arXiv:2608.10618v1 [cs.RO] 11 Aug 2026

Toward the Cognitive–Physical Limits of Embodied Intelligence through a World-Model-Centric Autonomous Racing Agent

Zitong Shan[1†], Baichuan Lou[1†], Yanxin Zhou[1†], Shuge Wu[1], Xianqi He[1], Bolin Zhao[1], Sheng Zhao[1], Zhouheng Li[1], Chee Kiong Ong[2], King Ho Holden Li[1], Chen Lv[1*]

[1]School of Mechanical and Aerospace Engineering, Nanyang Technological University, 639798, Singapore.
[2]K2 Holding, L.L.C, Abu Dhabi, United Arab Emirates.

*Corresponding author(s). E-mail(s): lyuchen@ntu.edu.sg;
†These authors contributed equally to this work.

Abstract

Embodied artificial intelligence aims to develop agents that perceive, reason, and act through continuous interaction with the physical world. Despite rapid progress, most embodied systems are still evaluated within conservative safety margins or moderately complex interaction regimes, leaving their capability boundaries under extreme conditions insufficiently understood. For safe real-world deployment, it is essential to understand where the cognitive and physical limits of embodied agents lie, and how these limits interact. Autonomous racing provides a stringent testbed for this question by combining high-frequency localization and perception, adversarial interaction prediction and decision-making, near-saturated vehicle dynamics, and strict safety constraints. Existing autonomous racing systems have advanced high-speed performance, but rarely provide an explicit mechanism for jointly modeling and refining these coupled limits. Here we show that a world-model-centric autonomous racing agent provides a concrete step toward exploring the cognitive-physical limits of embodied intelligence. The framework learns predictive world models from near-limit successes and failures, capturing interaction evolution, ego dynamics, and feasible-motion boundaries. This couples world-state construction, future-aware reasoning, and near-limit control within a closed-loop refinement process. High quality training data were collected from real-vehicle autonomous racing, where the onboard system maintains robust localization and perception at speeds approaching 256.3

km/h with a peak lateral acceleration of 26.8 m/s^2. In full-scale autonomous racing setting, the well trained world-model-centric agent achieves an 88.3% interaction success rate across various challenging simulated racing scenarios. Through closed-loop continual refinement of the world model and policy, the agent improves its utilization of cognitive-physical limits while recovering from failure modes and generalizing across varying conditions and unseen circuits. These results suggest a boundary-aware methodology in which the world model helps embodied agents represent, predict, and continually refine their capability boundaries for safer deployment.

Keywords: Embodied intelligence, autonomous racing agent, world model, extreme conditions, ability limit

Introduction

Embodied artificial intelligence (Embodied AI) has emerged as a central paradigm in robotics and artificial intelligence by emphasizing intelligence that arises through continuous interaction with the physical world [1, 2]. Unlike conventional AI systems that operate primarily in abstract or virtual domains, embodied agents must perceive, decide, and act under noisy sensing, nonlinear dynamics, and irreversible physical consequences [3]. Although substantial progress has been made across autonomous driving, aerial robotics, legged locomotion, and manipulation [4], most embodied systems are still developed and evaluated within conservative safety margins, moderately complex interactions, or well-modeled physical conditions [5]. As a result, their validity and capability boundaries under extreme conditions remain insufficiently understood [6].

For safe real-world deployment, it is not sufficient to study only how embodied intelligence is generated, learned, or improved. It is equally important to understand where its limits lie, how these limits emerge from cognition and physics, and how they jointly constrain agent capability under demanding conditions. This challenge becomes most evident in extreme tasks, where agents operate persistently near the boundaries of feasible state and action spaces [7, 8]. In such settings, small errors in perception, prediction, reasoning, or control can rapidly escalate into instability, failure, or collision [9, 10].

In this work, the cognitive limit refers to the boundary of an agent's ability to construct, update, predict, and use task-relevant world understanding under severe time pressure, uncertainty, and dynamic interaction. For autonomous racing, this includes high-frequency localization and perception under high-speed motion, prediction of ego–opponent interaction evolution, and decision-making for safe, feasible, and competitive maneuvers. The physical limit refers to the boundary of feasible motion imposed by vehicle dynamics, actuator saturation, tire–road interaction, aerodynamic loading, friction, tire temperature, and stability constraints. These limits are coupled rather than independent: uncertainty in localization, perception, and interaction prediction constrains how safely the physical envelope can be exploited, while the evolving physical state determines which future actions remain predictable, feasible, and safe.

Autonomous racing provides a uniquely suitable testbed for investigating these coupled limits. While agile drones and legged robots have demonstrated remarkable near-limit physical performance [10–15], and urban driving or long-horizon robotic manipulation emphasize complex reasoning under uncertainty [16, 17], relatively few embodied systems require sustained cognitive complexity and persistent operation near physical feasibility boundaries. Racing naturally combines high-speed motion, near-saturated tire–road dynamics, strict collision constraints, and adversarial multi-agent interaction among competing vehicles. It therefore requires an agent to operate near the edge of physical feasibility while simultaneously performing high-frequency localization and perception, interaction prediction, and rapid decisions such as overtaking, defending, and interaction-aware planning under uncertainty [18]. Autonomous racing is thus not merely a high-performance control problem, but a platform for examining how cognitive reasoning and physical execution co-evolve under extreme conditions [19, 20].

Autonomous racing is distinguished by a mature and cross-scale research community [7, 21]. Scaled platforms offer low-cost, reproducible, and extensible testbeds for autonomous racing research [21, 22]. These platforms are supported by open-source vehicle models, planning and control methods, and experimental tool chains [23–26]. Recent work has further demonstrated the growing capability of autonomous racing systems, including trajectory-conditioned zero-shot racing and on-board reasoning with deployed large language models [22, 27]. At the full-vehicle scale, studies on real-track racing have accumulated valuable experience in high-speed trajectory generation, near-limit maneuvering, and safety-constrained control [28–30], providing a concrete basis for studying perception, planning, and control near physical feasibility boundaries [7]. The research ecosystem has given rise to a broad range of autonomous racing methods.

Existing autonomous racing approaches can be broadly grouped into optimization-based pipelines, learning-based policies, and hybrid frameworks [31–33]. Optimization-based methods explicitly encode vehicle dynamics and feasibility constraints, allowing structured and interpretable operation near friction limits [34–37]; however, they remain difficult to extend to rapidly evolving multi-agent interactions involving opponent intent, behavioral uncertainty, and long-horizon coupling [26, 38]. Learning-based methods, inspired by advances in game playing, simulated racing, and drone racing [8, 39–44], can capture complex observation–action mappings for competitive driving [45–47], but often depend strongly on the training vehicle, track geometry, and interaction distribution [48, 49], with limited interpretability and reliability outside the original operating regime [50–52]. Hybrid methods improve robustness by combining model-based and data-driven components [53], for example by learning cost terms, vehicle dynamics, or residual corrections [54–56]. Yet most efforts still optimize task performance within a given sensing–planning–control pipeline, rather than explicitly modeling how predictive reasoning, interaction understanding, and physical feasibility jointly influence the agent's behavior. As a result, autonomous racing lacks a systematic framework for studying embodied intelligence under coupled cognitive–physical limits.

This gap motivates the need for a unified predictive substrate that can connect cognitive reasoning with physical execution. From a system-level perspective, traditional autonomous racing methods often rely on instantaneous state estimation, reactive feature representations, or task-specific predictive modules rather than an integrated model of the world [1, 7, 57]. For near-limit autonomous racing, such a model must predict ego dynamics near feasibility boundaries, capture structured interaction with opponents, represent uncertainty under tight safety margins, and refine decision-making and control capability over time. Recent advances in world models provide a promising foundation for this role [58–60]. By enabling structured simulation of future dynamics and interactions [57, 60], imagination-based reasoning, and counterfactual evaluation, world models offer a natural mechanism for linking predictive world understanding with decision-making and control [41, 61, 62].

Fig. 1 Autonomous racing under cognitive–physical limits. Autonomous racing requires the agent to construct predictive world understanding, reason about the current and future positions of surrounding opponents, generate an ego trajectory that remains feasible under physical limits, and maneuver near the dynamics boundary.

To this end, we take a step toward the cognitive-physical limits of embodied intelligence through a world-model-centric autonomous racing agent. The proposed

framework places a predictive world model at the center of the agent, where it jointly models external interaction evolution, ego dynamics, and feasible motion boundaries. This supports high-frequency world-state construction, future-aware interaction reasoning, risk-sensitive decision-making, and limit-aware control across track geometries and operating conditions. Through continual refinement of both the world model and the policy, the agent improves decision-making and control capability for autonomous racing under joint cognitive-physical limits, as shown in Fig. 1. In this way, autonomous racing becomes not only an application domain, but also a concrete proving ground for studying how embodied agents can approach, exploit, and regulate coupled cognitive-physical limits.

Results

World-Model-Centric Racing Agent Architecture

Extreme autonomous racing makes the cognitive and physical limits of an embodied agent jointly observable. Cognitively, the agent must maintain robust localization and perception under high-speed and high-frequency operation, predict ego–opponent interaction evolution, and make safe and competitive decisions under severe time pressure. Physically, it must exploit the feasible dynamics envelope of the vehicle while respecting actuator constraints, tire–road interaction, and stability margins. The following results therefore evaluate not only racing performance, but also how the agent approaches, exploits, and regulates coupled cognitive-physical limits.

Rather than treating localization, prediction, planning, and control as isolated modules, the proposed agent is organized around a shared world model that connects cognitive reasoning with physical control. As illustrated in Fig. 2, the architecture consists of three coupled components: spatial intelligence, future-aware reasoning, and limit-aware control. Spatial intelligence provides the observable basis of the agent's world-state representation through multi-modal sensor fusion for high-frequency localization and opponent perception. Future-aware reasoning uses world-model rollouts to evaluate possible interaction and dynamics futures. Limit-aware control translates the selected plan into executable commands while respecting the feasible motion boundary. Together, these components allow the agent to reason and act under simultaneous cognitive and physical constraints.

The world model serves as the central predictive substrate of the agent. It jointly models external interaction evolution, ego dynamics, and physical limits that constrain feasible motion. Candidate decisions are evaluated against predicted interaction outcomes and future physical feasibility, while physical-limit information feeds back into planning and control. At runtime, this forms a perception–prediction–decision–control flow: spatial intelligence constructs the current racing state; the world model predicts multi-agent interaction and ego-dynamics futures; future-aware reasoning selects a safe, feasible, and competitive trajectory; and limit-aware control executes it near the physical boundary.

Beyond online execution, the architecture supports continual refinement of both the world model and the policy. Real racing data update the model of ego dynamics, interaction evolution, and physical feasibility; the refined world model then provides

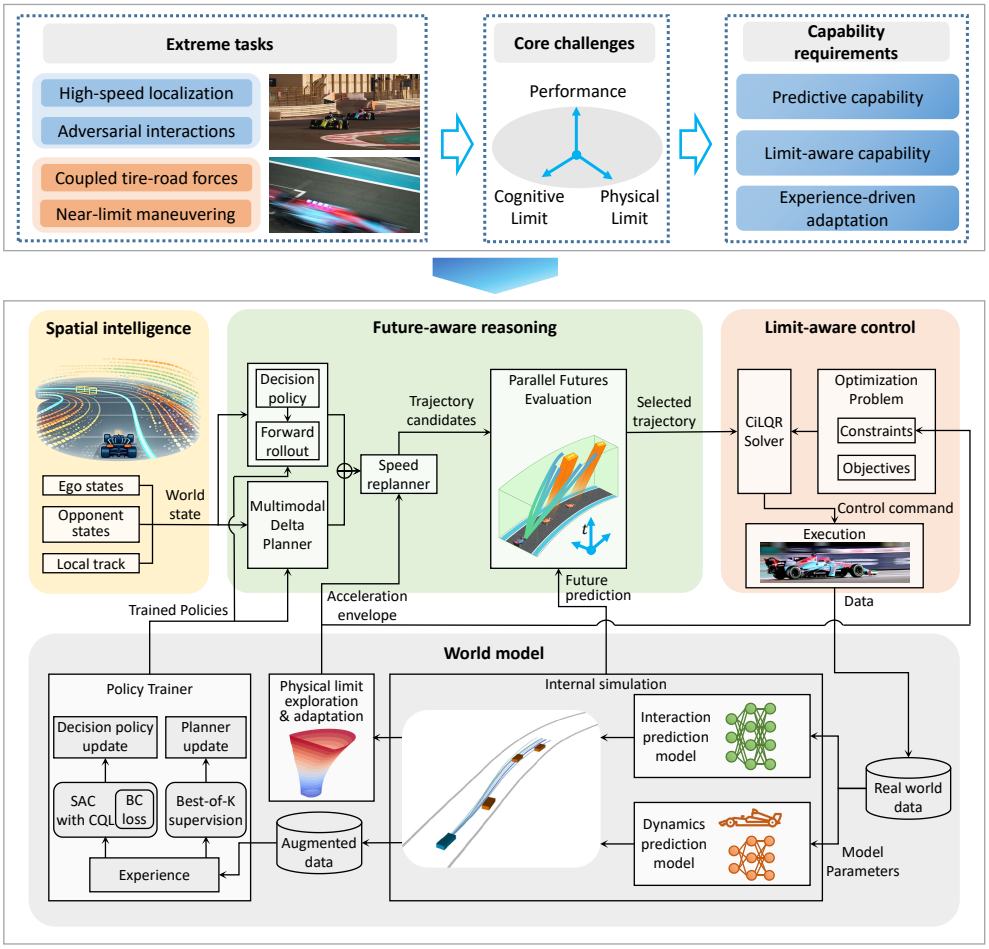


Fig. 2 World-model-centric architecture for autonomous racing under coupled cognitive-physical limits. Extreme racing tasks impose coupled cognitive and physical limits that require predictive capability, limit-aware control, and continual refinement beyond conventional paradigms. The proposed architecture is centered on a shared world model that connects spatial intelligence, future-aware reasoning, and limit-aware control. Real-vehicle racing data were collected for model training, providing physical grounding and interaction priors for world-model refinement. The the refined world model provides future predictions and feasible-motion priors for subsequent policy improvement.

structured predictions for improving future-aware reasoning and limit-aware control. This closed-loop refinement allows the agent to improve both cognitive capability, including interaction reasoning and decision-making, and physical capability, including dynamic-envelope utilization and stability near the limit.

Overall, the world-model-centric architecture provides a concrete mechanism for studying coupled cognitive-physical limits in autonomous racing. The subsequent results examine this mechanism through high-frequency localization and perception,

predictive world modeling, adversarial interaction reasoning, limit-aware control, and progressive agent evolution across conditions.

Predictive World Model under Cognitive-Physical Limits

Operating near the cognitive-physical limits requires the agent to both reconstruct the current racing state and predict its future evolution under near-limit conditions. This section evaluates these two connected layers: high-frequency localization and perception for current-state construction, and world modeling for predicting ego dynamics, physical feasibility, and interaction evolution. Together, they transform raw sensing into predictive world understanding for downstream reasoning and control.

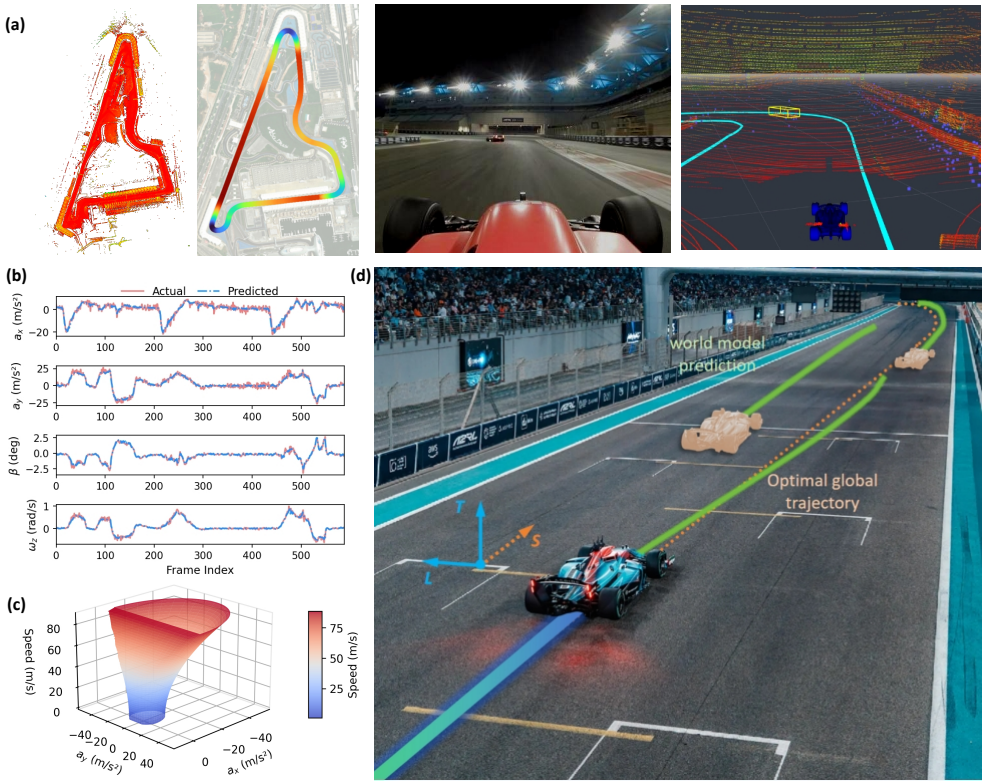


Fig. 3 Predictive world understanding from high-frequency localization and world modeling. High-frequency localization and opponent perception provide the current racing state, while the world model predicts ego dynamics, physical feasibility, and interaction evolution under near-limit racing conditions. (a) Localization and opponent perception in full-scale autonomous racing. (b) Ego dynamics prediction under aggressive maneuvering. (c) Physical motion boundary representation through feasible acceleration envelope evolution. (d) Interaction prediction under extreme conditions.

High-frequency localization and perception provide the observable foundation of the current racing state. The localization and ego-state estimation module fuses

LiDAR, RTK, IMU, and wheel-speed measurements to maintain a stable ego pose under aggressive maneuvering. As shown in Fig. 3(a), the point-cloud map and estimated motion trajectory remain consistently aligned during full-scale racing. At peak speeds approaching 256.3 km/h, the fused-state residuals remain bounded throughout the lap: the lateral position residual is generally within 0.3 m, the longitudinal position residual remains within about 2.0 m, and the longitudinal and lateral velocity residuals remain within about 1.8 m/s, with medians concentrated near 0.1 m/s. The yaw residual is bounded within about 0.06 rad relative to dual-GPS heading and within about 0.01 rad relative to LiDAR odometry. Opponent perception and tracking further extend the current racing state from ego localization to the interactive scene. Surrounding vehicles are detected and tracked under high-speed and close-proximity conditions by fusing LiDAR-based geometric measurements with radar-based velocity cues into unified object states through probabilistic state estimation. As shown in Fig. 3(a), the perceived opponent vehicles remain temporally coherent over full-lap runs, providing the opponent-state foundation required for interaction prediction and future-aware reasoning.

The world model then extends the current racing state into predictive understanding. It contains three complementary components: ego dynamics prediction, physical-limit representation, and interaction prediction. The ego dynamics model predicts future vehicle motion under varying control inputs using a differentiable dynamics backbone with a residual neural network model. As shown in Fig. 3(b), the model captures key near-limit motion states with RMSE values of 0.197 m/s for longitudinal velocity, 0.099 m/s for lateral velocity, 0.042 rad/s for yaw rate, 1.514 m/s^2 for longitudinal acceleration, and 1.983 m/s^2 for lateral acceleration. These prediction errors quantify the ability of the world model to represent the ego vehicle's dynamics response under aggressive control inputs, providing the basis for evaluating future candidate maneuvers near the physical boundary.

The physical-limit representation quantifies the feasible motion boundary of the ego vehicle. As shown in Fig. 3(c), the feasible acceleration envelope varies with vehicle speed: achievable lateral and braking accelerations expand with increasing aerodynamic effects, while longitudinal acceleration capability decreases due to powertrain limitations. By integrating actuator capability, tire–road constraints, and speed-dependent vehicle dynamics into a coupled acceleration envelope, the world model provides a quantitative representation of physical feasibility. This envelope acts as a dynamic constraint for planning and control, allowing candidate actions to be evaluated not only by racing utility, but also by whether they remain within the current physical capability of the vehicle.

The interaction prediction component models the future evolution of multi-agent racing scenarios. An attention-based predictive interaction model encodes the relative spatial relations among interacting agents. By predicting opponent intentions and recursively rolling out the resulting state transitions, the interaction model anticipates interaction evolution beyond immediate observations. As illustrated in Fig. 3(d), the predictions are generated in a structured track-coordinate system, preserving the spatio-temporal consistency of the interactive scene. Over a 3 s horizon, the model achieves a mean ADE of 1.58 m and a median ADE of 0.68 m, together with a

mean FDE of 2.55 m and a median FDE of 0.78 m. This predictive capability allows downstream reasoning to evaluate candidate maneuvers against anticipated opponent motion rather than only the current observed state.

This two-layer characterization links the cognitive foundation of the agent to downstream decision-making and control. High-frequency localization and perception establish a reliable current racing state, while the world model predicts future ego dynamics, physical feasibility, and interaction evolution. The resulting predictive world understanding enables candidate maneuvers to be evaluated against both interaction uncertainty and physical feasibility, supporting future-aware decision-making and limit-aware control.

Future-Aware Reasoning for Adversarial Interaction

Adversarial interaction in autonomous racing exposes the cognitive-physical coupling of the agent. A competitive maneuver is not safe or effective simply because it appears strategically advantageous at the current instant; it must also remain feasible under the future evolution of opponent motion and near-limit ego dynamics. Therefore, future-aware reasoning is formulated as a predictive decision process that evaluates candidate actions against both interaction uncertainty and physical feasibility.

As illustrated in Fig. 4(a), the proposed agent performs future-aware reasoning through parallel world-model rollouts. Based on the observed world state, the system generates a main-decision trajectory that represents the nominal racing intention. A delta-trajectory manifold is produced to encode multimodal deviations around this main trajectory. These delta trajectories are applied to the main-decision trajectory to form multimodal trajectories. To further diversify longitudinal behavior, the multimodal trajectories are combined with prescribed speed factors, yielding 36 candidate trajectories for world-model evaluation. These candidate trajectories are propagated through the world model to produce parallel futures. Each future jointly contains predicted opponent responses, ego-motion evolution, and feasibility information under near-limit conditions. The resulting futures are then evaluated according to safety, physical feasibility, and speed efficiency, enabling the agent to select a maneuver that is not only competitive, but also executable by the downstream limit-aware control module.

The predictive basis of this reasoning process is validated through a real-world wheel-to-wheel racing interaction. As shown in Fig. 4(b), the onboard image sequence captures a short-horizon overtaking process, where the realized trajectories of the ego vehicle and the opponent are compared with the corresponding world-model prediction. The predicted interaction remains consistent with the observed trajectories, indicating that the world model captures the coupled short-horizon evolution of ego and opponent motion. This consistency is critical for decision-making near the cognitive-physical limit, because prediction errors can cause the agent to misjudge whether an overtaking maneuver will remain safe, feasible, and competitive in the next few seconds.

