST at the IAC," *Field Robotics*, vol. 3, no. 1, pp. 766–800, 2023. DOI: `10.55417/fr.2023024`.

[36] A. Raji *et al.*, "Er.autopilot 1.1: A Software Stack for Autonomous Racing on Oval and Road Course Tracks," *IEEE Transactions on Field Robotics*, vol. 1, pp. 332–359, 2024. DOI: `10.1109/TFR.2024.3501252`.

[37] J. Betz *et al.*, "TUM autonomous motorsport: An autonomous racing software for the Indy Autonomous Challenge," *Journal of Field Robotics*, vol. 40, no. 4, pp. 783–809, 2023. DOI: `10.1002/rob.22153`.

[38] T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory, 2nd Edition*. Society for Industrial and Applied Mathematics, 1998. DOI: `10.1137/1.9781611971132`.

[39] A. Zanardi, S. Bolognani, A. Censi, and E. Frazzoli, "Game Theoretical Motion Planning,"

[40] C. Burger, J. Fischer, F. Bieder, O. S. Tas, and C. Stiller, "Interaction-Aware Game-Theoretic Motion Planning for Automated Vehicles using Bi-level Optimization," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, Macau, China: IEEE, 2022, pp. 3978–3985. DOI: `10.1109/ITSC55140.2022.9922600`.

[41] F. Facchinei and J.-S. Pang, Eds., *Finite-Dimensional Variational Inequalities and Complementarity Problems*. New York, NY: Springer-Verlag New York, Inc, 2004. DOI: `10.1007/b97543`.

[42] F. Facchinei and C. Kanzow, "Generalized Nash Equilibrium Problems," *Annals of Operations Research*, vol. 175, no. 1, pp. 177–211, 2010. DOI: `10.1007/s10479-009-0653-x`.

[43] S. Le Cleac'h, M. Schwager, and Z. Manchester, "ALGAMES: A fast augmented Lagrangian solver for constrained dynamic games," *Autonomous Robots*, vol. 46, no. 1, pp. 201–215, 2022. DOI: `10.1007/s10514-021-10024-7`.

[44] A. Zanardi, E. Mion, M. Bruschetta, S. Bolognani, A. Censi, and E. Frazzoli, "Urban Driving Games With Lexicographic Preferences and Socially Efficient Nash Equilibria," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4978–4985, 2021. DOI: `10.1109/LRA.2021.3068657`.

[45] L. Ögretmen, M. Chen, P. Pitschi, and B. Lohmann. "Trajectory Planning Using Reinforcement Learning for Interactive Overtaking Maneuvers in Autonomous Racing Scenarios." arXiv: `2404.10658 [cs]`. (2024), pre-published.

[46] V. Raman, M. Maasoumy, and A. Donzé, "Model predictive control from signal temporal logic specifications: A case study," in *Proceedings of the 4th ACM SIGBED International Workshop on Design, Modeling, and Evaluation of Cyber-Physical Systems*, Berlin Germany: ACM, 2014, pp. 52–55. DOI: 10.1145/2593458.2593472.

[47] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999. DOI: 10.1016/S0005-1098(98)00178-2.

[48] R. Quirynen and S. D. Cairano, "Sequential Quadratic Programming Algorithm for Real-Time Mixed-Integer Nonlinear MPC," in *2021 60th IEEE Conference on Decision and Control (CDC)*, Austin, TX, USA: IEEE, 2021, pp. 993–999. DOI: 10.1109/CDC45484.2021.9683714.

[49] A. Ghezzi, W. V. Roy, S. Sager, and M. Diehl. "A Sequential Benders-based Mixed-Integer Quadratic Programming Algorithm." arXiv: 2404.11786 [math]. (2024), pre-published.

[50] R. D. McAllister and J. B. Rawlings, "Advances in Mixed-Integer Model Predictive Control," in *2022 American Control Conference (ACC)*, Atlanta, GA, USA: IEEE, 2022, pp. 364–369. DOI: 10.23919/ACC53348.2022.9867869.

[51] G. Williams, B. Goldfain, P. Drews, J. M. Rehg, and E. A. Theodorou, "Best Response Model Predictive Control for Agile Interactions Between Autonomous Ground Vehicles," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD: IEEE, 2018, pp. 2403–2410. DOI: 10.1109/ICRA.2018.8462831.