The functional role of the world model for real-time execution is demonstrated by ablation in Fig. 4(c). Under the same racing scenario, future-aware reasoning supported by the world model leads to a feasible overtaking maneuver, whereas removing

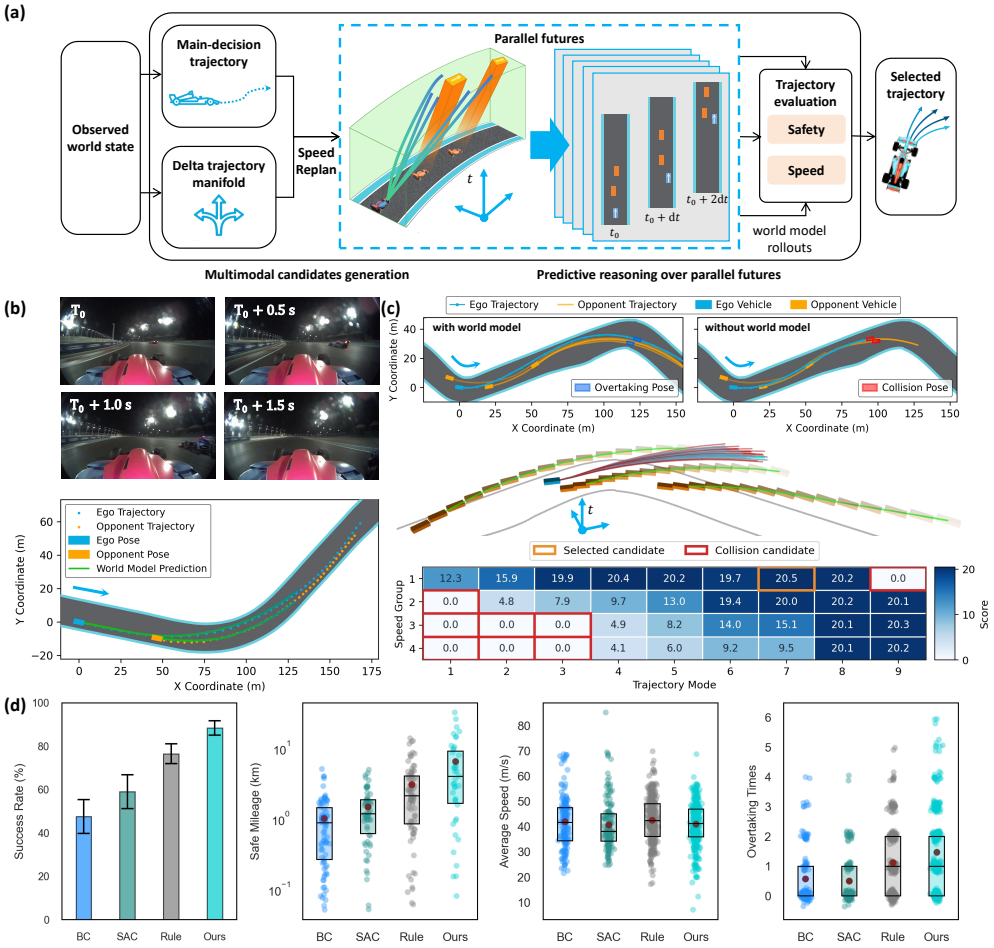


Fig. 4 Future-aware reasoning for adversarial interaction under coupled cognitive-physical limits. The world model supports multimodal trajectory generation, parallel future evaluation, and planning selection by jointly considering interaction safety, physical feasibility, and speed efficiency. (a) Future-aware reasoning process from observed world state to multimodal candidate generation and parallel future evaluation. (b) Real-world prediction validation during a wheel-to-wheel racing interaction. (c) Ablation of agent behavior with and without the world model, showing how prediction affects downstream physical execution. (d) Benchmark comparison of competitive interaction performance.

the world model results in a collision. With the world model, the agent generates multimodal candidate maneuvers in future predicted interactions and evaluates the combined speed and safety scores of the candidates. Collision futures are identified before execution, while the selected candidate achieves the highest score among feasible maneuvers and results in a successful overtaking pose. Without the world model, the agent lacks reliable anticipation of future interaction and physical feasibility, leading to a decision that is plausible from the current observation but unsafe when executed.

This comparison shows how world-model prediction directly affects downstream physical execution: insufficient anticipation of interaction evolution can turn a nominally reasonable decision into a physically unsafe maneuver.

Quantitative comparison across benchmark methods further demonstrates the effectiveness of the world-model-driven policy (Ours), as shown in Fig. 4(d). The proposed policy is evaluated in full-scale autonomous racing simulation for adversarial interaction against behavior cloning (BC), soft actor-critic (SAC), and a rule-based planner. The proposed method achieves the highest success rate, reaching 88.3%, corresponding to gains of 40.8% over BC and 29.3% over SAC. It also delivers the longest safe mileage, exceeding 6.4 km, while maintaining competitive average speed. Moreover, the higher overtaking frequency indicates that the performance gain is not obtained by conservative driving alone, but by more effective interaction-aware decision-making. World-model rollouts screen multimodal trajectory candidates in predicted interaction futures, filtering out maneuvers with safety risks before execution and thereby improving decision reliability.

These metrics jointly reflect the cognitive side of the limit: the agent must predict interaction evolution accurately enough to distinguish safe, feasible, and competitive maneuvers under severe time pressure. The results show that world-model rollouts link adversarial interaction reasoning with downstream limit-aware control. By evaluating candidate maneuvers against predicted opponent responses, safety risks, physical feasibility, and efficiency before execution, the agent selects safe actions that are both competitive and executable under joint cognitive-physical limits.

Limit-Aware Control near the Dynamics Boundary

Limit-aware control tests whether maneuvers selected by future-aware reasoning can be executed safely near the dynamics boundary of the vehicle. In autonomous racing, this boundary is not a fixed speed or acceleration value, but a coupled feasible-motion envelope shaped by vehicle dynamics, actuator constraints, tire–road interaction, aerodynamic loading, road condition, tire temperature, and stability margins. The world model provides a predictive representation of this boundary through ego-dynamics prediction and physical-limit modeling, which regularizes candidate trajectories upstream and constrains the dynamics controller downstream.

Fig. 5(a) shows the full-lap racing trajectory and the corresponding vehicle states projected into the a_x–a_y–v_x space. This projection provides a vehicle-level view of physical-limit utilization by comparing coupled longitudinal–lateral acceleration states with the feasible dynamics envelope informed by the world model. Along the track, aggressive braking before corners and strong acceleration at corner exits are reflected in the variation of a_x, showing active exploitation of the vehicle's coupled motion capability. The autonomous racing vehicle reaches a maximum speed of 71.2 m/s and maintains an average speed of 47.6 m/s over the full lap. The maximum longitudinal acceleration and deceleration reach approximately $1g$ and $3g$, respectively, while the peak lateral acceleration reaches 26 m/s^2. The vehicle states in Fig. 5(a) indicate that the vehicle motion is competitive while remaining within the feasible motion region. Within the world model, physical-limit exploration yields a vehicle-level acceleration envelope that characterizes the feasible boundary of coupled longitudinal–lateral

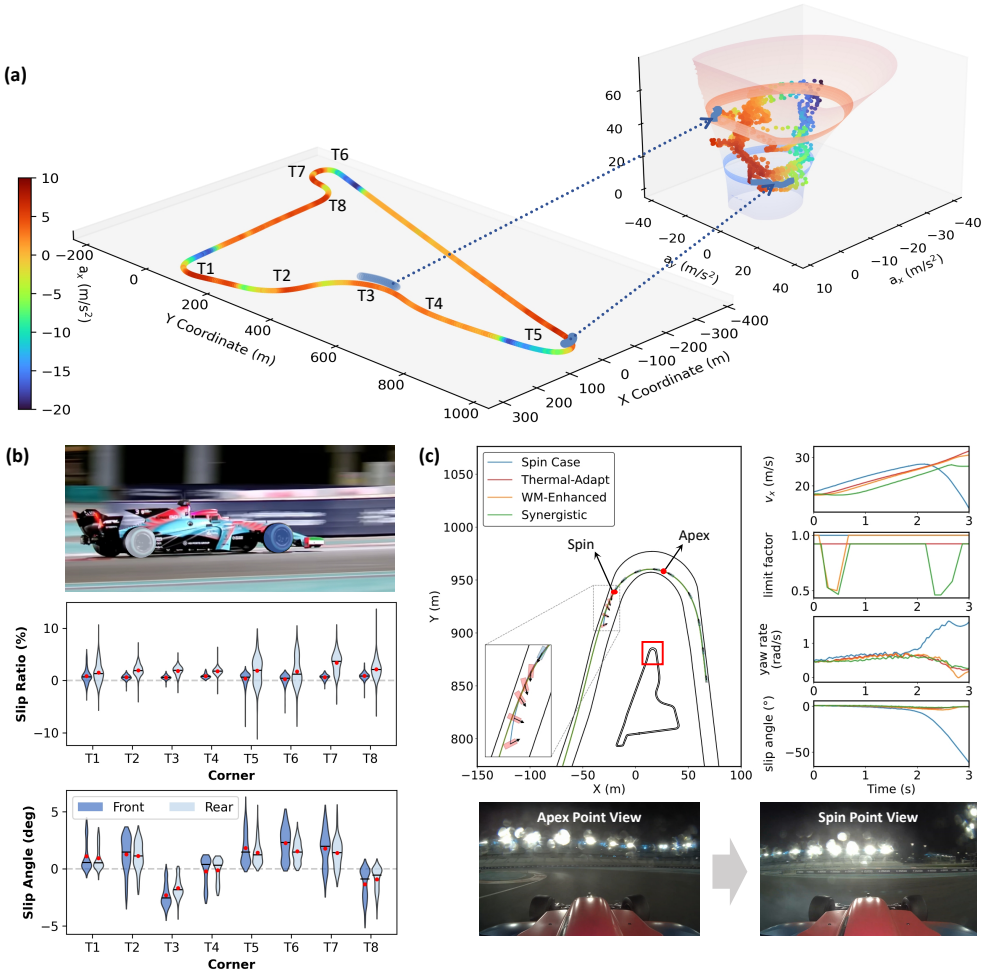


Fig. 5 Limit-aware high-speed dynamics control performance. World-model-informed physical-limit awareness constrains downstream control and enables aggressive high-speed operation near the feasible dynamics boundary while maintaining stability. (a) Full-lap trajectory and vehicle states in the a_x–a_y–v_x space, showing operation near the feasible dynamics envelope. (b) Tire-level limit utilization by slip ratio and slip angle. (c) Spin suppression through adaptive degradation of the effective physical limit.

motion. By imposing this envelope as a systematic constraint of the racing agent, the vehicle motion is regulated to approach the physical boundary for performance exploitation while avoiding persistent excursions beyond the feasible region.

The vehicle-level envelope is supported by tire-level force generation and closed-loop controllability. As shown in Fig. 5(b), tire utilization across the T1–T8 corners is characterized by slip ratio and slip angle distributions. The slip ratio peaks at approximately $\pm 10\%$, while the slip angles of both axles remain within about $\pm 5°$, indicating operation close to the stable tire-force generation region without persistent

saturation. The downstream controller also maintains accurate tracking and stability during high-speed near-limit racing: the mean absolute lateral tracking error is 0.11 m and the mean absolute speed tracking error is 0.34 m/s, with peak values of 0.69 m and 3.23 m/s, respectively. The mean absolute yaw-rate discrepancy is 0.073 rad/s, with a peak value of 0.525 rad/s. These peak errors correspond mainly to short-duration excursions during aggressive braking, cornering, or acceleration, rather than persistent tracking degradation or instability. The physical-limit acceleration envelope moderates downstream tire-force demand and prevents persistent excursions beyond tire-force limits, thereby enabling safe and high-performance maneuvering near the boundary.

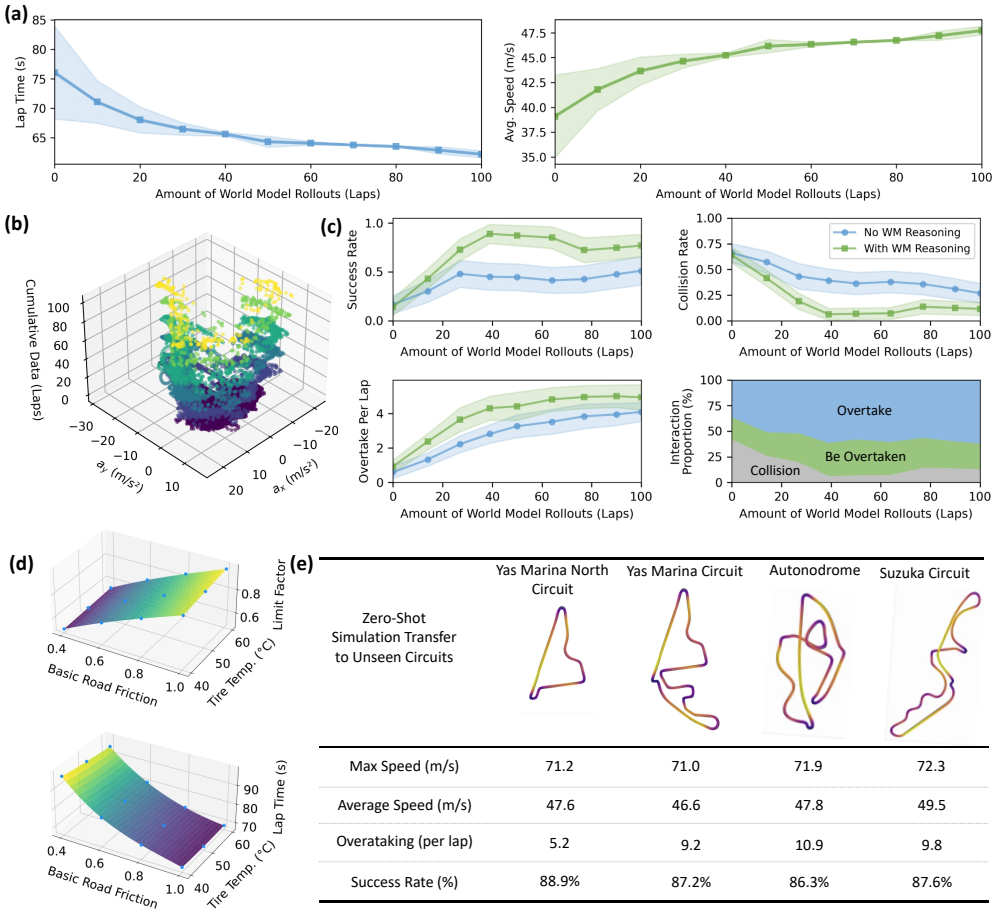
The role of world-model-informed physical-limit-aware adaptation is further highlighted by the failure and recovery scenario in Fig. 5(c). In real vehicle racing, when the safety-car phase ends, a sudden acceleration command near the apex of a high-curvature corner, combined with high lateral load and suboptimal tire temperature, causes abrupt traction loss and spin. To improve robustness near the physical boundary, three limit-degradation mechanisms are evaluated: a direct thermal-adaptive mechanism (Thermal-Adapt), a world-model-based prediction-deviation mechanism (WM-Enhanced), and a synergistic mechanism combining both sources of limit information. With the proposed approach, these mechanisms prevent the vehicle from spinning by reducing the effective physical boundary before instability develops. Thermal-Adapt and WM-Enhanced restrict the peak slip angle to -2.34° and -4.01°, respectively, while the synergistic mechanism applies a more conservative fused boundary and further restricts the peak slip angle to -1.43°, at the cost of reduced corner-exit performance. Physical-limit-aware adaptation therefore improves aggressive driving safety by adjusting the effective acceleration envelope according to thermal state and world-model prediction deviation. By contracting the admissible motion boundary before instability fully develops, it suppresses excessive acceleration demand and helps maintain stable vehicle motion during near-limit maneuvers.

Limit-aware control therefore forms the execution layer of the cognitive-physical framework. The world model provides predicted ego dynamics and feasible-motion boundaries, future-aware reasoning selects trajectories under safety, speed, and feasibility criteria, and the controller executes the selected maneuver while respecting tire, stability, and acceleration-envelope constraints. This closes the loop between upstream prediction, competitive decision-making, and near-limit physical execution.

Agent Evolution Through the World Model

The world model enables the agent to refine its use of coupled cognitive-physical limits over time. Rather than improving racing performance through policy optimization alone, the framework updates predictive models of ego dynamics, interaction evolution, and physical feasibility, and uses them to improve both future-aware reasoning and limit-aware control. Agent evolution is therefore reflected not only in faster lap times, but also in expanded physical-limit utilization, safer competitive interaction, and adaptability across operating conditions.

The physical side of world-model-driven agent evolution is reflected in the physical boundary expansion and lap-time reduction. Fig. 6(a) shows that lap time decreases



	Yas Marina North Circuit	Yas Marina Circuit	Autonodrome	Suzuka Circuit
Max Speed (m/s)	71.2	71.0	71.9	72.3
Average Speed (m/s)	47.6	46.6	47.8	49.5
Overataking (per lap)	5.2	9.2	10.9	9.8
Success Rate (%)	88.9%	87.2%	86.3%	87.6%

Fig. 6 Agent evolution through the world model. Closed-loop refinement improves racing performance, interaction capability, and robustness across conditions. (a) Improved racing performance through cumulative world-model data. (b) Progressive expansion of reachable states in the a_x–a_y–v_x space during agent evolution. (c) Improved interaction performance with increasing world-model rollouts. (d) Adaptation under varying tire temperatures and road friction. (e) Zero-shot transfer to unseen circuits.

while average speed increases as cumulative world-model data grows. Across the refinement process, the average lap time improves by 17.87%, while the average speed increases by 22.02%. This improvement is accompanied by expansion of the reachable acceleration states. As shown in Fig. 6(b), the distribution of vehicle states in the longitudinal–lateral acceleration space expands toward the feasible acceleration envelope, indicating that the agent exploits a larger portion of the vehicle's physical capability while maintaining feasible and stable motion.

The cognitive side of this evolution appears in improved interaction reasoning and decision-making. As shown in Fig. 6(c), reinforcement learning based on world-model rollouts generates interactive scenarios that are scarce in real racing, allowing the

agent to improve competitive behavior under adversarial interaction. With accumulated world-model refinement, the interaction success rate increases from 42.6% to 88.9%, reaching a peak of 91.7%; the average overtaking count per lap increases from 2.78 to 4.80; and the collision rate per interaction decreases from 41.9% to 5.8%, with a minimum of 4.0%. Compared with the ablated variant without future-aware reasoning, the proposed method improves success rate by 34.7%, increases overtaking frequency by 23.4%, and reduces collision rate by 19.8%. These improvements indicate that the refined world model helps the agent distinguish actions that are safe, feasible, and competitive under interaction uncertainty.

Fig. 6(d) further shows how cognitive and physical limits are jointly regulated under changing operating conditions. The ego limit factor is adaptively adjusted under different tire temperatures and road friction levels, and the corresponding lap-time surface reflects how performance varies as the available physical boundary changes. Instead of treating the physical limit as fixed, the agent adapts the effective boundary to current conditions so that planning and control remain compatible with the vehicle's available capability.

The evolved agent also generalizes across unseen circuits in simulation, as shown in Fig. 6(e). After training on the Yas Marina North Circuit, the agent was transferred without additional training to the Yas Marina Full GP layout, Autonodrome, and Suzuka Circuit. These circuits differ in track length, curvature distribution, corner density, high-speed straight sections, and overtaking opportunities, thereby testing whether the refined agent preserves cognitive reasoning and physical-limit-aware execution under changed track geometry. Maximum speeds remain around 70 m/s across all circuits, while overtaking frequency varies with track characteristics. The success rate in 20 s random interactive scenarios remains close to 90%, indicating consistent interaction performance in zero-shot transfer. These results show that world-model refinement supports the co-evolution of cognitive reasoning and physical-limit utilization, enabling the agent to better predict interaction futures, select competitive yet feasible maneuvers, exploit the dynamics envelope, and adapt the effective physical boundary across conditions.

Discussion

This work uses autonomous racing as an extreme testbed for exploring the cognitive-physical limits of embodied intelligence. The goal is not merely to develop a faster racing system, but to investigate how an embodied agent can construct, predict, exploit, and regulate its capability boundaries under simultaneous cognitive and physical constraints. In this setting, cognitive limits arise from high-frequency localization and perception, interaction prediction, and decision-making under severe time pressure, while physical limits arise from vehicle dynamics, actuator constraints, tire–road interaction, thermal variation, and stability margins. The results show that these limits are coupled: uncertainty in perception or interaction prediction affects how safely the physical envelope can be exploited, while the evolving physical state constrains which future actions remain feasible and safe.

A central implication is that world models can serve as a bridge between cognitive reasoning and physical control in extreme embodied tasks. In the proposed racing agent, the world model jointly predicts external interaction evolution, ego dynamics, and feasible motion boundaries. This allows candidate maneuvers to be evaluated before execution in terms of opponent response, safety risk, physical feasibility, and speed efficiency. As shown by the interaction-reasoning and dynamics-control results, this predictive structure transforms adversarial reasoning into executable near-limit maneuvers while constraining downstream control near the feasible dynamics envelope.

The results also suggest that embodied capability can evolve through closed-loop refinement of world modeling, decision-making, and physical-limit regulation. As the world model is refined, the agent improves lap time, average speed, acceleration-state coverage, interaction success, overtaking capability, collision avoidance, and adaptation to changing tire and road conditions. This improvement is therefore not a single-dimensional performance gain, but a coupled evolution of cognitive reasoning and physical-limit utilization.

Beyond autonomous racing, the present work suggests a boundary-aware methodology for embodied intelligent systems. For robots operating in the real world, safety and performance depend not only on selecting desirable actions, but also on whether those actions remain predictable, feasible, and safe under current sensing, interaction, and physical constraints. This perspective shifts the design objective from optimizing isolated modules toward modeling and regulating coupled system-level boundaries. High-speed mobile robots, agile aerial vehicles, legged robots on risky terrain, human–robot collaborative systems, and manipulation systems with rich contact dynamics all involve different forms of such boundary awareness: localization uncertainty can reduce control margins, inaccurate prediction of human or agent motion can make a feasible action unsafe, and changing contact, friction, or load conditions can invalidate a previously reasonable plan.

Several limitations remain. The current framework is developed and validated in autonomous racing, where task structure, vehicle dynamics, track geometry, and interaction patterns are more constrained than in many open-world robotic applications. Extending this approach will require richer world representations, broader uncertainty modeling, and stronger mechanisms for semantic ambiguity, human intent, deformable contact, and unmodeled environmental changes. Moreover, the results do not establish an absolute theoretical upper bound of embodied intelligence; rather, they operationalize cognitive and physical limits through observable indicators such as localization robustness, interaction prediction, decision safety, dynamic-envelope utilization, tire-level behavior, and stability margins. Overall, this work suggests that cognitive and physical limits should be treated as coupled system-level design objects rather than isolated module-level constraints, and that world-model-centric agents provide a concrete path toward representing, exploiting, and regulating these limits for safer real-world deployment.

Methods

System Formulation under Cognitive-Physical Limits

The proposed system is formulated as a world-model-centric architecture for autonomous racing near coupled cognitive-physical limits. The agent integrates current-state construction, predictive reasoning, and motion control through a shared world model. At runtime, the world model predicts both interaction evolution and ego dynamics, allowing candidate maneuvers to be evaluated against future safety, racing efficiency, and physical feasibility before execution. Over longer time scales, real racing data refine the world model, and the refined world model provides structured predictions for updating decision-making, planning, and limit-aware control.

At time step t, the agent receives surrounding observation o_t, ego vehicle state x_t, and local track information Γ_t. These inputs are integrated into a structured world-state representation

$$w_t = \Psi_w(x_t, o_t, \Gamma_t), \tag{1}$$

where $\Psi_w(\cdot)$ denotes the state representation function. Here, w_t contains not only the basic ego, opponent, and track information, but also evaluation-related information that can be derived from the structured scene, such as collision status, boundary violation, and physical feasibility. Using the world model W, the future evolution of the world state $\hat{w}_{t:t+H}$ is predicted over a finite horizon H.

Based on the current world state, the decision policy π generates a main racing decision. Together with a transformer-based multimodal planner, this decision produces a set of candidate trajectories,

$$\mathcal{T}_t = \left\{\tau_t^k\right\}_{k=1}^{K}, \tag{2}$$

where K is the number of candidate trajectories over the planning horizon H. Each candidate trajectory is propagated through the world model to generate a corresponding future. The candidate set is then evaluated using a task-oriented score that jointly considers interaction safety, physical feasibility, and speed efficiency. The optimal trajectory τ^* is then selected from $\mathcal{T}_t$ based on their future-evaluation scores. This formulation allows the agent to choose maneuvers that are competitive in the racing task while remaining safe and physically executable.

Given the selected trajectory τ_t^*, the motion-control module computes executable commands by solving a constrained optimization problem,

$$u_t^* = \arg\min_{u_{t:t+H_c-1}} J(x_t, u_{t:t+H_c-1}; \tau_t^*, \mathcal{E}_t), \tag{3}$$

subject to

$$x_{t+j+1} = f_{\mathrm{dyn}}(x_{t+j}, u_{t+j}), \quad j = 0, \ldots, H_c - 1, \tag{4}$$

$$h(x_{t+j}, u_{t+j}, \mathcal{E}_{t+j}) \le 0, \quad j = 0, \ldots, H_c - 1, \tag{5}$$

where x_t is the ego vehicle state, $u_t = [a_x, \delta]^\top$ contains longitudinal acceleration and steering commands, $f_{\mathrm{dyn}}(\cdot)$ is the control-oriented vehicle dynamics model, $\mathcal{E}_{t+j}$ denotes the feasible-motion envelope estimated from the world model, and $h(\cdot)$ enforces coupled longitudinal–lateral motion feasibility. The reference state x_{t+j}^{ref} is induced by the selected trajectory τ_t^* and enters the tracking objective in $J(\cdot)$. In this way, the

downstream controller executes the selected maneuver while respecting the current physical capability of the ego vehicle.

Beyond online decision-making and control, the architecture supports closed-loop refinement of the world model and the policy. Real racing data $\mathcal{D}_{\text{real}}$ are used to train the world model through supervised prediction of ego dynamics, interaction evolution, and physical feasibility. The refined world model then generates simulated rollouts $\mathcal{D}_{\text{wm}}$ under controlled variations of interaction and near-limit operating conditions. The combined dataset $\mathcal{D}$ is used to update the decision and planning models and to refine physical-limit constraints. This closed-loop process links real racing data, world-model prediction, policy improvement, and limit-aware control, enabling progressive improvement under coupled cognitive and physical limits.

Predictive World Modeling for the Autonomous Racing Agent

The world model is implemented as a structured predictive module of the autonomous racing agent operating near cognitive-physical limits. It supports online future-aware reasoning and limit-aware control, while also enabling offline policy refinement through structured internal inference. Starting from the world-state representation w_t, which integrates the ego vehicle state, surrounding opponents, and local track context, the world model is composed of three predictive components: an ego dynamics model, an interaction prediction model, and a physical-limit representation. These components jointly predict how the ego vehicle, the surrounding racing scene, and the feasible motion boundary evolve under near-limit conditions.

The ego dynamics model provides short-horizon prediction of vehicle motion under aggressive control inputs. It adopts a physics-constrained residual-learning formulation, in which the next-step state is represented by a differentiable dynamics backbone with residual correction:

$$\hat{x}_{t+1} = f_{\text{dyn}}(x_t, u_t; \theta_{\text{dyn}}) + f_{\text{res}}(x_{t-h:t}, u_{t-h:t}; \theta_{\text{res}}) \,, \tag{6}$$

where f_{dyn} provides a structured state transition and f_{res} captures unmodeled nonlinear effects. The physics backbone preserves the main coupling among longitudinal motion, lateral motion, yaw dynamics, and wheel rotational behavior, while the dynamics-related parameters θ_{dyn} are constrained within physically plausible ranges through bounded learnable parameters. The residual model, parameterized by θ_{res}, further refines the prediction using recent state and control histories. This formulation allows the model to capture transient nonlinearities that are difficult to analytically describe while retaining physical coherence under near-limit maneuvering.

The interaction prediction model captures the evolution of multi-agent racing scenarios. Given the structured world state, surrounding opponents are represented by their relative spatial relations to the ego vehicle and organized into position-aware slots. An attention-based interaction encoder, consisting of multilayer perceptrons followed by a masked attention-pooling layer, embeds this relational context and predicts a latent interaction intention for each agent. Racing-relevant behaviors, including attack, defense, hold, and yield, are implicitly represented in this latent intention space. At inference time, the world model is recursively rolled out to generate future

interaction scenarios:

$$\hat{w}_{t:t+H} = W^H(w_t, a_{t:t+H-1}), \tag{7}$$

where $W^H(\cdot)$ denotes recursive world-model prediction over horizon H, $a_{t:t+H-1}$ denotes candidate action inputs or behavior hypotheses, and $\hat{w}_{t:t+H}$ denotes the predicted future world states. These predicted futures allow the agent to evaluate candidate trajectories against possible opponent responses rather than relying only on the current observation.

The physical-limit representation describes the feasible motion boundary of the ego vehicle. In autonomous racing, this boundary is shaped by vehicle dynamics, actuator constraints, tire–road interaction, aerodynamic loading, road friction, tire temperature, and stability margins, and is manifested at the vehicle level as a coupled longitudinal–lateral acceleration envelope. The envelope is represented as

$$\begin{cases} \left(\dfrac{a_{x,k}}{a^{\mathrm{acc}}_{x,\max}(v_{x,k},\mu_k)}\right)^c + \left(\dfrac{|a_{y,k}|}{a_{y,\max}(v_{x,k},\mu_k)}\right)^c \le 1, & a_{x,k} \ge 0, \\ \left(\dfrac{|a_{x,k}|}{a^{\mathrm{dec}}_{x,\max}(v_{x,k},\mu_k)}\right)^c + \left(\dfrac{|a_{y,k}|}{a_{y,\max}(v_{x,k},\mu_k)}\right)^c \le 1, & a_{x,k} < 0. \end{cases} \tag{8}$$

Here $a_{x,k}$ and $a_{y,k}$ denote longitudinal and lateral accelerations, $v_{x,k}$ is the longitudinal speed, μ_k is the road-friction coefficient, and c controls the shape of the coupled envelope. The acceleration limits $a^{\mathrm{acc}}_{x,\max}$, $a^{\mathrm{dec}}_{x,\max}$, and $a_{y,\max}$ describe speed- and friction-dependent acceleration, braking, and lateral capabilities.

A nominal envelope $\mathcal{E}_0$ is induced from the dynamics parameters θ_{dyn} identified by the differentiable dynamics model, providing a structured prior of the vehicle's motion capability. Because the effective physical boundary varies with operating conditions and model mismatch, physical-limit exploration and online adaptation are introduced to scale the nominal envelope:

$$\mathcal{E}_t = \mathcal{E}(\rho_a(w_t), \rho_e; \mathcal{E}_0(\theta_{\mathrm{dyn}})), \tag{9}$$

The exploration vector ρ_e enlarges the envelope from a conservative initialization as near-limit racing data accumulate, while the adaptive factor $\rho_a(w_t)$ modulates the envelope online according to road friction, tire temperature, and world-model prediction error. This allows the effective envelope $\mathcal{E}_t$ to approach the available physical capability without exceeding the prescribed limit criterion.

Through these three components, the world model provides a predictive representation of both cognition-relevant and physics-relevant future evolution. Ego dynamics prediction estimates how the vehicle responds near the physical boundary; interaction prediction anticipates how opponents may react; and physical-limit representation determines which future motions remain feasible. This predictive representation supplies the basis for future-aware reasoning, limit-aware control, and closed-loop refinement of the agent.

Quantifying Cognitive and Physical Limits

The cognitive and physical limits in this work are quantified through observable indicators rather than treated as absolute theoretical bounds. These indicators measure how the agent constructs the current racing state, predicts future evolution, selects competitive actions, and executes them near the feasible motion boundary. Detailed indicator definitions, normalization thresholds, and implementation settings are provided in Supplementary Methods Section S3.

The cognitive limit is evaluated along the perception–prediction–decision chain. Current-state construction is quantified by localization and ego-state residuals,

$$\mathbf{e}_t^{\text{state}} = \left[|e_{x,t}|,\ |e_{y,t}|,\ |e_{v_x,t}|,\ |e_{v_y,t}|,\ |e_{\psi,t}|\right], \tag{10}$$

where $e_{x,t}$ and $e_{y,t}$ are longitudinal and lateral position residuals, $e_{v_x,t}$ and $e_{v_y,t}$ are velocity residuals, and $e_{\psi,t}$ is the heading residual. Opponent perception is further assessed by the temporal consistency of tracked opponent states. Predictive capability is evaluated by ego-dynamics prediction errors and interaction-prediction accuracy. For short-horizon ego–opponent interaction prediction, average displacement error and final displacement error are defined as

$$\text{ADE} = \frac{1}{T}\sum_{i=1}^{T} \|\hat{o}_{t+i} - o_{t+i}\|_2, \tag{11}$$

$$\text{FDE} = \|\hat{o}_{t+T} - o_{t+T}\|_2, \tag{12}$$

where $\hat{o}_{t+i}$ and o_{t+i} denote the predicted and ground-truth opponent states at prediction step i. Closed-loop decision capability is measured by interaction success rate, collision rate, safe mileage, and overtaking frequency. The interaction success rate is defined as

$$R_{\text{succ}} = \frac{N_{\text{succ}}}{N_{\text{total}}}, \tag{13}$$

where N_{succ} the number of successfully completed interaction trials and N_{total} is the total number of interaction trials. These metrics characterize whether the agent can distinguish maneuvers that are safe, feasible, and competitive under severe time pressure and interaction uncertainty.

The physical limit is evaluated by how closely the ego vehicle approaches and maintains feasible motion under near-limit execution. At the vehicle level, acceleration-envelope utilization is defined by the normalized envelope function

$$\eta_{\text{env},t} = g_{\mathcal{E}}\left(a_{x,t}, a_{y,t}, v_{x,t}, \mu_t; \mathcal{E}_t\right), \tag{14}$$

where $g_{\mathcal{E}}(\cdot)$ is induced by the coupled acceleration envelope in Eq. (8). A value $\eta_{\text{env},t} \leq 1$ indicates that the executed longitudinal–lateral acceleration remains inside the feasible envelope, while values close to one indicate operation near the physical boundary.

To capture physical-limit phenomena across different layers, we further define a hybrid physical-limit utilization,

$$\eta_{\mathrm{lim},t} = \max\left(\eta_{\mathrm{tire},t}, \eta_{\mathrm{control},t}, \eta_{\mathrm{stability},t}\right), \tag{15}$$

where $\eta_{\mathrm{tire},t}$ is computed from tire slip ratio and slip angle, $\eta_{\mathrm{control},t}$ from lateral and speed tracking errors, and $\eta_{\mathrm{stability},t}$ from yaw-rate discrepancy. These terms respectively characterize tire–road force utilization, trajectory tracking capability, and maneuvering stability. The metric $\eta_{\mathrm{env},t}$ further represents $\eta_{\mathrm{lim},t}$ with the feasible envelope, thereby quantifying the proximity of the current state to the feasible motion boundary. Their detailed normalized forms are provided in Supplementary Methods.

The cognitive and physical indicators are interpreted jointly. Localization and prediction errors quantify cognitive uncertainty; acceleration-envelope utilization, tire behavior, tracking errors, and yaw-rate discrepancy quantify physical-limit utilization; and closed-loop outcomes such as success rate, collision rate, safe mileage, and overtaking frequency reveal how the two sides interact during execution. A maneuver is effective only when it is strategically competitive, predicted to be safe, and executable within the current feasible-motion envelope.

World-Model-Driven Future-Aware Reasoning

Future-aware reasoning integrates decision-making, multimodal trajectory generation, and predictive evaluation around the world model. The decision and planning models are trained offline with real and world-model-generated data, while online deployment uses world-model rollouts to evaluate candidate maneuvers before execution.

At runtime, the structured world state w_t is input to a decision policy π, which outputs a compact racing intention containing longitudinal and lateral targets. This decision is converted into a main reference trajectory through a transition model:

$$\tau_t^{\mathrm{ref}} = f_{\mathrm{tran}}(x_t, \pi(w_t; \theta_\pi), H), \tag{16}$$

where θ_π denotes the parameters of the decision policy, $f_{\mathrm{tran}}(\cdot)$ is the transition model, and H is the planning horizon. The decision policy uses a structured interaction encoder that represents the ego vehicle, surrounding opponents, and relative racing context, and outputs longitudinal–lateral intentions for downstream trajectory generation.

To capture multiple possible interaction strategies, a multimodal delta planner generates trajectory deviations from recent world-state history. These deviations are added to the main reference trajectory and regularized by the current feasible-motion envelope $\mathcal{E}_t$:

$$\mathcal{T}_t = R\big(\tau_t^{\mathrm{ref}} + \phi(w_{t-h:t}; \theta_\phi), \mathcal{E}_t\big), \tag{17}$$

where $\phi(\cdot)$ is the delta trajectory planning network, θ_ϕ denotes its parameters, h is the history length, and $R(\cdot, \mathcal{E}_t)$ denotes physical-limit regularization. The planner encodes ego history, opponent history, and local track context with transformer-based modules, and outputs multimodal candidate trajectories representing different competitive behaviors. These multimodal trajectories are further combined with prescribed speed

factors to diversify longitudinal behavior. The regularization step ensures that candidate trajectories are compatible with the current physical capability of the ego vehicle before predictive evaluation.

Each candidate trajectory is then evaluated through recursive world-model rollouts. Following Eq. (7), the world model predicts the future world-state evolution induced by each candidate, yielding a set of parallel futures:

$$\mathcal{F}_t = \left\{ \left(\tau_t^k, \hat{w}_{t:t+H}^k \right) \mid \tau_t^k \in \mathcal{T}_t \right\}. \tag{18}$$

Each parallel future is scored by a task-oriented function that considers interaction safety, physical feasibility, and speed efficiency. Candidate trajectories that lead to predicted collision, boundary violation, or loss of control are rejected, and the selected trajectory is

$$\tau_t^* = \arg\max_{\tau_t^k \in \mathcal{T}_t} S\left(\tau_t^k, \hat{w}_{t:t+H}^k \right). \tag{19}$$

This procedure selects trajectories that are safe, feasible, and competitive before execution. By combining physical-limit regularization with world-model rollout evaluation, future-aware reasoning links adversarial interaction prediction to downstream limit-aware control.

Physical-Limit-Constrained Motion Control

The motion-control module converts the trajectory selected by future-aware reasoning into executable commands under the current physical-limit envelope. Given the selected trajectory τ_t^* and the world-model-informed feasible-motion envelope $\mathcal{E}_t$, control is formulated as a constrained model predictive control problem. The controller tracks τ_t^* while enforcing vehicle dynamics, road-boundary constraints, actuator limits, and the coupled longitudinal–lateral acceleration constraint induced by $\mathcal{E}_t$. The detailed vehicle dynamics model, Jacobians, and barrier derivatives are provided in Supplementary Methods Section S6.

At each control step, the controller solves the constrained optimization problem described in Eq. (3)–Eq. (5). Given the selected trajectory τ_t^*, it optimizes a finite-horizon control sequence to track the induced reference states while satisfying the control-oriented vehicle dynamics and the feasible-motion constraints defined by the physical-limit envelope. The control input $u_t = [a_x, \delta]^\top$ includes the longitudinal acceleration and steering command, and the envelope $\mathcal{E}_{t+j}$ constrains the coupled longitudinal–lateral feasibility estimated from the world model. This formulation enables the selected maneuver to be executed as physically feasible control commands under the current capability of the ego vehicle.

The cost is decomposed into a tracking-and-smoothness term and a constraint-barrier term,

$$J = J_{\text{track}}(x_t, u_{t:t+H_c-1}; \tau_t^*) + J_{\text{barrier}}(x_t, u_{t:t+H_c-1}; \mathcal{E}_t). \tag{20}$$

The tracking term is

$$J_{\text{track}} = \sum_{j=0}^{H_c-1} \left(\|x_{t+j} - x_{t+j}^{\text{ref}}\|_Q^2 + \|u_{t+j}\|_R^2 + \|\Delta u_{t+j}\|_P^2 \right), \tag{21}$$

where x_{t+j}^{ref} is induced by τ_t^*. Inequality constraints are imposed through exponential barriers,

$$J_{\text{barrier}} = \sum_{k,i} q_1^{(i)} \exp\Big(q_2^{(i)} c_i(x_k, u_{k-1}; \mathcal{E}_k) \Big), \tag{22}$$

where $c_i \leq 0$ indicates satisfaction of the corresponding constraint and $c_i > 0$ indicates violation. The physical-limit barrier is induced by the coupled acceleration envelope in Eq. (8), so violations of the current feasible-motion boundary are penalized directly during optimization.

The constrained optimization is solved using a constrained iterative linear quadratic regulator (CiLQR). At each iteration, the dynamics and costs are locally approximated, a backward pass computes the feedforward correction and feedback gain, and a forward rollout tests the updated control sequence. The update takes the affine feedback form

$$u_k^{\text{new}} = u_k + \alpha d_k + \gamma K_k (x_k^{\text{new}} - x_k), \tag{23}$$

where d_k is the feedforward correction, K_k is the feedback gain, α is the line-search parameter, and γ scales the feedback term. This receding-horizon formulation allows the controller to regulate speed and steering before the vehicle violates the physical-limit envelope, thereby executing the selected maneuver under near-limit conditions.

Closed-Loop World-Model and Policy Refinement

The world model and the policy are refined through a closed-loop process that links real racing data, predictive model learning, internal simulation, and policy update. During real-world racing, the onboard system collected high-quality dynamic data under real road, tire, vehicle, and interaction conditions, forming a dataset $\mathcal{D}_{\text{real}}$. These data are used in model training to align the world model with observed ego dynamics, opponent interactions, and physical-limit behavior.

The ego dynamics model and interaction model are updated through supervised prediction. The differentiable dynamics backbone and residual model are trained by minimizing the next-state prediction error:

$$(\theta_{\text{dyn}}, \theta_{\text{res}}) = \arg\min_{\theta_{\text{dyn}}, \theta_{\text{res}}} \sum_{\mathcal{D}_{\text{real}}} \|x_{t+1} - f_{\text{dyn}}(x_t, u_t; \theta_{\text{dyn}}) - f_{\text{res}}(x_{t-h:t}, u_{t-h:t}; \theta_{\text{res}})\|^2 . \tag{24}$$

The interaction model is updated by supervised prediction of world-state evolution:

$$\theta_{\text{int}} = \arg\min_{\theta_{\text{int}}} \sum_{\mathcal{D}_{\text{real}}} \|w_{t+1} - W(w_t; \theta_{\text{int}})\|^2 , \tag{25}$$

where $W(\cdot;\theta_{\text{int}})$ denotes the learned interaction transition model. The physical-limit model is initialized conservatively and refined through world-model-based limit exploration. At each update step, a safe exploration policy proposes an envelope expansion based on local track context and limit-utilization information. The expansion is accepted only when the predicted rollout with the updated envelope satisfies the prescribed limit criterion, enabling progressive exploration of the feasible motion boundary without violating safety constraints.

The refined world model is then used to generate structured rollout data under controlled variations of interaction and near-limit operating conditions:

$$\mathcal{D}_{\text{wm}} = \text{Rollout}(W, \pi, \phi, \mathcal{E}). \tag{26}$$

Compared with real racing data alone, $\mathcal{D}_{\text{wm}}$ can cover a broader range of adversarial interactions, corner cases, and near-limit scenarios. The real and world-model-generated datasets are combined as

$$\mathcal{D} = \mathcal{D}_{\text{real}} \cup \mathcal{D}_{\text{wm}}, \tag{27}$$

which is used to update the decision policy and multimodal planner.

The decision policy is updated using a soft actor–critic objective with conservative Q-learning and behavior-cloning regularization. Given state s_t, corresponding to the world state w_t, the policy samples a racing intention $a_t^{\pi} \sim \pi_{\theta_\pi}(\cdot|s_t)$ containing longitudinal and lateral decision components. The critic parameters are optimized by

$$\begin{aligned} L_Q(\theta_Q) =& \mathbb{E}_{(s_t,a_t,s_{t+1})\sim\mathcal{D}}\left[\left(Q_{\theta_Q}(s_t,a_t) - y_t\right)^2\right] \\ &+ \lambda_{\text{cql}}\left\{\mathbb{E}_{s_t\sim\mathcal{D}}\left[Q_{\theta_Q}(s_t,a_t^{\pi})\right] - \mathbb{E}_{(s_t,a_t)\sim\mathcal{D}}\left[Q_{\theta_Q}(s_t,a_t)\right]\right\}, \end{aligned} \tag{28}$$

and the policy is optimized by

$$L_\pi(\theta_\pi) = \mathbb{E}_{(s_t,a_t^{\text{e}})\sim\mathcal{D}}\Big[\alpha \log \pi_{\theta_\pi}(a_t^{\pi}|s_t) - \min(Q_1,Q_2) + \lambda_{\text{bc}}\left\|a_t^{\pi} - a_t^{\text{e}}\right\|^2\Big], \tag{29}$$

where a_t^{e} denotes the demonstration action in the dataset. The conservative term discourages overestimation on out-of-distribution actions, while the behavior-cloning term preserves useful behavior from real vehicle data and simulated data.

The multimodal planner is updated using best-of-K supervision. Hybrid data are reorganized into pairs of world-state history and future trajectory. Given the historical segment $w_{t-h:t}$, the planner generates candidate trajectories as in Eq. (17), and only the output mode closest to the ground-truth future trajectory is supervised:

$$\theta_\phi = \arg\min_{\theta_\phi} \sum_{\mathcal{D}} \left[\min_{\tau_t^k \in \mathcal{T}_t} \left\|\tau_t^{\text{gt}} - \tau_t^k\right\|^2\right], \tag{30}$$

where τ_t^{gt} denotes the ground-truth future trajectory corresponding to the segment $w_{t:t+H}$ extracted from $\mathcal{D}$. This training strategy allows different output heads to specialize in different behavioral modes.

This refinement process closes the loop between real-world deployment and world-model-based learning. Real racing data improve the fidelity of ego-dynamics, interaction, and physical-limit modeling; the refined world model generates structured rollouts for rare interactions and near-limit conditions; and the updated policy and planner collect more informative data in subsequent deployment. Through this cycle, the agent progressively improves future prediction, interaction reasoning, and physical-limit utilization under coupled cognitive-physical limits.