[52] R. Spica, E. Cristofalo, Z. Wang, E. Montijano, and M. Schwager, "A Real-Time Game Theoretic Planner for Autonomous Two-Player Drone Racing," *IEEE Transactions on Robotics*, vol. 36, no. 5, pp. 1389–1403, 2020. DOI: 10.1109/TRO.2020.2994881.

[53] Z. Wang, R. Spica, and M. Schwager, "Game Theoretic Motion Planning for Multi-robot Racing," in *Distributed Autonomous Robotic Systems*, N. Correll, M. Schwager, and M. Otte, Eds., vol. 9, Cham: Springer International Publishing, 2019, pp. 225–238. DOI: 10.1007/978-3-030-05816-6_16.

[54] Z. Wang, T. Taubner, and M. Schwager, "Multi-agent sensitivity enhanced iterative best response: A real-time game theoretic planner for drone racing in 3D environments," *Robotics and Autonomous Systems*, vol. 125, p. 103 410, 2020. DOI: 10.1016/j.robot.2019.103410.

[55] I. Kolmanovsky, E. Garone, and S. Di Cairano, "Reference and command governors: A tutorial on their theory and automotive applications," in *2014 American Control Conference*, Portland, OR, USA: IEEE, 2014, pp. 226–241. DOI: 10.1109/ACC.2014.6859176.

[56] E. Garone, S. Di Cairano, and I. Kolmanovsky, "Reference and command governors for systems with constraints: A survey on theory and applications," *Automatica*, vol. 75, pp. 306–328, 2017. DOI: 10.1016/j.automatica.2016.08.013.

[57] T. Novi, A. Liniger, R. Capitani, and C. Annicchiarico, "Real-time control for at-limit handling driving on a predefined path," *Vehicle System Dynamics*, vol. 58, no. 7, pp. 1007–1036, 2020. DOI: 10.1080/00423114.2019.1605081.

[58] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a Frenét Frame," in *2010 IEEE International Conference on Robotics and Automation*, Anchorage, AK: IEEE, 2010, pp. 987–993. DOI: 10.1109/ROBOT.2010.5509799.

[59] A. Wischnewski, T. Herrmann, F. Werner, and B. Lohmann, "A Tube-MPC Approach to Autonomous Multi-Vehicle Racing on High-Speed Ovals," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 368–378, 2023. DOI: 10.1109/TIV.2022.3169986.

[60] M. Rowold, L. Ögretmen, T. Kerbl, and B. Lohmann, "Efficient Spatiotemporal Graph Search for Local Trajectory Planning on Oval Race Tracks," *Actuators*, vol. 11, no. 11, p. 319, 2022. DOI: 10.3390/act11110319.

[61] M. McNaughton, C. Urmson, J. M. Dolan, and J.-W. Lee, "Motion planning for autonomous driving with a conformal spatiotemporal lattice," in *2011 IEEE International Conference on Robotics and Automation*, Shanghai, China: IEEE, 2011, pp. 4889–4895. DOI: 10.1109/ICRA.2011.5980223.

[62] Y. Song, H. Lin, E. Kaufmann, P. Durr, and D. Scaramuzza, "Autonomous Overtaking in Gran Turismo Sport Using Curriculum Reinforcement Learning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China: IEEE, 2021, pp. 9403–9409. DOI: 10.1109/ICRA48506.2021.9561049.

[63] A. Liniger and J. Lygeros, "A Noncooperative Game Approach to Autonomous Racing," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 3, pp. 884–897, 2020. DOI: 10.1109/TCST.2019.2895282.

[64] M. Wang, Z. Wang, J. Talbot, J. C. Gerdes, and M. Schwager, "Game-Theoretic Planning for Self-Driving Cars in Multivehicle Competitive Scenarios," *IEEE Transactions on Robotics*, vol. 37, no. 4, pp. 1313–1325, 2021. DOI: 10.1109/TRO.2020.3047521.

[65] M. Guiggiani, *The Science of Vehicle Dynamics: Handling, Braking, and Ride of Road and Race Cars*. Cham: Springer International Publishing, 2018. DOI: 10.1007/978-3-319-73220-6.

[66] K. J. Åström and T. Hägglund, *PID Controllers: Theory, Design, and Tuning*, 2. ed. Research Triangle Park, NC: Instrument Society of America, 1995, 343 pp.