Experimental Platform and Validation

A full-scale autonomous racing platform was used for collection of real-vehicle data collection with physics grounding. Real-vehicle deployment of the base software stack exposed the agent to extreme racing conditions and provided high-value data for world-model refinement. The racing task included high-speed operation, near-limit tire–road conditions, and adversarial multi-agent interactions, making it a stringent source of real-world data for studying embodied intelligence near cognitive and physical limits.

During real-vehicle experiments, state, control, and environment data were collected from the onboard system. These real-world racing data were used to train and refine the world model, including ego dynamics, interaction evolution, and physical-limit representation. Rather than treating real-world deployment as the final validation of the proposed method, the collected data were utilized and incorporated into a closed-loop data-driven refinement pipeline: competition deployment generated real racing data, the data updated the world model, and the refined world model supported subsequent policy improvement.

To evaluate the agent updated with the refined world model under controlled and repeatable interactive conditions, validation was conducted in the Autoverse simulator with virtual opponent vehicles. This setup provides high-fidelity vehicle dynamics simulation, allowing the updated agent to be tested in adversarial multi-vehicle scenarios with repeatable initial conditions. Interaction scenarios of 20 s duration were constructed to reproduce representative competitive situations, with opponent behaviors generated by imitation learning from real racing interactions. In each trial, 2 to 4 opponent vehicles with racing behaviors were generated within a longitudinal range extending 80 m ahead of and 50 m behind the ego vehicle, with their lateral offsets sampled from $\mathcal{N}(0, 2.5^2)$. A trial is considered successful if the ego vehicle reaches the end of the 20 s evaluation horizon without collision, track-boundary violation, or unrecoverable instability; otherwise, it is considered failed.

The proposed method was compared with three baselines: behavioral cloning (BC), soft actor–critic (SAC), and the state-machine-based rule planner used in the competition stack. BC and SAC serve as representative imitation-learning and model-free reinforcement-learning baselines, while the rule-based planner serves as the baseline corresponding to the real-vehicle competition stack. All learning-based approaches were trained using the same training data. Each method was evaluated in 360 trials, of which two were excluded due to simulation failures, leaving 358 valid trials.

Beyond statistical performance comparison, the validation further reproduced overtaking and spin scenarios observed in real-world racing. The overtaking case was used to assess whether the world model could predict short-horizon interaction evolution

during wheel-to-wheel racing. The spin scenario was used to evaluate whether world-model-informed physical-limit-aware adaptation could correct a failure mode observed during the race by improving recovery and reducing instability. This validation pipeline links real-world deployment, data collection, world-model refinement, policy testing in simulation, and failure-case reproduction, thereby supporting continuous agent improvement under coupled cognitive-physical limits.

References

[1] Pfeifer, R., Bongard, J.: How the Body Shapes the Way We Think: a New View of Intelligence. MIT press, ??? (2006)

[2] Duan, J., Yu, S., Tan, H.L., Zhu, H., Tan, C.: A survey of embodied ai: From simulators to research tasks. IEEE Transactions on Emerging Topics in Computational Intelligence **6**(2), 230–244 (2022)

[3] Spielberg, N.A., Brown, M., Kapania, N.R., Kegelman, J.C., Gerdes, J.C.: Neural network vehicle models for high-performance automated driving. Science robotics **4**(28), 1975 (2019)

[4] Liu, Y., Chen, W., Bai, Y., Liang, X., Li, G., Gao, W., Lin, L.: Aligning cyber space with physical world: A comprehensive survey on embodied ai. IEEE/ASME Transactions on Mechatronics (2025)

[5] Yifan, C., Wei, M., Wang, X., Liu, Y., Wang, J., Song, H., Ma, L., Di, D., Sun, C., Liu, K., *et al.*: Embodied ai: A survey on the evolution from perceptive to behavioral intelligence. SmartBot **1**(3), 70003 (2025)

[6] Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., Mané, D.: Concrete problems in ai safety. arXiv preprint arXiv:1606.06565 (2016)

[7] Betz, J., Zheng, H., Liniger, A., Rosolia, U., Karle, P., Behl, M., Krovi, V., Mangharam, R.: Autonomous vehicles on the edge: A survey on autonomous vehicle racing. IEEE Open Journal of Intelligent Transportation Systems **3**, 458–488 (2022)

[8] Kaufmann, E., Bauersfeld, L., Loquercio, A., Müller, M., Koltun, V., Scaramuzza, D.: Champion-level drone racing using deep reinforcement learning. Nature **620**(7976), 982–987 (2023)

[9] Ames, A.D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., Tabuada, P.: Control barrier functions: Theory and applications. In: 2019 18th European Control Conference (ECC), pp. 3420–3431 (2019). Ieee

[10] Hanover, D., Loquercio, A., Bauersfeld, L., Romero, A., Penicka, R., Song, Y., Cioffi, G., Kaufmann, E., Scaramuzza, D.: Autonomous drone racing: A survey. IEEE Transactions on Robotics **40**, 3044–3067 (2024)

[11] Sun, Q., Fang, J., Zheng, W.X., Tang, Y.: Aggressive quadrotor flight using curiosity-driven reinforcement learning. IEEE Transactions on Industrial Electronics **69**(12), 13838–13848 (2022)

[12] Geles, I., Bauersfeld, L., Romero, A., Xing, J., Scaramuzza, D.: Demonstrating Agile Flight from Pixels without State Estimation (2024). https://arxiv.org/abs/2406.12505

[13] Hwangbo, J., Lee, J., Dosovitskiy, A., Bellicoso, D., Tsounis, V., Koltun, V., Hutter, M.: Learning agile and dynamic motor skills for legged robots. Science Robotics **4**(26), 5872 (2019)

[14] Wu, J., Xin, G., Qi, C., Xue, Y.: Learning robust and agile legged locomotion using adversarial motion priors. IEEE Robotics and Automation Letters **8**(8), 4975–4982 (2023)

[15] Zhang, C., Rudin, N., Hoeller, D., Hutter, M.: Learning agile locomotion on risky terrains. In: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 11864–11871 (2024). IEEE

[16] Huang, W., Shan, Z., Lou, S., Lv, C.: Uncertainty-aware reinforcement learning for autonomous driving with multimodal digital driver guidance. In: 2024 IEEE International Conference on Robotics and Automation (ICRA), pp. 18355–18361 (2024). IEEE

[17] Shan, Z., Zhao, J., Huang, W., Zhao, Y., Ge, L., Zhong, S., Hu, H., Lv, C., Zhu, B.: Deep imitative reinforcement learning with gradient conflict-free for decision-making in autonomous vehicles. Transportation Research Part C: Emerging Technologies **173**, 105047 (2025)

[18] Funke, J., Theodosis, P., Hindiyeh, R., Stanek, G., Kritatakirana, K., Gerdes, C., Langer, D., Hernandez, M., Müller-Bessler, B., Huhnke, B.: Up to the limits: Autonomous audi tts. 2012 IEEE intelligent vehicles symposium, 541–547 (2012). IEEE

[19] Liniger, A., Lygeros, J.: A noncooperative game approach to autonomous racing. IEEE Transactions on Control Systems Technology **28**(3), 884–897 (2019)

[20] Wang, M., Wang, Z., Talbot, J., Gerdes, J.C., Schwager, M.: Game-theoretic planning for self-driving cars in multivehicle competitive scenarios. IEEE Transactions on Robotics **37**(4), 1313–1325 (2021)

[21] O'Kelly, M., Zheng, H., Karthik, D., Mangharam, R.: F1tenth: An open-source evaluation environment for continuous control and reinforcement learning. Proceedings of Machine Learning Research **123** (2020)

[22] Baumann, N., Hu, C., Sivasothilingam, P., Qin, H., Xie, L., Magno, M., Benini, L.:

Enhancing autonomous driving systems with on-board deployed large language models. arXiv preprint arXiv:2504.11514 (2025)

[23] Baumann, N., Ghignone, E., Kühne, J., Bastuck, N., Becker, J., Imholz, N., Kränzlin, T., Lim, T.Y., Lötscher, M., Schwarzenbach, L., *et al.*: Forzaeth race stack—scaled autonomous head-to-head racing on fully commercial off-the-shelf hardware. Journal of Field Robotics **42**(4), 1037–1079 (2025)

[24] Becker, J., Imholz, N., Schwarzenbach, L., Ghignone, E., Baumann, N., Magno, M.: Model-and acceleration-based pursuit controller for high-performance autonomous racing. arXiv preprint arXiv:2209.04346 (2022)

[25] Li, Z., Zhou, B., Piccinini, M., Hu, C., Zarrouki, B., Mangharam, R., Xie, L.: Evo-mpcc: Enhanced velocity optimization with learning-based auto-tuning for real-time vehicle trajectory planning. Available at SSRN 6127037 (2026)

[26] Hu, C., Huang, J., Mao, W., Fu, Y., Chi, X., Qin, H., Baumann, N., Liu, Z., Magno, M., Xie, L.: Fsdp: Fast and safe data-driven overtaking trajectory planning for head-to-head autonomous racing competitions. In: 2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5824–5831 (2025). IEEE

[27] Ghignone, E., Baumann, N., Magno, M.: Tc-driver: A trajectory-conditioned reinforcement learning approach to zero-shot autonomous racing. IEEE Transactions on Field Robotics **1**, 527–536 (2024)

[28] Heilmeier, A., Wischnewski, A., Hermansdorfer, L., Betz, J., Lienkamp, M., Lohmann, B.: Minimum curvature trajectory planning and control for an autonomous race car. Vehicle System Dynamics (2020)

[29] Piccinini, M., Taddei, S., Pagot, E., Bertolazzi, E., Biral, F.: How optimal is the minimum-time manoeuvre of an artificial race driver? Vehicle System Dynamics **63**(12), 2213–2240 (2025)

[30] Zarrouki, B., Spanakakis, M., Betz, J.: A safe reinforcement learning driven weights-varying model predictive control for autonomous vehicle motion control. In: 2024 IEEE Intelligent Vehicles Symposium (IV), pp. 1401–1408 (2024). IEEE

[31] Kabzan, J., Valls, M.I., Reijgwart, V.J., Hendrikx, H.F., Ehmke, C., Prajapat, M., Bühler, A., Gosala, N., Gupta, M., Sivanesan, R., *et al.*: Amz driverless: The full autonomous racing system. Journal of Field Robotics **37**(7), 1267–1294 (2020)

[32] Wischnewski, A., Geisslinger, M., Betz, J., Betz, T., Fent, F., Heilmeier, A., Hermansdorfer, L., Herrmann, T., Huch, S., Karle, P., et al.: Indy autonomous challenge-autonomous race cars at the handling limits. 12th International Munich Chassis Symposium 2021: chassis. tech plus, 163–182 (2022). Springer

[33] Betz, J., Betz, T., Fent, F., Geisslinger, M., Heilmeier, A., Hermansdorfer, L., Herrmann, T., Huch, S., Karle, P., Lienkamp, M., *et al.*: Tum autonomous motorsport: An autonomous racing software for the indy autonomous challenge. Journal of Field Robotics **40**(4), 783–809 (2023)

[34] Liniger, A., Domahidi, A., Morari, M.: Optimization-based autonomous racing of 1: 43 scale rc cars. Optimal Control Applications and Methods **36**(5), 628–647 (2015)

[35] Scheffe, P., Henneken, T.M., Kloock, M., Alrifaee, B.: Sequential convex programming methods for real-time optimal trajectory planning in autonomous vehicle racing. IEEE Transactions on Intelligent Vehicles **8**(1), 661–672 (2022)

[36] Alcalá, E., Puig, V., Quevedo, J., Rosolia, U.: Autonomous racing using linear parameter varying-model predictive control (lpv-mpc). Control Engineering Practice **95**, 104270 (2020)

[37] Wischnewski, A., Euler, M., Gümüs, S., Lohmann, B.: Tube model predictive control for an autonomous race car. Vehicle System Dynamics **60**(9), 3151–3173 (2022)

[38] Jung, C., Lee, S., Seong, H., Finazzi, A., Shim, D.H.: Game-theoretic model predictive control with data-driven identification of vehicle model for head-to-head autonomous racing. arXiv preprint arXiv:2106.04094 (2021)

[39] Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., *et al.*: Mastering the game of go with deep neural networks and tree search. nature **529**(7587), 484–489 (2016)

[40] Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., *et al.*: A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. Science **362**(6419), 1140–1144 (2018)

[41] Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., Guez, A., Lockhart, E., Hassabis, D., Graepel, T., *et al.*: Mastering atari, go, chess and shogi by planning with a learned model. Nature **588**(7839), 604–609 (2020)

[42] Wurman, P.R., Barrett, S., Kawamoto, K., MacGlashan, J., Subramanian, K., Walsh, T.J., Capobianco, R., Devlic, A., Eckert, F., Fuchs, F., *et al.*: Outracing champion gran turismo drivers with deep reinforcement learning. Nature **602**(7896), 223–228 (2022)

[43] Song, Y., Romero, A., Müller, M., Koltun, V., Scaramuzza, D.: Reaching the limit in autonomous racing: Optimal control versus reinforcement learning. Science Robotics **8**(82), 1462 (2023)

[44] Zhao, F., Zhang, H., Li, Z., Guan, X., Li, S.: Vision-guided mppi for agile drone racing: Navigating arbitrary gate poses via neural signed distance fields. arXiv preprint arXiv:2603.07199 (2026)

[45] Wurman, P.R., Stone, P., Spranger, M.: Challenges and opportunities of applying reinforcement learning to autonomous racing. IEEE Intelligent Systems **37**(3), 20–23 (2022)

[46] Ji, X., He, X., Lv, C., Liu, Y., Wu, J.: Adaptive-neural-network-based robust lateral motion control for autonomous vehicle at driving limits. Control Engineering Practice **76**, 41–53 (2018)

[47] Fuchs, F., Song, Y., Kaufmann, E., Scaramuzza, D., Dürr, P.: Super-human performance in gran turismo sport using deep reinforcement learning. IEEE Robotics and Automation Letters **6**(3), 4257–4264 (2021)

[48] Pan, Y., Cheng, C.-A., Saigol, K., Lee, K., Yan, X., Theodorou, E.A., Boots, B.: Imitation learning for agile autonomous driving. The International Journal of Robotics Research **39**(2-3), 286–302 (2020)

[49] Sun, X., Zhou, M., Zhuang, Z., Yang, S., Betz, J., Mangharam, R.: A benchmark comparison of imitation learning-based control policies for autonomous racing. 2023 IEEE Intelligent Vehicles Symposium (IV), 1–5 (2023). IEEE

[50] Hamilton, N., Musau, P., Lopez, D.M., Johnson, T.T.: Zero-shot policy transfer in autonomous racing: reinforcement learning vs imitation learning. 2022 IEEE International Conference on Assured Autonomy (ICAA), 11–20 (2022). IEEE

[51] Prudencio, R.F., Maximo, M.R., Colombini, E.L.: A survey on offline reinforcement learning: Taxonomy, review, and open problems. IEEE Transactions on Neural Networks and Learning Systems **35**(8), 10237–10257 (2023)

[52] Zare, M., Kebria, P.M., Khosravi, A., Nahavandi, S.: A survey of imitation learning: Algorithms, recent developments, and challenges. IEEE Transactions on Cybernetics (2024)

[53] Evans, B.D., Engelbrecht, H.A., Jordaan, H.W.: High-speed autonomous racing using trajectory-aided deep reinforcement learning. IEEE Robotics and Automation Letters (2023)

[54] Ivanov, R., Carpenter, T.J., Weimer, J., Alur, R., Pappas, G.J., Lee, I.: Case study: verifying the safety of an autonomous racing car with a neural network controller. Proceedings of the 23rd International Conference on Hybrid Systems: Computation and Control, 1–7 (2020)

[55] Hermansdorfer, L., Trauth, R., Betz, J., Lienkamp, M.: End-to-end neural network for vehicle dynamics modeling. 2020 6th IEEE Congress on Information

Science and Technology (CiSt), 407–412 (2021). IEEE

[56] Chen, H., Lou, S., Lv, C.: Hybrid physics-data-driven online modelling: Framework, methodology and application to electric vehicles. Mechanical Systems and Signal Processing **185**, 109791 (2023)

[57] Feng, T., Wang, X., Jiang, Y.-G., Zhu, W.: Embodied ai: From llms to world models. arXiv preprint arXiv:2509.20021 (2025)

[58] Ha, D., Schmidhuber, J.: Recurrent world models facilitate policy evolution. Advances in neural information processing systems **31** (2018)

[59] Hafner, D., Pasukonis, J., Ba, J., Lillicrap, T.: Mastering diverse domains through world models. arXiv preprint arXiv:2301.04104 (2023)

[60] Li, X., He, X., Zhang, L., Wu, M., Li, X., Liu, Y.: A comprehensive survey on world models for embodied ai. arXiv preprint arXiv:2510.16732 (2025)

[61] Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., Davidson, J.: Learning latent dynamics for planning from pixels. In: International Conference on Machine Learning, pp. 2555–2565 (2019). PMLR

[62] Ding, J., Zhang, Y., Shang, Y., Zhang, Y., Zong, Z., Feng, J., Yuan, Y., Su, H., Li, N., Sukiennik, N., *et al.*: Understanding world or predicting future? a comprehensive survey of world models. ACM Computing Surveys **58**(3), 1–38 (2025)

Acknowledgements

We would like to thank K2 Holding, Abu Dhabi, UAE, for its project support, and Alp Autonomy, Turkey, for its technical support. These contributions were instrumental in the implementation and experimental validation of this work. We also thank the members of the NTU AutoMan Research Lab and the former members of the Kinetiz autonomous racing team, particularly Binbin Hu, Wenhui Huang, Haohan Yang, Haochen Liu, Yiran Zhang, Tianchu Su, Andrea Piazzoni, Xiangkun He, Xiaoyu Mo, and Xinyu Zhou, for their valuable contributions to system development, vehicle integration, testing, and experimental deployment.

Funding

This work was supported by K2 Holding L.LC, Abu Dhabi, UAE, under Research Collaboration Agreement (No. REQ0474834).

Competing interests

The authors declare no competing interests.

Data availability

The datasets generated and/or analyzed during the current work are available from the corresponding author on reasonable request.

Code availability

The source code developed in current work is available from the corresponding author on reasonable request.

Supplementary information

Supplementary information accompanies this paper and includes Supplementary Methods S1–S10 and Supplementary Videos 1–5.

Appendix A Supplementary Methods

S1. Full System Architecture and Data Flow

This section provides an implementation-level overview of the world-model-centric autonomous racing agent and its full-scale racing platform. The main manuscript defines the system-level formulation under coupled cognitive–physical limits; here we further introduce the full-scale racing platform, notation, online data flow, and closed-loop refinement pipeline used by the implemented system. The purpose of this section is to connect the physical platform with the detailed supplementary modules, including world-state construction, world modeling, future-aware reasoning, physical-limit-constrained control, policy refinement, and safety monitoring.

S1.1 Full-Scale Racing Platform and Hardware Domains

The proposed agent is implemented on a full-scale autonomous racing platform equipped with multi-modal sensing, onboard computation, actuation interfaces, and a real-time autonomy software stack, as shown in Fig. A1. The racing vehicle is organized into three hardware domains: the powertrain domain, the autonomous-driving domain, and the chassis domain. Together, these domains provide the physical basis for high-speed sensing, onboard computation, and executable actuation under near-limit racing conditions.

The autonomous-driving domain contains core sensing and computing units, including GNSS, 4D millimeter-wave radars, LiDARs, and an onboard computing unit. These components support ego-state estimation, opponent perception, world-state construction, and onboard decision-making and control computation. Ego-state estimation uses measurements from LiDAR, RTK GNSS, IMU, wheel-speed sensors. Opponent perception uses LiDAR and 4D millimeter-wave radar to detect and track surrounding vehicles under high-speed and close-proximity racing conditions. The powertrain domain supplies driving power through a 6-speed gearbox and a 2.0-L turbocharged racing engine. The chassis domain provides steer-by-wire and brake-by-wire actuation interfaces that realize the steering and braking commands on the vehicle. This hardware organization connects the physical racing platform with the world-model-centric autonomy stack.

S1.2 World-State Representation and Notation

At each time step t, the agent receives multi-modal sensory observations, the ego vehicle state estimate, and local track information. We denote the surrounding observations by o_t, the ego vehicle state by x_t, and the local track context by Γ_t. These inputs are integrated into a structured world-state representation w_t.

The ego vehicle state contains the pose, velocity, heading, and relevant dynamic variables required by prediction, planning, and control. The surrounding observation o_t contains object-level opponent states obtained from LiDAR–radar perception and tracking. The local track context Γ_t contains the track-coordinate representation, road boundaries, curvature, and local reference information. The resulting world state w_t

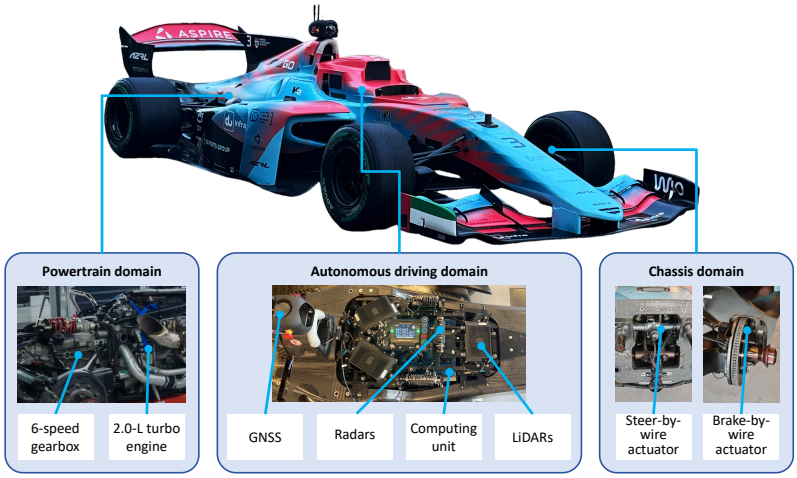


Fig. A1 Full-scale autonomous racing platform and onboard hardware.

provides a common representation shared by the world model, decision policy, multimodal planner, and control module. For consistency, we use the following notation throughout the Supplementary Methods:

x_t : ego vehicle state,
o_t : surrounding object observations,
Γ_t : local track context,
w_t : structured world state,
$\mathcal{E}_t$: world-model-informed feasible-motion envelope,
τ_t^k : the k-th candidate trajectory,
τ_t^* : selected trajectory,
u_t : control input.

The feasible-motion envelope $\mathcal{E}_t$ represents the current physical-limit boundary of the ego vehicle and is used both in trajectory regularization and in downstream constrained control. Unless otherwise specified, the term "agent" refers to the complete autonomous racing system, while "ego vehicle" refers to the physical racing platform.

S1.3 Online Perception–Prediction–Decision–Control Loop

The online system follows a perception–prediction–decision–control loop. First, high-frequency localization and opponent perception construct the current world state. The ego state is estimated from LiDAR, RTK GNSS, IMU, wheel-speed, and other onboard measurements, while opponent states are obtained by LiDAR–radar object detection and multi-object tracking. Details of the world-state construction process are provided in Supplementary Section S2.

Given the current world state, the world model predicts the future evolution of ego dynamics, opponent interaction, and physical-limit availability over a short planning horizon. These predictions provide the basis for evaluating how different candidate maneuvers may evolve under dynamic multi-agent racing conditions. The world model consists of ego dynamics prediction, interaction prediction, and physical-limit representation, as described in Supplementary Section S4.

The decision policy and multimodal planner then generate a set of candidate trajectories. These candidates correspond to different maneuver hypotheses, such as maintaining the current line, braking and following, or performing an overtaking maneuver. Candidate trajectories are regularized by the feasible-motion envelope to ensure compatibility with the physical capability of the ego vehicle. The world model evaluates the predicted future associated with each candidate trajectory according to interaction safety, physical feasibility, and speed efficiency. The candidate with the highest overall suitability is selected for execution. During onboard execution, the future-aware reasoning module operates at 10 Hz evaluating 36 candidate trajectories. Details of candidate generation and future-aware scoring are provided in Supplementary Section S5.

Finally, the motion-control module converts the selected trajectory into executable control commands. The controller tracks the selected trajectory while enforcing vehicle dynamics, road-boundary constraints, actuator limits, and coupled longitudinal–lateral physical-limit constraints. The control-oriented dynamics, Jacobians, barrier terms, and constrained CiLQR implementation are provided in Supplementary Section S6.

The online loop can therefore be summarized as

$$(x_t, o_t, \Gamma_t) \to w_t \to \left\{\hat{w}^k_{t:t+H}\right\}_{k=1}^{K} \to \tau_t^* \to u_t^*.$$

This flow links current-state construction, world-model prediction, future-aware decision-making, and physical-limit-constrained execution within a unified closed-loop racing agent.

S1.4 Closed-Loop Refinement Pipeline

Beyond online execution, the system is refined through a closed-loop pipeline that links real racing data, world-model learning, internal simulation, and policy update. Real-vehicle deployment produces data containing ego dynamics, opponent interactions, physical-limit behavior, and track context under real operating conditions. These data provide the basis for improving the agent after each deployment phase.

The world model is trained and refined using real racing data. The ego dynamics component learns the vehicle response under high-speed and near-limit operation, the interaction component learns the temporal evolution of multi-agent racing scenarios. The refined world model is then used to generate additional rollout data covering interaction variations, rare competitive scenarios, and near-limit operating conditions that are difficult to collect exhaustively in real racing.

The real and world-model-generated data are combined to update the decision policy, multimodal planner, and physical-limit constraints. The decision policy is updated

using conservative reinforcement learning with behavior-cloning regularization, while the planner is updated through best-of-K trajectory supervision. The physical-limit representation is refined using validated near-limit data so that the agent can progressively improve its utilization of the available performance envelope. Details of this refinement process are provided in Supplementary Section S7.

The refinement pipeline can be summarized as

$$\mathcal{D}_{\text{real}} \rightarrow W \rightarrow \mathcal{D}_{\text{wm}} \rightarrow \mathcal{D} \rightarrow (\pi, \phi, \mathcal{E}) \rightarrow \text{next deployment}.$$

This closed-loop process improves the fidelity of world-model prediction, expands the coverage of interaction and near-limit scenarios, and progressively refines the agent's decision-making and physical-limit utilization.

S2. High-Frequency World-State Construction

This section describes how the current racing world state w_t is constructed from high-frequency ego-state estimation, multi-modal sensing, and opponent perception. The resulting world state provides the observable foundation for world-model prediction, future-aware reasoning, and physical-limit-constrained control.

S2.1 Ego Vehicle State and Motion Model

The overall multi-sensor ego-state estimation framework is shown in Fig. A2, where LiDAR map matching, RTK GNSS, optical sensing, wheel encoders, and IMU measurements are fused with delay compensation, adaptive covariance estimation, innovation gating, and EKF update to estimate position, orientation, velocity, and sensor time delay.

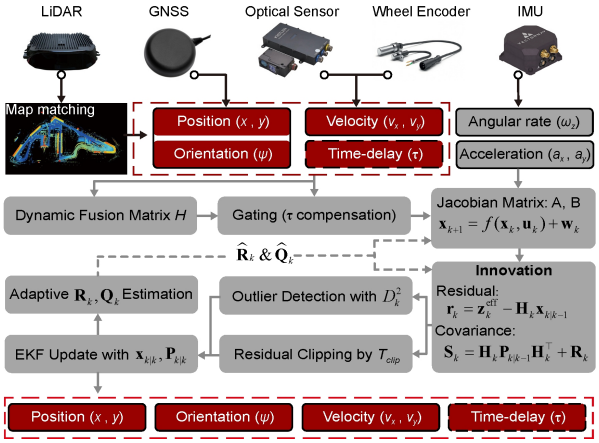


Fig. A2 Multi-sensor ego-state estimation framework for high-frequency world-state construction.

The ego-state estimation module maintains a compact state of the ego vehicle in the global track frame,

$$\mathbf{x} = \big(\mathbf{R},\, \mathbf{p},\, \mathbf{v},\, \mathbf{b}_g,\, \mathbf{b}_a,\, \mathbf{g}\big), \tag{A1}$$

where $\mathbf{R} \in SO(3)$ is the body-to-track orientation, $\mathbf{p}, \mathbf{v} \in \mathbb{R}^3$ are the position and velocity in the track frame, $\mathbf{b}_g, \mathbf{b}_a \in \mathbb{R}^3$ are gyroscope and accelerometer biases, and $\mathbf{g} \in \mathbb{R}^3$ is the gravity vector.

The IMU measures angular velocity and specific force in the body frame,

$$\begin{aligned} \boldsymbol{\omega}_m &= \boldsymbol{\omega} + \mathbf{b}_g + \mathbf{w}_g, \\ \mathbf{a}_m &= \mathbf{a} + \mathbf{b}_a + \mathbf{w}_a, \end{aligned} \tag{A2}$$

where $\boldsymbol{\omega}$ and $\mathbf{a}$ denote the true angular velocity and specific force, and $\mathbf{w}_g$ and $\mathbf{w}_a$ are zero-mean white noises. The bias terms follow random walks driven by additional Gaussian noise. Gravity is assumed to be constant over the time scale of a racing lap.

Over one IMU sampling period Δt, the noise-free discrete-time propagation is

$$\begin{aligned} \mathbf{R}_{k+1} &= \mathbf{R}_k \exp((\boldsymbol{\omega}_{m,k} - \mathbf{b}_{g,k})^{\wedge} \Delta t)\,, \\ \mathbf{v}_{k+1} &= \mathbf{v}_k + (\mathbf{R}_k(\mathbf{a}_{m,k} - \mathbf{b}_{a,k}) + \mathbf{g}_k)\,\Delta t, \\ \mathbf{p}_{k+1} &= \mathbf{p}_k + \mathbf{v}_k \Delta t + \frac{1}{2}\,(\mathbf{R}_k(\mathbf{a}_{m,k} - \mathbf{b}_{a,k}) + \mathbf{g}_k)\,\Delta t^2, \\ \mathbf{b}_{g,k+1} &= \mathbf{b}_{g,k}, \qquad \mathbf{b}_{a,k+1} = \mathbf{b}_{a,k}, \qquad \mathbf{g}_{k+1} = \mathbf{g}_k. \end{aligned} \tag{A3}$$

Here $(\cdot)^{\wedge}$ denotes the mapping from a rotation vector to a skew-symmetric matrix. Orientation is updated through right multiplication by $\exp(\cdot)$, while linear quantities are propagated in Euclidean space.

S2.2 Multi-Sensor Measurement Models

All exteroceptive measurements are expressed as noisy observations of position, orientation, or velocity in the track frame. A generic measurement $\mathbf{z}_j$ is written as

$$\mathbf{z}_j = h_j(\mathbf{x}) + \mathbf{n}_j, \qquad \mathbf{n}_j \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_j), \tag{A4}$$

where $h_j(\cdot)$ selects and transforms the relevant components of $\mathbf{x}$, and $\mathbf{R}_j$ is the measurement covariance.

RTK GNSS and LiDAR map matching provide position observations,

$$\mathbf{z}_p = \mathbf{p} + \mathbf{n}_p. \tag{A5}$$

Dual-antenna GNSS heading and LiDAR odometry provide orientation observations,

$$\mathbf{z}_R \approx \mathbf{R} \exp(\mathbf{n}_R^{\wedge}). \tag{A6}$$

Wheel-speed encoders and the downward-looking optical-flow sensor provide body-frame velocity observations,

$$\mathbf{z}_v = \mathbf{R}^{\top} \mathbf{v} + \mathbf{n}_v, \tag{A7}$$

which constrain longitudinal and lateral velocity relative to the road surface. The covariance of each modality is identified from calibration runs and short test laps.

S2.3 Delayed Measurements and Factorized Covariance

High-speed autonomous racing requires delayed LiDAR and GNSS measurements to be inserted at their correct physical times. To support this, the estimator maintains a short history of recent motion states. The rapidly varying motion components are defined as

$$\mathbf{x}^{\mathrm{mot}} = (\mathbf{R}, \mathbf{p}, \mathbf{v}, \mathbf{b}_g), \tag{A8}$$

while slowly varying parameters $(\mathbf{b}_a, \mathbf{g})$ are maintained outside the history.

At estimator time k, the short-horizon history is

$$\mathcal{X}_k = \left\{\mathbf{x}_{k,1}^{\mathrm{mot}}, \ldots, \mathbf{x}_{k,N}^{\mathrm{mot}}\right\}, \qquad t_{k,1} > \cdots > t_{k,N}, \tag{A9}$$

where $\mathbf{x}_{k,1}^{\mathrm{mot}}$ corresponds to the current time and the remaining nodes represent older states. The history length and capacity satisfy

$$t_{k,1} - t_{k,N} \le T_{\mathrm{hist}}, \qquad N \le N_{\max}. \tag{A10}$$

At each IMU step, a new state is propagated using Eq. (A3) and appended to the front of $\mathcal{X}_k$. The oldest node is removed once the time horizon or capacity is exceeded.

A delayed measurement $\mathbf{z}_j$ time-stamped at t_j is associated with the closest history node,

$$i^\star = \arg\min_{i \in \{1,\ldots,N\}} |t_{k,i} - t_j|, \tag{A11}$$

subject to

$$|t_{k,i^\star} - t_j| < \tau_{\max}, \tag{A12}$$

where $\tau_{\max}$ is the maximum allowed time offset. If this condition is not satisfied, the measurement is discarded. Multiple delayed measurements available at the same estimator time are collected into an ensemble and updated jointly after linearization around the current prediction.

The delay sensitivity analysis of localization error is shown in Fig. A3, where the APE distributions remain stable under different assumed measurement delays.

The covariance of the stack history $\mathbf{X}_k$ is stored in a factorized form,

$$\mathbf{P}_k = \mathbf{S}_k^\top \mathbf{S}_k, \tag{A13}$$

where $\mathbf{S}_k$ is an upper-triangular square-root factor. Prediction and update operations are performed on $\mathbf{S}_k$ rather than on $\mathbf{P}_k$ directly, avoiding explicit matrix inversion and improving numerical conditioning as nodes are inserted and removed.

When process noise is injected or linear propagation is applied, covariance addition is implemented by stacking square-root factors and applying QR decomposition. Conceptually, if $\mathbf{S}_A$ and $\mathbf{S}_B$ are square-root factors of two covariance terms, a factor of their sum is obtained from the upper-triangular part of the QR factorization of

$$\begin{bmatrix}\mathbf{S}_A^\top & \mathbf{S}_B^\top\end{bmatrix}^\top. \tag{A14}$$

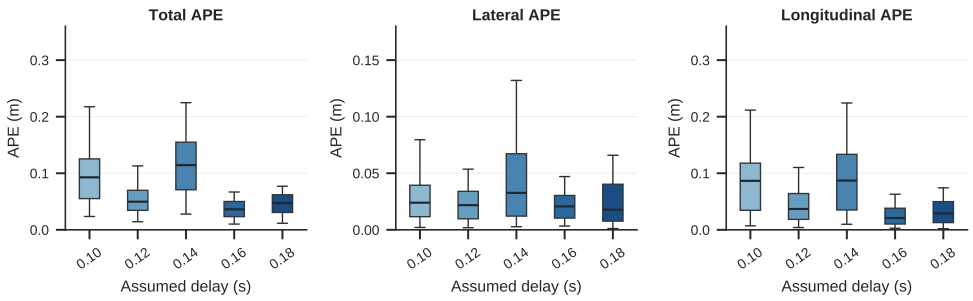


Fig. A3 Delay sensitivity analysis of localization error.

This operation is used when advancing the history with the IMU model and when compressing information from older nodes.

For measurement updates, the linearized ensemble model is

$$\mathbf{Z} \approx \mathbf{H}\delta\mathbf{X} + \mathbf{N}, \tag{A15}$$

where $\delta\mathbf{X}$ is the stacked perturbation of all history nodes, $\mathbf{H}$ is a block-sparse Jacobian, and $\mathbf{N}$ has block-diagonal covariance

$$\mathbf{R}_{\text{ens}} = \text{diag}(\mathbf{R}_1, \ldots, \mathbf{R}_M). \tag{A16}$$

The residuals and Jacobian are pre-whitened using a square-root factor of $\mathbf{R}_{\text{ens}}$, stacked with $\mathbf{S}_k$, and factorized by QR decomposition to obtain the updated factor $\mathbf{S}_k^+$ and the corresponding state correction.

S2.4 Robust Measurement Update

Occasional outliers can arise from GNSS multipath, partial LiDAR occlusion, sparse radar returns, or short-lived modeling errors. To reduce their effect, measurement innovations are weighted using a Huber-type robust loss. Let $\mathbf{r}_j$ denote the residual of measurement j with nominal covariance $\mathbf{R}_j$. Its Mahalanobis norm is

$$\|\mathbf{r}_j\|_{\mathbf{R}_j^{-1}} = \sqrt{\mathbf{r}_j^\top \mathbf{R}_j^{-1} \mathbf{r}_j}. \tag{A17}$$

The Huber cost is

$$\rho_H(\mathbf{r}_j) = \begin{cases} \frac{1}{2}\mathbf{r}_j^\top \mathbf{R}_j^{-1}\mathbf{r}_j, & \|\mathbf{r}_j\|_{\mathbf{R}_j^{-1}} \le \delta, \\ \delta\|\mathbf{r}_j\|_{\mathbf{R}_j^{-1}} - \frac{1}{2}\delta^2, & \|\mathbf{r}_j\|_{\mathbf{R}_j^{-1}} > \delta, \end{cases} \tag{A18}$$

where $\delta > 0$ is the Huber threshold. In an iteratively reweighted form, this gives the scalar weight

$$w(\mathbf{r}_j) = \begin{cases} 1, & \|\mathbf{r}_j\|_{\mathbf{R}_j^{-1}} \le \delta, \\ \dfrac{\delta}{\|\mathbf{r}_j\|_{\mathbf{R}_j^{-1}}}, & \|\mathbf{r}_j\|_{\mathbf{R}_j^{-1}} > \delta. \end{cases} \tag{A19}$$

The ensemble covariance is then adjusted as

$$\mathbf{R}_{\text{ens}} = \operatorname{diag}\left(w(\mathbf{r}_1)\mathbf{R}_1, \ldots, w(\mathbf{r}_M)\mathbf{R}_M\right). \tag{A20}$$

In practice, a single reweighting step per update is sufficient to suppress outliers while preserving sensitivity to consistent trends in the measurements.

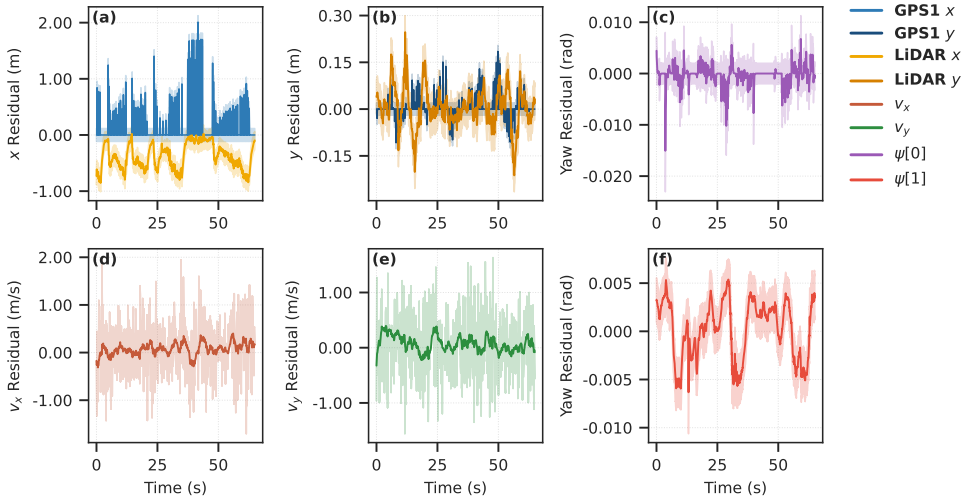


Fig. A4 Residual profiles of multi-sensor ego-state localization.

The resulting residual profiles for state estimation including the position, velocity, and yaw residuals over a full racing lap are shown in Fig. A4, indicating bounded and temporally stable innovations across GNSS, LiDAR, velocity, and yaw observations.

S2.5 Opponent Perception and Multi-Object Tracking

Opponent perception estimates the position, velocity, heading, and spatial extent of surrounding vehicles under high-speed and close-proximity racing conditions. LiDAR and 4D millimeter-wave radar are used as the primary exteroceptive sensors, and fusion is performed at the object-tracking level.

The point cloud observations from LiDAR and radar are shown in Fig. A5.

The opponent perception pipeline is illustrated in Fig. A6.

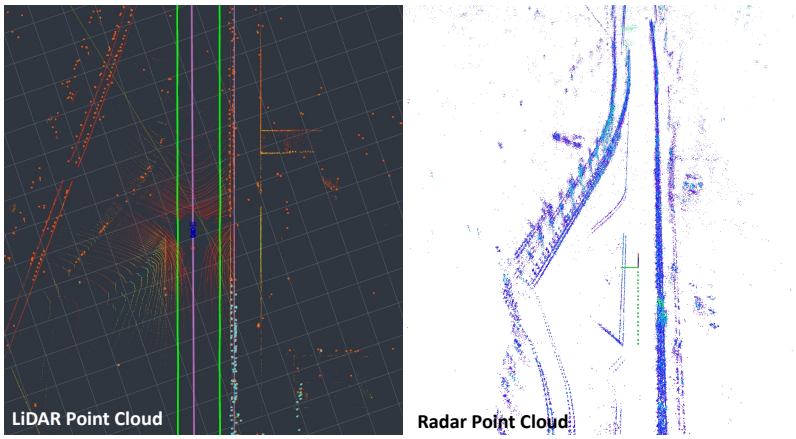


Fig. A5 LiDAR and radar point clouds during driving.

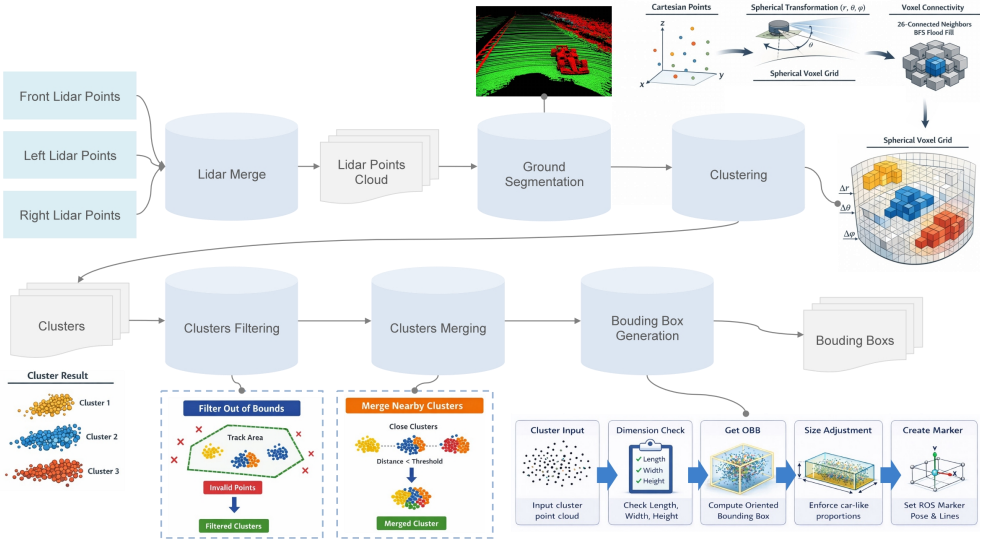


Fig. A6 Point cloud processing pipeline for opponent perception.

Raw LiDAR and radar point clouds are first transformed into a common vehicle-centric coordinate frame. Ground points are removed using an elevation-consistency model defined over radial bins under bounded slope assumptions. The maximum allowable height increase between consecutive radial rings is

$$\Delta z_{\max} = \min\left(\Delta r \tan(\text{slope}_{\max}), h_{\text{th}} - \varepsilon\right), \tag{A21}$$

where Δr is the radial increment, $\text{slope}_{\max}$ is the maximum admissible ground slope, h_{th} is the height threshold, and ε is a safety margin. Local plane refinement is further applied in the near field.

After ground removal, non-ground points are discretized in spherical voxel space,

$$(r, \text{az}, \text{el}) = \left(\sqrt{x^2 + y^2 + z^2}, \text{atan2}(y, x), \arctan\left(\frac{z}{\sqrt{x^2 + y^2}} \right) \right), \tag{A22}$$

where r, az, and el denote range, azimuth, and elevation. Connected components are extracted by neighborhood traversal in the voxel grid, and small clusters are removed. Spatially adjacent clusters are merged according to proximity and admissible vehicle dimensions. Hypotheses that violate geometric consistency are rejected.

Each retained cluster is represented by an oriented bounding box. Given a point set $\mathcal{P} = \{p_i\}_{i=1}^{N}$, the principal axes are obtained from the covariance matrix C by eigen decomposition,

$$Cv_j = \lambda_j v_j, \qquad j = 1, 2, 3, \tag{A23}$$

where λ_j and v_j denote eigenvalues and eigenvectors. The rotation matrix is

$$R_{\text{obb}} = [v_1 \;\; v_2 \;\; v_3], \tag{A24}$$

and axis-aligned extents are computed in the principal frame and mapped back to obtain the oriented bounding-box pose and dimensions. Radar point clouds follow the same pipeline with modality-specific clustering thresholds to account for sparse and anisotropic returns.

The extracted bounding boxes are treated as object-level measurements and passed to an EKF-based multi-object tracking module. Each tracked opponent is represented by

$$\mathbf{x}_o = (\mathbf{p}_o, \mathbf{v}_o, \psi_o), \tag{A25}$$

where $\mathbf{p}_o$ and $\mathbf{v}_o$ denote object position and velocity in the vehicle-centric frame, and ψ_o denotes object heading. Object hypotheses from LiDAR and radar are associated using spatial gating and track-consistency checks. Measurement updates are applied asynchronously as observations arrive, with modality-specific covariance settings retained in each measurement model. This tracking-level fusion allows LiDAR geometry and radar velocity cues to jointly support temporally coherent opponent states.