[67] H. B. Pacejka, *Tyre and Vehicle Dynamics*, 2nd ed. Amsterdam: Elsevier/BH, 2006, 642 pp.

[68] G. Frison and M. Diehl, "HPIPM: A high-performance quadratic programming framework for model predictive control," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6563–6569, 2020. DOI: 10.1016/j.ifacol.2020.12.073.

[69] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, "OSQP: An operator splitting solver for quadratic programs," *Mathematical Programming Computation*, vol. 12, no. 4, pp. 637–672, 2020. DOI: 10.1007/s12532-020-00179-2.

[70] E. C. Kerrigan and J. M. Maciejowski, "SOFT CONSTRAINTS AND EXACT PENALTY FUNCTIONS IN MODEL PREDICTIVE CONTROL,"

[71] E. Kerrigan and J. Maciejowski, "Designing model predictive controllers with prioritised constraints and objectives," in *Proceedings. IEEE International Symposium on Computer Aided Control System Design*, Glasgow, UK: IEEE, 2002, pp. 33–38. DOI: 10.1109/CACSD.2002.1036925.

[72] Y. S. Quan, M. Jeddi, F. Prignoli, and P. Falcone, "Priority-Driven Constraints Softening in Safe MPC for Perturbed Systems," *IEEE Control Systems Letters*, vol. 9, pp. 1069–1074, 2025. DOI: 10.1109/LCSYS.2025.3580494.

[73] A. Toschi, F. Prignoli, and M. Bertogna, "Modular Decision-Making and Drivable Areas for Multi-Agent Autonomous Racing," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hangzhou, China: IEEE, 2025, pp. 12 435–12 441. DOI: 10.1109/IROS60139.2025.11246897.

[74] Abu Dhabi Autonomous Racing League. "A2RL 2024 – Yas Marina Circuit." (2024), [Online]. Available: https://www.youtube.com/watch?v=TPzBH-7ckO0.

[75] Indy Autonomous Challenge. "IAC 2024 – Indianapolis Motor Speedway." (2024), [Online]. Available: https://www.youtube.com/watch?v=PclyOWRlrSo.

[76] R. Verschueren *et al.*, "Acados—a modular open-source framework for fast embedded optimal control," *Mathematical Programming Computation*, vol. 14, no. 1, pp. 147–183, 2022. DOI: 10.1007/s12532-021-00208-8.

[77] F. Prignoli. "UNIMORE Racing – Overtaking Maneuver (IAC 2024 Final, Onboard + Telemetry View)." (2024), [Online]. Available: https://youtu.be/_egseE5LC6E.

[78] F. Facchinei, V. Piccialli, and M. Sciandrone, "Decomposition algorithms for generalized potential games," *Computational Optimization and Applications*, vol. 50, no. 2, pp. 237–262, 2011. DOI: 10.1007/s10589-010-9331-9.

[79] M. Pustilnik, A. Loquercio, and F. Borrelli. "Non-Normalized Solutions of Generalized Nash Equilibrium in Autonomous Racing." arXiv: 2503.12002 [cs]. (2025), pre-published.

[80] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019. DOI: 10.1007/s12532-018-0139-4.

[81] Gurobi Optimization, LLC, *Gurobi Optimizer Reference Manual*, 2024. [Online]. Available: https://www.gurobi.com.

[82] F. Prignoli. "Regulation-Aware Game-Theoretic Motion Planning for Autonomous Racing." (2025), [Online]. Available: https://youtu.be/7Z6xG69eFhI.

[83] S. Gros, M. Zanon, R. Quirynen, A. Bemporad, and M. Diehl, "From linear to nonlinear MPC: Bridging the gap via the real-time iteration," *International Journal of Control*, vol. 93, no. 1, pp. 62–80, 2020. DOI: 10.1080/00207179.2016.1222553.

[84] F. Prignoli, *Regulation-Aware Game-Theoretic Planner: Overtaking Demonstration*, 2025. [Online]. Available: https://youtu.be/XYtuzYaZ5ks.

[85] J. B. Rosen, "Existence and Uniqueness of Equilibrium Points for Concave N-Person Games," *Econometrica*, vol. 33, no. 3, p. 520, 1965. DOI: 10.2307/1911749. JSTOR: 1911749.