S2.6 Construction of the Racing World State

The racing world state used by the world model and downstream decision-making is assembled from ego-state estimation, opponent tracking, and local track information. At time t, the structured world state is written as

$$w_t = \Psi_w(x_t, o_t, \Gamma_t), \tag{A26}$$

where x_t denotes the ego vehicle state, o_t denotes the set of tracked opponent states, and Γ_t denotes local track context.

For the ego vehicle, the state includes the pose, velocity, heading, yaw rate, and other dynamic quantities required by ego-dynamics prediction and motion control. For each opponent, the tracked object state includes position, velocity, heading, and bounding-box information. The local track context includes the track-coordinate representation, local curvature, reference line, and left–right road boundaries.

This world-state representation provides a common interface for the three downstream components. The world model uses w_t to predict ego dynamics, interaction evolution, and physical feasibility. Future-aware reasoning uses w_t and predicted futures to generate and evaluate candidate trajectories. Physical-limit-constrained control uses the selected trajectory together with the feasible-motion envelope $\mathcal{E}_t$ to compute executable control commands. In this way, high-frequency localization and opponent perception serve as the observable foundation for predictive world understanding under cognitive-physical limits.

S3. Quantification of Cognitive and Physical Limits

The main manuscript evaluates cognitive and physical limits through observable indicators rather than claiming absolute theoretical upper bounds. This section provides the detailed definitions used to quantify these indicators. The cognitive side is evaluated along the perception–prediction–decision chain, while the physical side is evaluated through acceleration-envelope utilization, tire-force utilization, tracking capability, and maneuvering stability. These indicators are interpreted jointly to assess how the agent approaches and regulates coupled cognitive-physical limits during autonomous racing.

S3.1 Cognitive-Limit Indicators

The cognitive limit refers to the boundary of the agent's ability to construct, update, predict, and use task-relevant world understanding under severe time pressure, uncertainty, and dynamic interaction. For autonomous racing, this includes high-frequency localization and perception under high-speed motion, prediction of ego–opponent interaction evolution, and decision-making for safe, feasible, and competitive maneuvers.

The cognitive limit is not determined by a single error source. Instead, uncertainty propagates along the perception–prediction–decision chain. Localization and ego-state residuals introduce uncertainty into the reconstructed racing world state. This uncertainty propagates into ego-dynamics prediction and interaction prediction, where it affects future trajectory rollouts. Prediction errors then influence trajectory ranking and action selection, and eventually appear as closed-loop outcomes such as unsafe overtaking, collision, or overly conservative behavior. The cognitive limit is approached when accumulated uncertainty prevents the agent from reliably distinguishing actions that are safe, feasible, and competitive.

Current-state construction is quantified by localization and ego-state residuals,

$$\mathbf{e}_t^{\text{state}} = \left[|e_{x,t}|,\, |e_{y,t}|,\, |e_{v_x,t}|,\, |e_{v_y,t}|,\, |e_{\psi,t}|\right], \tag{A27}$$

where $e_{x,t}$ and $e_{y,t}$ denote longitudinal and lateral position residuals, $e_{v_x,t}$ and $e_{v_y,t}$ denote longitudinal and lateral velocity residuals, and $e_{\psi,t}$ denotes the heading residual. Opponent perception is assessed by the temporal consistency of tracked opponent states, including whether object identities, positions, velocities, and bounding boxes remain coherent under high-speed and close-proximity racing conditions.

Ego-dynamics prediction is evaluated by the prediction error between the predicted ego state and the corresponding measured or estimated state,

$$\mathbf{e}_{t+i}^{\text{ego}} = \hat{x}_{t+i} - x_{t+i}, \qquad i = 1, \ldots, T. \tag{A28}$$

For each ego-state channel, the root-mean-square error is computed as

$$\text{RMSE}_{x_j} = \sqrt{\frac{1}{NT} \sum_{n=1}^{N} \sum_{i=1}^{T} \left(\hat{x}_{j,t+i}^{(n)} - x_{j,t+i}^{(n)} \right)^2}, \tag{A29}$$

where x_j denotes the j-th ego-state variable, N is the number of evaluated sequences, and T is the prediction horizon.

Future-interaction prediction is evaluated using average displacement error and final displacement error. For an opponent trajectory predicted over a horizon T,

$$\text{ADE} = \frac{1}{T} \sum_{i=1}^{T} \left\| \hat{o}_{t+i} - o_{t+i} \right\|_2, \tag{A30}$$

$$\text{FDE} = \left\| \hat{o}_{t+T} - o_{t+T} \right\|_2, \tag{A31}$$

where $\hat{o}_{t+i}$ and o_{t+i} denote the predicted and ground-truth opponent states at prediction step i. In racing scenarios, these states are evaluated in a structured track-coordinate representation so that longitudinal progress, lateral offset, and interaction geometry are preserved.

Closed-loop decision capability is quantified by interaction success rate, collision rate, safe mileage, and overtaking frequency. The interaction success rate is

$$R_{\text{succ}} = \frac{N_{\text{succ}}}{N_{\text{total}}}, \tag{A32}$$

where N_{succ} denotes the number of successful interactions and N_{total} denotes the total number of interaction trials. The collision rate is

$$R_{\text{col}} = \frac{N_{\text{col}}}{N_{\text{total}}}, \tag{A33}$$

where N_{col} denotes the number of interactions that lead to collision. The overtaking frequency is computed as

$$F_{\text{over}} = \frac{N_{\text{over}}}{N_{\text{lap}}}, \tag{A34}$$

where N_{over} is the number of completed overtaking maneuvers and N_{lap} is the number of evaluated laps. Safe mileage is the accumulated distance travelled without collision, boundary violation, or emergency termination.

These cognitive indicators are complementary. State residuals measure the reliability of current world-state construction; ego-dynamics and interaction-prediction errors measure the quality of future prediction; and closed-loop outcomes measure whether the agent can convert state estimation and prediction into safe, feasible, and competitive behavior.

S3.2 Physical-Limit Indicators

The physical limit refers to the boundary of feasible motion imposed by vehicle dynamics, actuator constraints, tire–road interaction, aerodynamic loading, road friction, tire temperature, and stability constraints. For autonomous racing, this boundary is reflected not only by vehicle-level acceleration capability, but also by tire utilization, controller tracking capability, and maneuvering stability. Therefore, the physical limit is quantified using a set of layered indicators.

At the vehicle level, physical-limit utilization is evaluated by the executed acceleration state relative to the feasible-motion envelope $\mathcal{E}_t$. The coupled acceleration envelope is defined in the main manuscript as a speed- and friction-dependent boundary in the (a_x, a_y) plane. We define the normalized envelope utilization as

$$\eta_{\text{env},t} = g_{\mathcal{E}}\left(a_{x,t}, a_{y,t}, v_{x,t}, \mu_t; \mathcal{E}_t\right), \tag{A35}$$

where $g_{\mathcal{E}}(\cdot)$ is induced by the coupled acceleration-envelope function. For the super-ellipse envelope used in this work, the utilization can be written as

$$\eta_{\text{env},t} = \begin{cases} \left(\dfrac{a_{x,t}}{a^{\text{acc}}_{x,\max}(v_{x,t}, \mu_t)}\right)^c + \left(\dfrac{|a_{y,t}|}{a_{y,\max}(v_{x,t}, \mu_t)}\right)^c, & a_{x,t} \geq 0, \\ \left(\dfrac{|a_{x,t}|}{a^{\text{dec}}_{x,\max}(v_{x,t}, \mu_t)}\right)^c + \left(\dfrac{|a_{y,t}|}{a_{y,\max}(v_{x,t}, \mu_t)}\right)^c, & a_{x,t} < 0. \end{cases} \tag{A36}$$

A value $\eta_{\text{env},t} \leq 1$ indicates that the executed acceleration remains inside the feasible envelope, while values close to one indicate operation near the vehicle-level physical boundary.

At the tire level, slip ratio and slip angle are used to quantify tire–road force utilization. The normalized tire-limit utilization is

$$\eta_{\text{tire},t} = \min\left(\frac{1}{4}\sum_{i\in\{\text{fl,fr,rl,rr}\}} \max\left(\frac{|s_{x,i,t}|}{s_x^{\text{th}}}, \frac{|s_{y,i,t}|}{s_y^{\text{th}}}\right), 2\right), \tag{A37}$$

where $s_{x,i,t}$ denotes the slip ratio of tire i, $s_{y,i,t}$ denotes the slip angle of tire i, and s_x^{th} and s_y^{th} are the corresponding limit thresholds. The value is clipped at 2 instead of 1 to retain information about the severity of limit violation for soft-constraint penalties.

At the control level, tracking capability is quantified by lateral and speed tracking errors,

$$\eta_{\text{control},t} = \min\left(\max\left(\frac{|e_{y,t}^{\text{trk}}|}{e_y^{\text{th}}}, \frac{|e_{v,t}^{\text{trk}}|}{e_v^{\text{th}}}\right), 2\right), \tag{A38}$$

where $e_{y,t}^{\text{trk}}$ denotes the lateral tracking error, $e_{v,t}^{\text{trk}}$ denotes the speed tracking error, and e_y^{th} and e_v^{th} are the corresponding thresholds.

At the stability level, the discrepancy between actual and desired yaw rate is used to detect the onset of maneuvering instability. The desired yaw rate is approximated by

$$r_t^{\text{des}} = \frac{v_{x,t}}{L}\tan(\delta_{\text{fb},t}) \cdot \frac{1}{1 + K v_{x,t}^2}, \tag{A39}$$

where L is the wheelbase, $\delta_{\text{fb},t}$ is the steering feedback angle, and K is the speed-dependent yaw-rate gain coefficient. The stability utilization is

$$\eta_{\text{stability},t} = \min\left(\frac{|r_t - r_t^{\text{des}}|}{r^{\text{th}}}, 2\right), \tag{A40}$$

where r_t is the actual yaw rate and r^{th} is the stability threshold.

The hybrid physical-limit utilization combines the vehicle-level, tire-level, control-level, and stability-level indicators:

$$\eta_{\text{lim},t} = \max\left(\eta_{\text{tire},t}, \eta_{\text{control},t}, \eta_{\text{stability},t}\right). \tag{A41}$$

This value is used to assess whether the ego vehicle remains within the prescribed physical-limit criterion during envelope exploration, planning, and control. An expanded physical-limit envelope is accepted only if the maximum utilization over the evaluated rollout remains below the prescribed threshold. The metric $\eta_{\text{env},t}$ serves as an envelope-based representation of $\eta_{\text{lim},t}$ to quantify the proximity of the current state to the feasible motion boundary.

S3.3 Coupled Interpretation of Cognitive and Physical Indicators

The cognitive and physical indicators are interpreted jointly because the two limits are coupled in closed-loop racing. A candidate maneuver may be strategically favorable according to interaction reasoning, but it is useful only if it remains executable within the current feasible-motion envelope. Conversely, a maneuver may be physically feasible in isolation but unsafe under localization uncertainty, opponent-tracking uncertainty, or interaction-prediction error.

In this work, localization residuals, state residuals, ego-dynamics prediction errors, and interaction-prediction ADE/FDE quantify cognitive uncertainty. Acceleration-envelope utilization, tire slip utilization, tracking errors, and yaw-rate discrepancy quantify physical-limit utilization. Closed-loop outcomes such as success rate, collision rate, safe mileage, and overtaking frequency reflect the consequence of their coupling during execution.

For a trajectory candidate τ_t^k, the closed-loop interpretation can be summarized as

$$\tau_t^k \text{ is effective} \iff \begin{cases} \text{interaction prediction indicates acceptable risk,} \\ \text{decision score indicates competitive racing utility,} \\ \eta^k_{\mathrm{lim},t:t+H} \le \eta_{\mathrm{th}}, \\ \text{execution produces no collision or instability.} \end{cases} \tag{A42}$$

where $\eta^k_{\mathrm{lim},t:t+H}$ denotes the maximum physical-limit utilization along the predicted or executed horizon of candidate τ_t^k, and η_{th} is the prescribed physical-limit threshold.

This joint interpretation connects the supplementary metrics to the main results. The localization and prediction metrics support the assessment of current and future world understanding; the physical-limit metrics support the assessment of dynamic-envelope utilization and stability; and the interaction outcomes evaluate whether the agent can select and execute maneuvers that remain safe, feasible, and competitive under coupled cognitive-physical limits.

S4. Predictive World Modeling

The world model provides the predictive representation used by future-aware reasoning, physical-limit-constrained control, and closed-loop refinement. Starting from the structured racing world state w_t, it predicts three coupled components: the future ego vehicle response, the evolution of ego–opponent interaction, and the feasible-motion boundary of the ego vehicle. This section provides the implementation details of these three components.

S4.1 Ego Dynamics Prediction

The ego dynamics model predicts the short-horizon response of the ego vehicle under aggressive control inputs. It adopts a physics-constrained residual-learning structure, combining a differentiable vehicle-dynamics backbone with a learnable residual model:

$$\hat{x}_{t+1} = f_{\mathrm{dyn}}\left(x_t, u_t; \theta_{\mathrm{dyn}}\right) + f_{\mathrm{res}}\left(x_{t-h:t}, u_{t-h:t}; \theta_{\mathrm{res}}\right), \tag{A43}$$

where x_t is the ego vehicle state, u_t is the control input, $f_{\mathrm{dyn}}(\cdot)$ is the differentiable dynamics backbone, and $f_{\mathrm{res}}(\cdot)$ is the residual model using recent state and control histories.

The differentiable backbone preserves the main physical coupling among longitudinal motion, lateral motion, yaw dynamics, and wheel rotational behavior. Its parameter vector θ_{dyn} is constrained within physically plausible ranges through bounded learnable parameters. The residual model, parameterized by θ_{res}, compensates for effects that are difficult to capture analytically, including transient tire behavior, unmodeled aerodynamic variation, local surface variation, and actuator response mismatch. Model training based on next-state prediction error is described in Supplementary Section S7.

Prediction accuracy is evaluated using the channel-wise ego-dynamics prediction RMSE defined in Supplementary Section S3.1. In the experiments, the evaluated channels include longitudinal velocity, lateral velocity, yaw rate, longitudinal acceleration, and lateral acceleration, corresponding to the near-limit motion states reported in the main manuscript.

S4.2 Interaction Prediction

The interaction prediction model captures the future evolution of multi-agent racing scenarios. The input is the structured world state w_t, which contains ego vehicle state, tracked opponent states, and local track context. In a track-coordinate system, each surrounding opponent is encoded from its relative longitudinal and lateral position and speed with respect to the ego vehicle, together with a learned slot embedding indicating its relative bearing.

The ego encoder and opponent encoders produce features that a masked attention-pooling layer aggregates into a scene representation. This representation is decoded into a latent interaction intention, expressed as a lateral reference offset and a target speed, that implicitly captures racing-relevant behaviors such as attack, defense, hold, and yield. A multilayer-perceptron transition model then predicts per-agent state increments and is recursively applied to roll out future interaction evolution.

At inference time, the interaction model is recursively rolled out as part of the world model:

$$\hat{w}_{t:t+H} = W^H\left(w_t, a_{t:t+H-1}\right), \tag{A44}$$

where $W^H(\cdot)$ denotes recursive world-model prediction over horizon H, and $a_{t:t+H-1}$ denotes candidate action inputs or behavior hypotheses. The predicted world-state sequence $\hat{w}_{t:t+H}$ contains both ego-state evolution and opponent-state evolution. The interaction transition model is trained by supervised prediction of world-state evolution as described in Supplementary Section S7.

For evaluation, predicted opponent trajectories are compared with ground-truth trajectories using the ADE and FDE metrics defined in Supplementary Section S3.1. These metrics are computed in the structured track-coordinate representation and quantify whether the world model provides sufficiently accurate short-horizon interaction futures for trajectory evaluation.

S4.3 Physical-Limit Representation

The physical-limit representation models the feasible-motion boundary of the ego vehicle. At the vehicle level, this boundary is represented as a coupled longitudinal–lateral acceleration envelope. For a state indexed by k, the envelope is written as

$$\begin{cases} \left(\dfrac{a_{x,k}}{a^{\mathrm{acc}}_{x,\max}(v_{x,k},\mu_k)}\right)^c + \left(\dfrac{|a_{y,k}|}{a_{y,\max}(v_{x,k},\mu_k)}\right)^c \le 1, & a_{x,k} \ge 0, \\ \left(\dfrac{|a_{x,k}|}{a^{\mathrm{dec}}_{x,\max}(v_{x,k},\mu_k)}\right)^c + \left(\dfrac{|a_{y,k}|}{a_{y,\max}(v_{x,k},\mu_k)}\right)^c \le 1, & a_{x,k} < 0, \end{cases} \tag{A45}$$

Algorithm 1 Nominal physical-limit envelope generation

Initialize $\mathcal{P}_{\text{lim}} \leftarrow \varnothing$
for each speed v_x **do**
  $a_{\text{drag}} \leftarrow f_{\text{drag}}(v_x)$
  $a_{x,\max}^{\text{eng}} \leftarrow \text{Engine}(v_x)$
  for each direction angle θ **do**
    $R \leftarrow 0$
    while $R < R_{\max}$ **do**
      $a_x \leftarrow R\cos\theta, \quad a_y \leftarrow R\sin\theta$
      $[F_x^{\text{tire}}, F_y^{\text{tire}}, F_z^{\text{tire}}] \leftarrow \text{VehicleDynamics}(a_x - a_{\text{drag}}, a_y; \theta_{\text{dyn}})$
      for each wheel **do**
        if $\text{TireLimit}(F_x^{\text{tire}}, F_y^{\text{tire}}, F_z^{\text{tire}}; \mu_{\text{ref}}) > 0$ **then**
          $\mathcal{P}_{\text{lim}} \leftarrow \mathcal{P}_{\text{lim}} \cup P_{\text{feasible}}$
          Update $a_{x,\max}^{\text{acc}}$, $a_{x,\max}^{\text{dec}}$, and $a_{y,\max}$ at v_x from P_{feasible}
          break
        end if
      end for
      $R \leftarrow R + \Delta R$
      $P_{\text{feasible}} \leftarrow [\min(a_x, a_{x,\max}^{\text{eng}}), a_y, v_x]$
    end while
  end for
end for
for each state $(a_x, a_y, v_x) \in \mathcal{P}_{\text{lim}}$ **do**
  $c \leftarrow c_{\min}$
  while $c \le c_{\max}$ **do**
    $a_{x,\max} \leftarrow a_{x,\max}^{\text{acc}}$ if $a_x \ge 0$, otherwise $a_{x,\max}^{\text{dec}}$
    if $[a_x/a_{x,\max}(v_x)]^c + [|a_y|/a_{y,\max}(v_x)]^c < 1$ **then**
      $c(v_x) \leftarrow c - \Delta c$
      break
    end if
    $c \leftarrow c + \Delta c$
  end while
end for
Return $\mathcal{E}_0$

where $a_{x,k}$ and $a_{y,k}$ denote longitudinal and lateral accelerations, $v_{x,k}$ is the longitudinal speed, μ_k is the road-friction coefficient, and c controls the shape of the super-ellipse envelope. The functions $a_{x,\max}^{\text{acc}}(\cdot)$, $a_{x,\max}^{\text{dec}}(\cdot)$, and $a_{y,\max}(\cdot)$ describe acceleration, braking, and lateral acceleration capabilities under the current speed and friction condition.

A nominal physical-limit envelope $\mathcal{E}_0$ is generated from the identified dynamics parameters θ_{dyn}. The envelope provides a structured prior of the vehicle's feasible motion capability before online adaptation. Boundary points are obtained by radial search in the (a_x, a_y) space over the operating speed range, and the envelope parameters are calibrated from the resulting feasible boundary.

Starting from the nominal envelope $\mathcal{E}_0$, the effective physical-limit boundary is refined through world-model-based exploration and online adaptation. The effective envelope is written as

$$\mathcal{E}_t = \mathcal{E}\left(\rho_a(w_t), \rho_e; \mathcal{E}_0(\theta_{\text{dyn}})\right), \tag{A46}$$

where ρ_e is the physical-limit exploration factor and $\rho_a(w_t)$ is the online adaptive limit factor.

Physical-limit exploration aims to expand the nominal envelope when doing so improves performance without violating the prescribed limit criterion. A safe

reinforcement-learning policy proposes an envelope expansion action

$$a_k = \left[\Delta\rho_{e,x}^{\mathrm{dec}}, \Delta\rho_{e,x}^{\mathrm{acc}}, \Delta\rho_{e,y}\right] \in [0, \Delta\rho_{\max}]^3, \tag{A47}$$

where $\rho_{e,x}^{\mathrm{acc}}$, $\rho_{e,x}^{\mathrm{dec}}$, and $\rho_{e,y}$ scale acceleration, braking, and lateral limits, respectively. The policy input is

$$s_k = [\Gamma_{k-1}, \Gamma_k, \eta_{\mathrm{lim},k-1}], \tag{A48}$$

where Γ_{k-1} and Γ_k denote preceding and upcoming track segments, and $\eta_{\mathrm{lim},k-1}$ is the limit utilization of the previous segment.

The exploration reward penalizes both excessive expansion and limit violation:

$$\bar{r}_k = r_k - \beta\|a_k\|_2^2 - \lambda c_k, \tag{A49}$$

where

$$r_k = -\frac{T_k}{T_k(\Gamma_{\mathcal{E}_0}^*)}, \qquad c_k = \max(\eta_{\mathrm{lim},k} - 1, 0)^2. \tag{A50}$$

Here T_k is the lap-time-related segment objective, $\Gamma_{\mathcal{E}_0}^*$ is the reference optimal trajectory under the nominal envelope, and c_k penalizes violation of the hybrid physical-limit utilization.

The proposed expansion is first applied as

$$\tilde{\rho}_{e,k+1} = \rho_{e,k} + a_k, \tag{A51}$$

$$\tilde{\mathcal{E}}_{k+1} = \tilde{\rho}_{e,k+1} \odot \mathcal{E}_0, \tag{A52}$$

where $\odot$ denotes element-wise scaling of the nominal envelope parameters. The update is accepted only if the maximum physical-limit utilization satisfies the prescribed criterion:

$$\rho_{e,k+1} = \begin{cases} \tilde{\rho}_{e,k+1}, & \eta_{\mathrm{lim}}^{\max}(\tilde{\mathcal{E}}_{k+1}) \le 1, \\ \rho_{e,k}, & \text{otherwise.} \end{cases} \tag{A53}$$

This rule prevents unsafe over-expansion of the feasible-motion boundary.

During online execution, the envelope is further adjusted according to operating conditions. The adaptive factor is defined as

$$\rho_a(w_t) = \frac{\mu_t}{\mu_{\mathrm{ref}}}\,\psi\left(\frac{T_{\mathrm{fl}} + T_{\mathrm{fr}} + T_{\mathrm{rl}} + T_{\mathrm{rr}}}{4}\right)\exp\left(-\lambda_{\mathrm{wm}}\varepsilon_{\mathrm{ego}}^{\mathrm{wm}}\right), \tag{A54}$$

where μ_t and μ_{ref} denote the estimated and reference friction coefficients, $\psi(\cdot)$ is the tire-temperature attenuation function, and $\varepsilon_{\mathrm{ego}}^{\mathrm{wm}}$ is the world-model prediction error for the ego state.

The implemented online envelope is

$$\mathcal{E}_t = \rho_a(w_t)\rho_e \odot \mathcal{E}_0(\theta_{\mathrm{dyn}}). \tag{A55}$$

Thus, the physical-limit representation evolves through two mechanisms: offline exploration enlarges the nominal feasible-motion boundary when supported by world-model

rollouts, while online adaptation contracts or scales the envelope according to friction, tire temperature, and prediction uncertainty.

S5. Future-Aware Reasoning and Trajectory Evaluation

Future-aware reasoning transforms the current racing world state into a selected trajectory that is safe, feasible, and competitive before execution. The module combines a high-level decision policy, a multimodal delta planner, physical-limit-based speed replanning, and world-model-based predictive evaluation. The resulting trajectory is passed to the physical-limit-constrained controller described in Supplementary Section S6.

S5.1 Decision Policy Network

At each time step, the decision policy receives the structured world state w_t, including the ego vehicle state, tracked opponent states, and local track context. The policy outputs a compact racing command

$$\xi_d = [v_d, l_d], \tag{A56}$$

where v_d denotes the target speed and l_d denotes the target lateral position or offset in the track-coordinate system. This command represents the main racing intention, such as accelerating, braking, holding line, defending, or preparing an overtaking maneuver.

The decision policy uses a structured interaction encoder. The ego vehicle is encoded using Frenet-frame states, while opponent vehicles are organized into semantic slots according to their relative longitudinal and lateral positions. Each opponent slot contains relative position, lateral offset, velocity, availability, and slot identity. Slot-wise interaction features are aggregated into a compact scene representation and combined with the ego feature. The policy head outputs the longitudinal–lateral racing command ξ_d. During training, stochastic action sampling is used for exploration; during deployment, deterministic inference is used for stable online decision-making.

Given the current ego state x_t and the decision command ξ_d, a nominal reference trajectory is generated over a finite horizon H:

$$\tau_t^{\mathrm{ref}} = f_{\mathrm{tran}}(x_t, \xi_d, H), \tag{A57}$$

where $f_{\mathrm{tran}}(\cdot)$ denotes the transition model that converts the high-level racing command into a reference trajectory. This trajectory represents the main decision branch implied by the current racing intention.

The multimodal planning framework including the decision network and the planning network is shown in Fig. A7. The decision network generates the decision-conditioned backbone trajectory, and the planning network produces the delta trajectory manifold for multimodal trajectory generation.

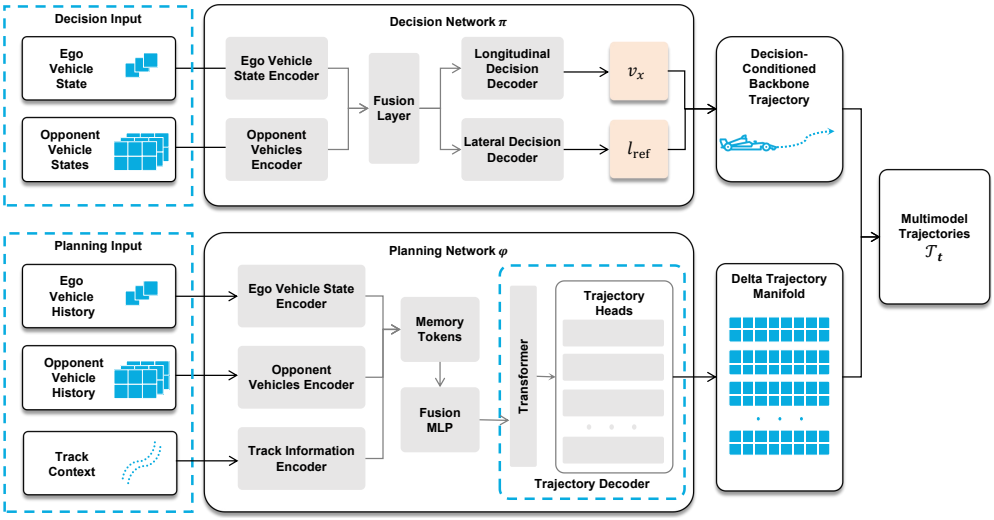


Fig. A7 The architecture of the multimodal planning networks.

S5.2 Multimodal Delta Planner

Competitive racing interactions are multimodal: the same observed world state may admit several plausible maneuvers, such as left-side overtaking, right-side overtaking, braking behind an opponent, or holding the current line. To represent this multimodality, a delta planner generates mode-specific trajectory deviations around the main reference trajectory.

Given recent world-state history $w_{t-h:t}$, the planner outputs K candidate trajectory offsets,

$$\Delta\tau_t^k = \phi^k(w_{t-h:t};\theta_\phi), \qquad k = 1,\dots,K, \tag{A58}$$

where $\phi^k(\cdot)$ denotes the k-th output mode of the multimodal planner and θ_ϕ denotes the planner parameters. Candidate trajectories are then formed as

$$\tau_t^k = \tau_t^{\text{ref}} + \Delta\tau_t^k. \tag{A59}$$

To diversify longitudinal behavior, a set of speed factors are prescribed. The multimodal trajectories are scaled with speed factors, producing candidate trajectories for a subsequent world-model rollout evaluation.

The planner encodes ego history, opponent history, and local track context using transformer-based modules. The resulting latent representation is decoded through multiple output heads, allowing different modes to specialize in different racing behaviors. This multimodal structure prevents the reasoning process from committing prematurely to a single maneuver before future interaction outcomes are evaluated. Planning results in diverse racing scenes are shown in Fig. A8. The deeper blue center trajectory is the decision-conditioned backbone trajectory. The light blue trajectories are the multimodal trajectories.

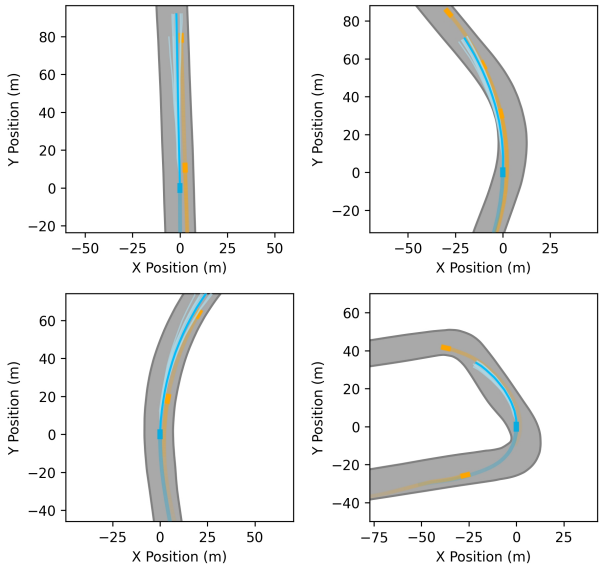


Fig. A8 Planning results in diverse racing scenes.

S5.3 Speed Replanning under Physical-Limit Constraints

The prescribed speed factors diversify the longitudinal profiles of the multimodal trajectories, while the following speed-replanning step enforces compatibility with the current physical-limit envelope.

The candidate trajectories are further regularized by the current feasible-motion envelope $\mathcal{E}_t$. This step converts each candidate into an executable trajectory with a speed profile compatible with the physical capability of the ego vehicle:

$$\mathcal{T}_t = \left\{ R(\tau_t^k, \mathcal{E}_t) \right\}_{k=1}^{K}, \tag{A60}$$

where $R(\tau_t^k, \mathcal{E}_t)$ denotes speed replanning under the adaptive physical-limit envelope. The replanning process respects the coupled longitudinal–lateral acceleration constraint induced by $\mathcal{E}_t$.

This replanning step provides a direct connection between physical-limit representation and future-aware reasoning. A candidate trajectory is not evaluated solely by its geometric path or racing intention; it is first converted into a physically compatible trajectory under the current envelope $\mathcal{E}_t$.

S5.4 Parallel World-Model Rollouts

After candidate generation and physical-limit regularization, each candidate trajectory is evaluated through recursive world-model rollouts. Starting from the current world state w_t, the world model predicts the future evolution induced by the candidate:

$$\hat{w}_{t:t+H}^k = W^H(w_t, a_{t:t+H-1}^k), \tag{A61}$$

Algorithm 2 Speed replanning under the feasible-motion envelope

Require: Candidate trajectory τ_t^k, current longitudinal speed v_x^0, feasible-motion envelope $\mathcal{E}_t$

Extract arc-length increments Δs_{t+h}^k and curvature κ_{t+h}^k from τ_t^k

$v_{t+h}^k \leftarrow \min\Big(\sqrt{a_{y,\max}(v_x^0, \mathcal{E}_t)/\max(|\kappa_{t+h}^k|, \epsilon)},\, v_{\max}\Big)$

Set $v_t^k \leftarrow v_x^0$

for $h = 0$ to $H-1$ **do**

  $a_{y,t+h} \leftarrow (v_{t+h}^k)^2 \kappa_{t+h}^k$

  $a_{x,t+h}^+ \leftarrow \mathcal{E}_t(a_{y,t+h}, v_{t+h}^k)$

  $v_{t+h+1}^k \leftarrow \min\Big(v_{t+h+1}^k, \sqrt{\max((v_{t+h}^k)^2 + 2a_{x,t+h}^+ \Delta s_{t+h}^k, 0)}\Big)$

end for

for $h = H$ down to 1 **do**

  $a_{y,t+h} \leftarrow (v_{t+h}^k)^2 \kappa_{t+h}^k$

  $a_{x,t+h}^- \leftarrow \mathcal{E}_t(a_{y,t+h}, v_{t+h}^k)$

  $v_{t+h-1}^k \leftarrow \min\Big(v_{t+h-1}^k, \sqrt{\max((v_{t+h}^k)^2 + 2a_{x,t+h}^- \Delta s_{t+h}^k, 0)}\Big)$

end for

$a_{x,t+h}^k \leftarrow \big((v_{t+h+1}^k)^2 - (v_{t+h}^k)^2\big)/(2\Delta s_{t+h}^k + \epsilon)$

Update τ_t^k with $\{v_{t+h}^k, a_{x,t+h}^k\}$

Densify and resample τ_t^k at fixed interval Δt

where $a_{t:t+H-1}^k$ denotes the behavior associated with candidate trajectory τ_t^k. The resulting set of parallel futures is

$$\mathcal{F}_t = \left\{ \left(\tau_t^k, \hat{w}_{t:t+H}^k\right) \mid \tau_t^k \in \mathcal{T}_t \right\}. \tag{A62}$$

Each predicted future contains ego-motion evolution, opponent-motion evolution, and feasibility information under the current physical-limit representation. This allows the agent to assess whether a maneuver that appears competitive at the current instant remains safe and executable over the prediction horizon.

S5.5 Safety, Feasibility, and Speed Scoring

Each candidate future is evaluated by a task-oriented score that combines safety, feasibility, and speed efficiency. The total score is written as

$$S^k = S_{\text{safety}}^k + S_{\text{speed}}^k - S_{\text{violation}}^k, \tag{A63}$$

where S_{safety}^k penalizes predicted collision or boundary risk, S_{speed}^k rewards racing progress and speed efficiency, and $S_{\text{violation}}^k$ penalizes predicted physical-limit violation or loss of control.

The track-boundary risk is defined as

$$\rho_{\text{track}}^k = \max_{h=1,\ldots,H} \max\left[\left(l_{t+h}^k - l_L(s_{t+h}^k)\right)_+, \left(l_R(s_{t+h}^k) - l_{t+h}^k\right)_+ \right], \tag{A64}$$

where $(x)_+ = \max(0, x)$, $l_L(\cdot)$ and $l_R(\cdot)$ are the safe left and right track boundaries, and (s, l) denotes the track-coordinate position.

The opponent-collision risk is

$$\rho^k_{\text{opponent}} = \max_{j\in\mathcal{V}_t,\, h\in\{1,\dots,H\}} \left[\left(d^{\text{th}}_s - \left| s^k_{t+h} - \hat{s}^j_{t+h} \right| \right)_+ \left(d^{\text{th}}_l - \left| l^k_{t+h} - \hat{l}^j_{t+h} \right| \right)_+ \right], \quad \text{(A65)}$$

where $\mathcal{V}_t$ is the set of valid opponent vehicles, $(\hat{s}^j_{t+h}, \hat{l}^j_{t+h})$ is the predicted track-coordinate position of opponent j, and d^{th}_s and d^{th}_l are longitudinal and lateral safety thresholds.

The safety score is

$$S^k_{\text{safety}} = \exp\left(-\rho^k_{\text{opponent}} - \rho^k_{\text{track}} - \lambda_r \max_{h=1,\dots,H} |r^k_{t+h}| \right), \quad \text{(A66)}$$

where r^k_{t+h} is the yaw rate along candidate trajectory k, and λ_r weights yaw-rate-related instability risk.

Physical feasibility is evaluated using the hybrid physical-limit utilization defined in Supplementary Section S3:

$$\eta^k_{\text{lim},t:t+H} = \max_{h=1,\dots,H} \eta^k_{\text{lim},t+h}. \quad \text{(A67)}$$

The physical-limit violation penalty is

$$S^k_{\text{violation}} = \lambda_\eta \left(\max\left(\eta^k_{\text{lim},t:t+H} - \eta_{\text{th}}, 0 \right) \right)^2, \quad \text{(A68)}$$

where η_{th} is the prescribed limit threshold and λ_η is a penalty weight.

The speed score is defined by the terminal speed ratio,

$$S^k_{\text{speed}} = \frac{v^k_{x,t+H}}{v^{\text{ref}}_x(s^k_{t+H}; \mathcal{E}_0)}, \quad \text{(A69)}$$

where $v^{\text{ref}}_x(s^k_{t+H}; \mathcal{E}_0)$ is the reference speed at terminal progress s^k_{t+H} under the nominal envelope.

The final selected trajectory is

$$\tau^*_t = \arg\max_{\tau^k_t \in \mathcal{T}_t} S^k. \quad \text{(A70)}$$

This selection rule favors trajectories that are interaction-safe, physically feasible, and competitive in speed before they are committed to the physical vehicle.

The resulting interaction cases and the interaction statistics of the scenarios are shown in Fig. A9.

S6. Physical-Limit-Constrained Motion Control

The physical-limit-constrained controller converts the trajectory selected by future-aware reasoning into executable control commands. Given the selected trajectory

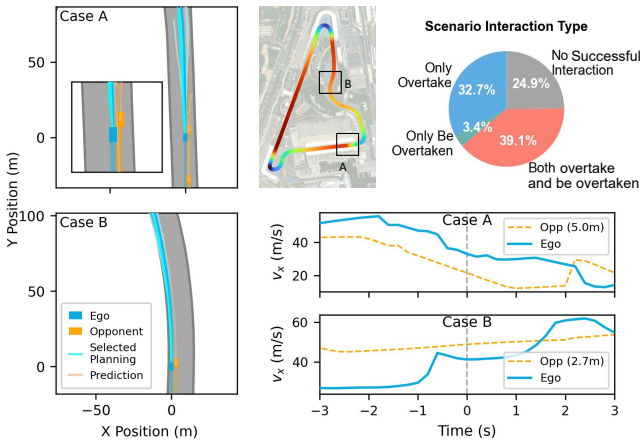


Fig. A9 Interaction cases and scenario interaction statistics.

τ_t^* and the world-model-informed feasible-motion envelope $\mathcal{E}_t$, the controller tracks the reference trajectory while enforcing vehicle dynamics, road-boundary constraints, actuator limits, and coupled longitudinal–lateral physical-limit constraints. This section provides the vehicle dynamics, linearization, barrier derivatives, and constrained iLQR implementation used by the controller. The architecture of the motion controller is illustrated in Fig. A10.

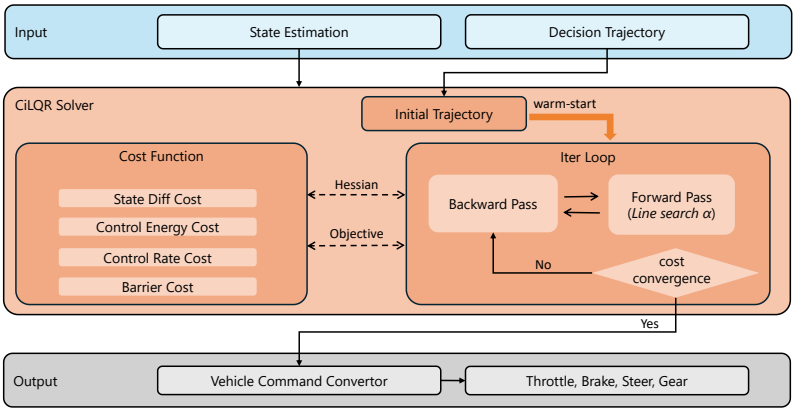


Fig. A10 Physical-limit-constrained CiLQR control framework.

S6.1 Control-Oriented Vehicle Dynamics

The control-oriented vehicle model uses the state

$$x = [X,\ Y,\ \psi,\ v_x,\ v_y,\ r]^\top, \tag{A71}$$

where (X, Y) denotes the position in the global frame, ψ is the heading angle, v_x and v_y are the longitudinal and lateral velocities in the body frame, and $r = \dot{\psi}$ is the yaw rate. The control input is

$$u = [a_x,\ \delta]^\top, \tag{A72}$$

where a_x is the commanded longitudinal acceleration and δ is the front-wheel steering angle.

The continuous-time vehicle dynamics are

$$\begin{cases} \dot{X} = v_x \cos\psi - v_y \sin\psi, & \dot{v}_x = a_x + v_y r, \\ \dot{Y} = v_x \sin\psi + v_y \cos\psi, & \dot{v}_y = \dfrac{F_{yf}\cos\delta + F_{yr}}{m} - v_x r, \\ \dot{\psi} = r, & \dot{r} = \dfrac{l_f F_{yf}\cos\delta - l_r F_{yr}}{I_z}. \end{cases} \tag{A73}$$

Here m is the vehicle mass, I_z is the yaw moment of inertia, l_f and l_r are the distances from the center of gravity to the front and rear axles, and F_{yf} and F_{yr} are the front and rear lateral tire forces.

The discrete-time transition is obtained by Euler integration:

$$x_{k+1} = x_k + f_{\text{dyn}}(x_k, u_k; \theta_{\text{dyn}})\Delta t, \tag{A74}$$

where $f_{\text{dyn}}(\cdot)$ denotes the control-oriented dynamics model and θ_{dyn} contains the physical parameters calibrated from the dynamics component of the world model.

S6.2 Dynamics Linearization and Jacobians

The iLQR backward pass requires local linearization of the discrete dynamics. Let $f_c(x, u)$ denote the continuous-time dynamics in Eq. (A73). The state Jacobian is

$$\frac{\partial f_c}{\partial x} = \begin{bmatrix} 0 & 0 & f_{1,3} & \cos\psi & -\sin\psi & 0 \\ 0 & 0 & f_{2,3} & \sin\psi & \cos\psi & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & r & v_y \\ 0 & 0 & 0 & f_{5,4} & f_{5,5} & f_{5,6} \\ 0 & 0 & 0 & f_{6,4} & f_{6,5} & f_{6,6} \end{bmatrix}, \tag{A75}$$

where

$$f_{1,3} = -v_x \sin\psi - v_y \cos\psi, \qquad f_{2,3} = v_x \cos\psi - v_y \sin\psi. \tag{A76}$$

The lateral and yaw dynamics terms are

$$f_{5,4} = \frac{\cos\delta}{m}\frac{\partial F_{yf}}{\partial v_x} + \frac{1}{m}\frac{\partial F_{yr}}{\partial v_x} - r, \tag{A77}$$

$$f_{5,5} = \frac{\cos\delta}{m}\frac{\partial F_{yf}}{\partial v_y} + \frac{1}{m}\frac{\partial F_{yr}}{\partial v_y}, \tag{A78}$$

$$f_{5,6} = \frac{\cos\delta}{m}\frac{\partial F_{yf}}{\partial r} + \frac{1}{m}\frac{\partial F_{yr}}{\partial r} - v_x, \tag{A79}$$

$$f_{6,4} = \frac{l_f \cos\delta}{I_z}\frac{\partial F_{yf}}{\partial v_x} - \frac{l_r}{I_z}\frac{\partial F_{yr}}{\partial v_x}, \tag{A80}$$

$$f_{6,5} = \frac{l_f \cos\delta}{I_z}\frac{\partial F_{yf}}{\partial v_y} - \frac{l_r}{I_z}\frac{\partial F_{yr}}{\partial v_y}, \tag{A81}$$

$$f_{6,6} = \frac{l_f \cos\delta}{I_z}\frac{\partial F_{yf}}{\partial r} - \frac{l_r}{I_z}\frac{\partial F_{yr}}{\partial r}. \tag{A82}$$

The control Jacobian is

$$\frac{\partial f_c}{\partial u} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & g_{5,2} & g_{6,2} \end{bmatrix}^{\mathsf{T}}, \tag{A83}$$

where

$$g_{5,2} = \frac{1}{m}\left(\cos\delta\frac{\partial F_{yf}}{\partial \delta} - \sin\delta F_{yf}\right), \tag{A84}$$

$$g_{6,2} = \frac{l_f}{I_z}\left(\cos\delta\frac{\partial F_{yf}}{\partial \delta} - \sin\delta F_{yf}\right). \tag{A85}$$

For the linear tire-force approximation used in the controller,

$$\frac{\partial F_{yf}}{\partial \delta} = -C_f, \tag{A86}$$

so that

$$g_{5,2} = \frac{1}{m}\left(-C_f\cos\delta - \sin\delta F_{yf}\right), \tag{A87}$$

$$g_{6,2} = \frac{l_f}{I_z}\left(-C_f\cos\delta - \sin\delta F_{yf}\right). \tag{A88}$$

The discrete-time Jacobians used by iLQR are obtained from the Euler-discretized model:

$$A_k = I + \Delta t\frac{\partial f_c}{\partial x}\bigg|_{x_k,u_k}, \qquad B_k = \Delta t\frac{\partial f_c}{\partial u}\bigg|_{x_k,u_k}. \tag{A89}$$

S6.3 Tracking, Smoothness, and Constraint Costs

Given the selected trajectory τ_t^*, the controller solves the finite-horizon problem

$$u_t^* = \arg\min_{u_{t:t+H_c-1}} J(x_t, u_{t:t+H_c-1}; \tau_t^*, \mathcal{E}_t), \tag{A90}$$

subject to vehicle dynamics and inequality constraints. The cost is decomposed as

$$J = J_{\text{track}} + J_{\text{barrier}}. \tag{A91}$$

The tracking and smoothness cost is

$$J_{\text{track}} = \sum_{j=0}^{H_c-1} \left(\|x_{t+j} - x_{t+j}^{\text{ref}}\|_Q^2 + \|u_{t+j}\|_R^2 + \|\Delta u_{t+j}\|_P^2 \right), \tag{A92}$$

where x_{t+j}^{ref} is induced by τ_t^*, and Q, R, and P are weighting matrices for tracking error, control effort, and control smoothness.

Inequality constraints are imposed through exponential barrier functions:

$$J_{\text{barrier}} = \sum_{k,i} q_1^{(i)} \exp\left(q_2^{(i)} c_i(x_k, u_{k-1}; \mathcal{E}_k) \right), \tag{A93}$$

where $c_i \leq 0$ indicates satisfaction of the corresponding constraint and $c_i > 0$ indicates violation. The constraints include road boundaries, velocity limits, lateral velocity limits, yaw-rate limits, input limits, and the physical-limit envelope.

S6.4 Physical-Limit Barrier Terms

The physical-limit constraint is induced by the coupled acceleration envelope $\mathcal{E}_k$. In the simplified quadratic form used for barrier differentiation, the constraint is expressed as

$$c_{\text{lim}} = \frac{a_x^2}{a_{x,\max}^2} + \frac{a_y^2}{a_{y,\max}^2} - 1, \tag{A94}$$

where $a_{x,\max}$ and $a_{y,\max}$ are selected from the current feasible-motion envelope according to speed, road condition, and acceleration or braking regime. A constraint violation corresponds to $c_{\text{lim}} > 0$.

The exponential barrier for the physical-limit constraint is

$$B(c_{\text{lim}}) = q_1^{\text{lim}} \exp\left(q_2^{\text{lim}} c_{\text{lim}} \right). \tag{A95}$$

For a generic variable vector z, its gradient and Gauss–Newton Hessian approximation are

$$\nabla_z B = q_2^{\text{lim}} B(c_{\text{lim}}) \nabla_z c_{\text{lim}}, \tag{A96}$$

$$\nabla_z^2 B \approx (q_2^{\text{lim}})^2 B(c_{\text{lim}}) (\nabla_z c_{\text{lim}}) (\nabla_z c_{\text{lim}})^\top. \tag{A97}$$

The corresponding control gradient and Hessian contributions are

$$l_u^{\text{lim}} = q_2^{\text{lim}} B(c_{\text{lim}}) \begin{bmatrix} 2a_x / a_{x,\max}^2 \\ 0 \end{bmatrix}, \tag{A98}$$

$$l_{uu}^{\text{lim}} = (q_2^{\text{lim}})^2 B(c_{\text{lim}}) \begin{bmatrix} 4a_x^2 / a_{x,\max}^4 & 0 \\ 0 & 0 \end{bmatrix}. \tag{A99}$$

Using the approximation $a_y \approx v_x r$, the corresponding state gradient contribution is

$$l_x^{\lim} = q_2^{\lim} B(c_{\lim}) \begin{bmatrix} 0 \\ 0 \\ 0 \\ 2v_x r^2 / a_{y,\max}^2 \\ 0 \\ 2v_x^2 r / a_{y,\max}^2 \end{bmatrix}, \tag{A100}$$

and the Gauss–Newton state Hessian contribution is

$$l_{xx}^{\lim} = (q_2^{\lim})^2 B(c_{\lim}) \frac{\partial c_{\lim}}{\partial x} \left(\frac{\partial c_{\lim}}{\partial x} \right)^{\top}. \tag{A101}$$

These derivatives are accumulated with the nominal stage-cost derivatives and other box-constraint barrier terms before the iLQR backward pass constructs the local quadratic model.

S6.5 Box Constraints and Numerical Safeguards

The box constraints are summarized in Table A1. Each constraint c_i is defined such that $c_i > 0$ indicates violation.

Table A1 Box constraints and non-zero derivatives used by the constrained controller.

Constraint	c_i	Non-zero derivative
Acceleration upper	$a_x - a_{\max}$	$\partial c_i / \partial a_x = +1$
Acceleration lower	$a_{\min} - a_x$	$\partial c_i / \partial a_x = -1$
Steering upper	$\delta - \delta_{\max}$	$\partial c_i / \partial \delta = +1$
Steering lower	$-\delta_{\max} - \delta$	$\partial c_i / \partial \delta = -1$
Velocity upper	$v - v_{\max}$	$\partial c_i / \partial v = +1$
Velocity lower	$v_{\min} - v$	$\partial c_i / \partial v = -1$
Lateral upper	$d_k - d_{\max}$	$\partial c_i / \partial p = \hat{n}$
Lateral lower	$d_{\min} - d_k$	$\partial c_i / \partial p = -\hat{n}$

Here d_k denotes the signed lateral distance to the reference line, $\hat{n}$ is the unit normal at the nearest reference point, and $d_{\max}$ and $d_{\min}$ are derived from road width and vehicle width. Each barrier is parameterized by a pair $(q_1^{(i)}, q_2^{(i)})$.

Several numerical safeguards are applied. First, for $|v_x| < v_{\text{thr}}$, the physical-limit constraint is set to $c_{\lim} = -1$ and its derivatives are set to zero. Second, the acceleration envelope switches between traction and braking bounds according to the sign of a_x. Third, barrier values are monitored to avoid numerical overflow in the exponential function. Fourth, the final control sequence is hard-clamped to box limits after the forward rollout.

S6.6 CiLQR Solver

The constrained iLQR solver minimizes the cost in Eq. (A91) over a finite horizon by iteratively linearizing the dynamics and quadratizing the cost. Starting from the terminal time step, the backward pass computes

$$Q_x = l_x + A^\top V_x, \tag{A102}$$

$$Q_u = l_u + B^\top V_x, \tag{A103}$$
$$Q_{xx} = l_{xx} + A^\top V_{xx} A, \tag{A104}$$
$$Q_{uu} = l_{uu} + B^\top V_{xx} B, \tag{A105}$$
$$Q_{ux} = l_{ux} + B^\top V_{xx} A. \tag{A106}$$

The feedforward correction and feedback gain are

$$d_k = -(Q_{uu} + \lambda I)^{-1} Q_u, \tag{A107}$$

$$K_k = -(Q_{uu} + \lambda I)^{-1} Q_{ux}, \tag{A108}$$

where $\lambda > 0$ is a Tikhonov regularization parameter.

The forward rollout applies the affine feedback update

$$u_k^{\text{new}} = u_k + \alpha d_k + \gamma K_k (x_k^{\text{new}} - x_k), \tag{A109}$$

where $\alpha \in (0, 1]$ is a line-search parameter and $\gamma \in [0, 1]$ scales the feedback term. If the rollout reduces the objective, the update is accepted and the regularization is relaxed. Otherwise, the regularization is increased to produce a more conservative update.

Algorithm 3 summarizes the constrained CiLQR implementation.

The physical-limit terms in Eqs. (A98)–(A101) are evaluated at each time step and accumulated into the stage-cost derivatives during the backward pass. This allows the controller to anticipate violations of the acceleration envelope before they appear in the executed vehicle response.

The resulting closed-loop control profiles are illustrated in Fig. A11, showing how the controller coordinates actuation and vehicle response along the racing lap.

The closed-loop state and control variable distribution are further summarized in Fig. A12.

S7. Closed-Loop World-Model and Policy Refinement

The autonomous racing agent is refined through a closed-loop pipeline that connects real-world deployment, world-model learning, internal simulation, and policy update. Rather than updating the decision policy alone, the framework refines the predictive models of ego dynamics, interaction evolution, and physical feasibility, and then uses the refined world model to generate structured rollout data for improving future-aware reasoning and planning.

Algorithm 3 Constrained CiLQR with physical-limit envelope

Require: Initial state x_0, selected reference trajectory τ_t^*, feasible-motion envelope $\mathcal{E}_t$
Initialize controls $u_{0:N-1}$ using warm start or reference-based initialization
Roll out dynamics to obtain $x_{0:N}$
$J \leftarrow \textsc{TotalCost}(x_{0:N}, u_{0:N-1}; \tau_t^*, \mathcal{E}_t)$
$\lambda \leftarrow \lambda_{\text{init}}$
for iter = 1 to max_iter **do**
  for $k = N$ down to 1 **do**
    Linearize dynamics to obtain A_k, B_k
    Compute stage-cost and barrier derivatives
    Form $Q_x, Q_u, Q_{xx}, Q_{uu}, Q_{ux}$
    Compute d_k, K_k
  end for
  for $\alpha \in \{1.0, 0.5, 0.25, \ldots\}$ **do**
    Roll out updated controls using affine feedback
    $J^+ \leftarrow \textsc{TotalCost}(x^+, u^+; \tau_t^*, \mathcal{E}_t)$
    if $J^+ < J$ **then**
      $x, u, J \leftarrow x^+, u^+, J^+$
      $\lambda \leftarrow \lambda \cdot \lambda_{\text{decay}}$
      break
    end if
  end for
  if no line-search step reduces cost **then**
    $\lambda \leftarrow \lambda \cdot \lambda_{\text{amp}}$
    if $\lambda > \lambda_{\max}$ **then**
      break
    end if
  end if
  if relative cost improvement $< \epsilon$ **then**
    break
  end if
end for
Hard-clamp final controls to actuator box limits
Return Optimized control sequence $u_{0:N-1}$ and predicted state sequence $x_{0:N}$

S7.1 Real Racing Dataset

During full-scale autonomous racing, the agent collects data under real road, tire, vehicle, and interaction conditions. The resulting real-racing dataset is denoted as

$$\mathcal{D}_{\text{real}} = \{(w_{t-h:t},\, a_t,\, w_{t+1:t+H}; \Gamma)\}_{t=1}^{N}. \tag{A110}$$

where $w_{t-h:t}$ denotes the historical structured racing world states, a_t denotes the executed agent action, $w_{t+1:t+H}$ denotes the future world-state sequence, and Γ denotes the global track context. Here, a_t is a composite action variable that contains the reasoning output, the motion-control command u_t, and the actuator-level command recorded by the onboard system. The global track reference Γ provides track-level information, including the racing line, track-coordinate map, road boundaries, and curvature profile, while the local track context used at each time step is included in the structured world state w_t.

These data provide the empirical basis for aligning the world model with the deployed system. In particular, $\mathcal{D}_{\text{real}}$ is used to update ego dynamics prediction, interaction prediction, and the physical-limit representation. The physical-limit envelope is initialized conservatively and then refined through the exploration and adaptation process described in Supplementary Section S4.

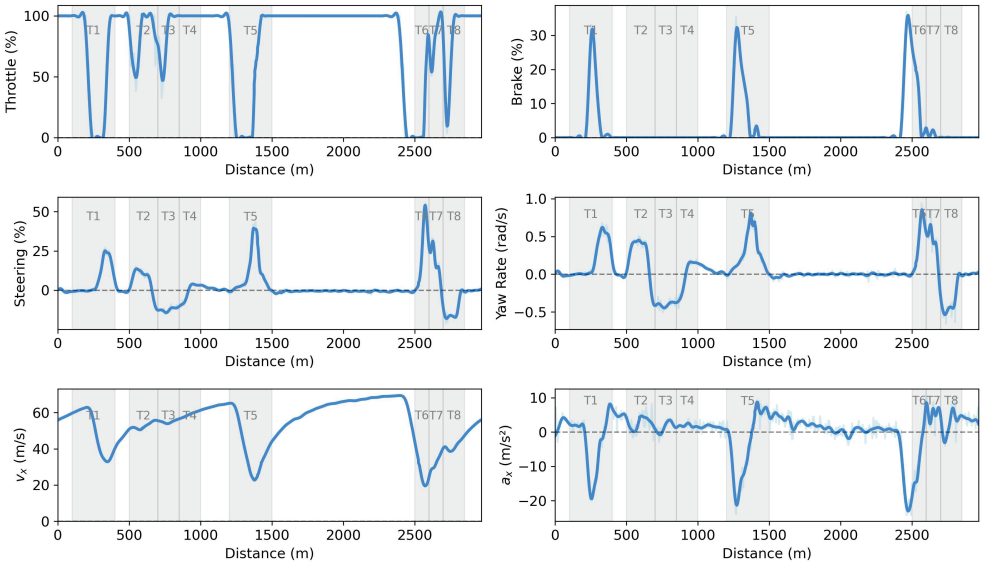


Fig. A11 Representative full-lap control response.

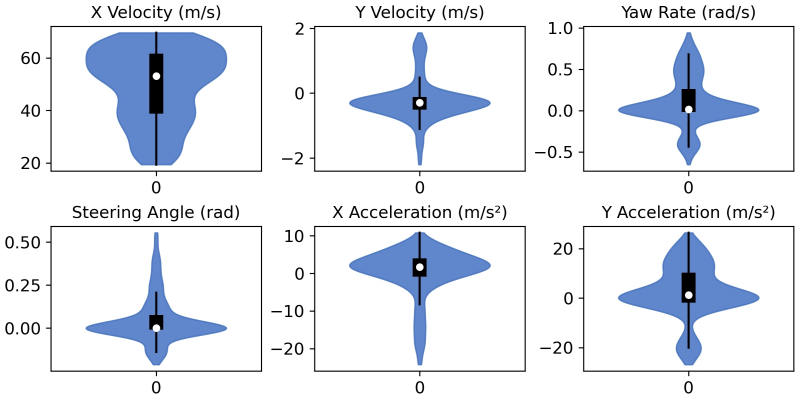


Fig. A12 Full-lap distributions of closed-loop vehicle states and control variables.

S7.2 World-Model Training

The ego dynamics model is trained using supervised next-state prediction. The differentiable dynamics backbone and residual model are updated by minimizing

$$(\theta_{\text{dyn}}, \theta_{\text{res}}) = \arg\min_{\theta_{\text{dyn}},\theta_{\text{res}}} \sum_{\mathcal{D}_{\text{real}}} \left\| x_{t+1} - f_{\text{dyn}}\left(x_t, u_t; \theta_{\text{dyn}}\right) - f_{\text{res}}\left(x_{t-h:t}, u_{t-h:t}; \theta_{\text{res}}\right) \right\|^2 . \tag{A111}$$

Here $f_{\text{dyn}}(\cdot)$ represents the differentiable dynamics backbone and $f_{\text{res}}(\cdot)$ represents the residual model.

The interaction model is updated by supervised prediction of world-state evolution,

$$\theta_{\text{int}} = \arg\min_{\theta_{\text{int}}} \sum_{\mathcal{D}_{\text{real}}} \left\| w_{t+1} - W(w_t; \theta_{\text{int}}) \right\|^2 , \tag{A112}$$

where $W(\cdot; \theta_{\text{int}})$ denotes the learned interaction transition model. This update aligns the world model with observed ego–opponent interaction patterns in real racing.

The physical-limit representation is refined using the same real racing data and world-model rollouts. The nominal envelope $\mathcal{E}_0$ is identified from the dynamics model, while the effective envelope $\mathcal{E}_t$ is adjusted through the exploration and adaptation factors described in Supplementary Section S4. This ensures that the physical-limit representation used by planning and control remains consistent with observed tire, road, and vehicle behavior.

S7.3 World-Model Rollout Data Generation

After training, the refined world model is used as a structured predictor for generating additional interaction and near-limit data. Given the current world model W, decision policy π, multimodal planner ϕ, and feasible-motion envelope $\mathcal{E}$, world-model-generated data are produced by

$$\mathcal{D}_{\text{wm}} = \text{Rollout}\left(W, \pi, \phi, \mathcal{E}\right). \tag{A113}$$

Each rollout contains predicted world-state transitions, candidate decisions, planned trajectories, opponent responses, and physical-limit indicators.

Compared with real racing data alone, $\mathcal{D}_{\text{wm}}$ can be generated under controlled variations of initial condition, opponent behavior, track segment, and physical-limit condition. This enables the training set to include rare adversarial interactions, near-limit maneuvers, and corner cases that are difficult or unsafe to collect exhaustively in real-world racing. The world model rollout interaction distribution is summarized in Fig. A13.

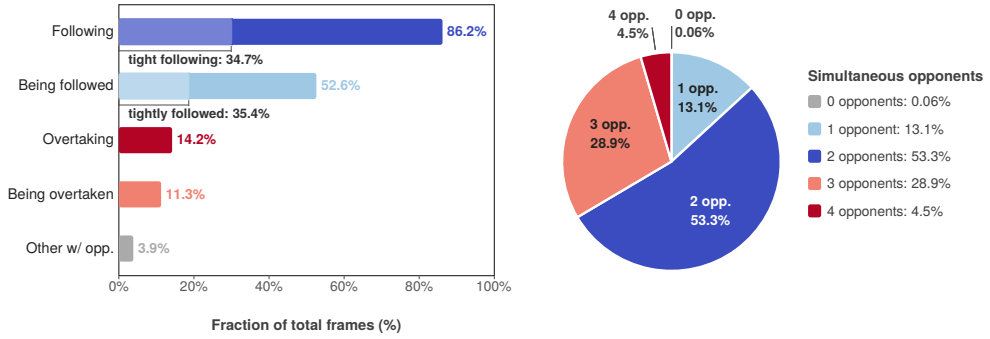


Fig. A13 World model rollout interaction distribution.

The real racing dataset and the world-model-generated dataset are combined as

$$\mathcal{D} = \mathcal{D}_{\text{real}} \cup \mathcal{D}_{\text{wm}}, \tag{A114}$$

where $\mathcal{D}$ is the hybrid dataset used for decision policy and planner update.

S7.4 Decision Policy Update

The decision policy is updated using a soft actor–critic objective with conservative Q-learning and behavior-cloning regularization. Let s_t denote the policy state corresponding to the racing world state w_t. The policy samples a racing intention

$$a_t^{\pi} \sim \pi_{\theta_\pi}(\cdot|s_t), \tag{A115}$$

where a_t^{π} contains longitudinal and lateral decision components.

The critic parameters are optimized by

$$\begin{aligned} L_Q(\theta_Q) =& \mathbb{E}_{(s_t,a_t,s_{t+1})\sim\mathcal{D}} \left[\left(Q_{\theta_Q}(s_t,a_t) - y_t \right)^2 \right] \\ &+ \lambda_{\text{cql}} \left\{ \mathbb{E}_{s_t\sim\mathcal{D}} \left[Q_{\theta_Q}(s_t,a_t^{\pi}) \right] - \mathbb{E}_{(s_t,a_t)\sim\mathcal{D}} \left[Q_{\theta_Q}(s_t,a_t) \right] \right\}. \end{aligned} \tag{A116}$$

The first term is the temporal-difference critic loss, and the conservative term discourages overestimation for actions outside the data distribution.

The policy network is optimized by

$$\begin{aligned} L_\pi(\theta_\pi) = \mathbb{E}_{(s_t,a_t^{\text{e}})\sim\mathcal{D}} \Big[& \alpha \log \pi_{\theta_\pi}(a_t^{\pi}|s_t) - \min(Q_1,Q_2) \\ & + \lambda_{\text{bc}} \left\| a_t^{\pi} - a_t^{\text{e}} \right\|^2 \Big], \end{aligned} \tag{A117}$$

where a_t^{e} denotes the demonstration or reference action in the dataset. The entropy term encourages exploration during training, the critic term improves competitive decision quality, and the behavior-cloning term preserves useful behavior observed in real and world-model-generated data.

S7.5 Multimodal Planner Update

The multimodal planner is updated using best-of-K supervision. The hybrid dataset is reorganized into pairs of recent world-state history and future trajectory:

$$\left(w_{t-h:t}, \tau_t^{\text{gt}} \right) \in \mathcal{D}, \tag{A118}$$

where $w_{t-h:t}$ is the recent history segment and τ_t^{gt} is the ground-truth or rollout-generated future trajectory.

Given the history $w_{t-h:t}$, the planner generates a set of candidate trajectories $\mathcal{T}_t$ as described in Supplementary Section S5. Only the output mode closest to the target

future trajectory is supervised:

$$\theta_\phi = \arg\min_{\theta_\phi} \sum_{\mathcal{D}} \left[\min_{\tau_t^k \in \mathcal{T}_t} \left\| \tau_t^{\mathrm{gt}} - \tau_t^k \right\|^2 \right]. \tag{A119}$$

This best-of-K objective allows different output heads to specialize in different behavioral modes, such as overtaking, defending, yielding, braking, or holding the current line.

S7.6 Closed-Loop Refinement Algorithm

Algorithm 4 summarizes the closed-loop refinement pipeline.

Algorithm 4 Closed-loop world-model and policy refinement

Require: Initial world model W, decision policy π, multimodal planner ϕ, feasible-motion envelope $\mathcal{E}$
for each refinement round **do**
  Deploy the agent in real racing or high-fidelity validation scenarios
  Collect real racing data $\mathcal{D}_{\mathrm{real}}$
  Update ego dynamics model using Eq. (A111)
  Update interaction model using Eq. (A112)
  $\mathcal{D}_{\mathrm{wm}} = \mathrm{Rollout}(W, \pi, \phi, \mathcal{E})$
  $\mathcal{D} = \mathcal{D}_{\mathrm{real}} \cup \mathcal{D}_{\mathrm{wm}}$
  Update decision policy π using Eqs. (A116)–(A117)
  Update multimodal planner ϕ using Eq. (A119)
  Refine feasible-motion envelope $\mathcal{E}$ as Section S4.4
end for
Return refined world model W, policy π, planner ϕ, and feasible-motion envelope $\mathcal{E}$

This loop links real-world deployment, world-model prediction, policy learning, planner update, and physical-limit regulation. Real racing data improve the fidelity of ego dynamics, interaction prediction, and physical-limit representation. The refined world model then generates structured rollouts for rare interactions and near-limit scenarios. The updated agent can subsequently collect more informative data in the next deployment round, enabling progressive improvement of cognitive reasoning and physical-limit utilization.

S8. Experimental Protocol, Simulation Setup, and Baselines

This section describes the experimental protocol, simulation environment, interaction scenario construction, baseline methods, and evaluation metrics used in the experiments. The validation combines real-vehicle deployment with simulation testing. Real-vehicle deployment provides high-value data under actual racing conditions, while simulation enables repeatable evaluation of adversarial interaction, ablation studies, and failure-case reproduction.

S8.1 Real-Vehicle Deployment

Real-vehicle experiments were conducted on the full-scale autonomous racing platform introduced in Supplementary Section S1.1 and Fig. A1. During real-vehicle deployment, the onboard system operated localization, opponent perception, behavior

decision and motion control modules and recorded synchronized state, control, perception, and environment data. The real-vehicle deployment exposed the onboard system to high-speed motion, near-limit tire–road interaction, and close-proximity racing conditions. It therefore provided the real-world basis for physical grounding, world-model construction, and validation in simulation.

The real-vehicle deployment was carried out in the context of the Abu Dhabi Autonomous Racing League (A2RL). A2RL is a full-scale autonomous racing competition designed to evaluate autonomous-driving software under high-speed and competitive racing conditions. The real-vehicle experiment was conducted on the North Circuit of the Yas Marina F1 circuit in Abu Dhabi, which provides a full-scale racing environment with high-speed straights and consecutive cornering segments. The track has a total length of 3002 m and contains 8 corners. The racetrack environment for real-vehicle deployment is shown in Fig. A14.

Fig. A14 Racetrack environment for real-vehicle deployment.

The real-vehicle deployment exposed the onboard system to high-speed motion, near-limit tire–road interaction, and close-proximity racing conditions. It therefore provided the real-world basis for physical grounding, world-model construction, and simulation-based validation.

S8.2 Real-Racing Data Collection

Real-racing data were collected during full-scale autonomous racing deployment under high-speed operation, near-limit tire–road interaction, and adversarial multi-agent interaction. Each record follows the transition structure defined in Supplementary Section S7 and contains the historical world states, the executed action, and the future world-state sequence, conditioned on the global track context. The onboard system recorded multi-source data streams at different sampling frequencies. These data

streams were time-synchronized and resampled to 10 Hz to form aligned real-racing samples.

The real-racing dataset is used in three ways. First, it provides ground-truth or reference signals for evaluating localization, ego-state estimation, opponent tracking, and prediction accuracy. Second, it is used to train and refine the ego dynamics model, interaction model, and physical-limit representation. Third, it provides real-world cases, including overtaking and spin events, that are reproduced in simulation for controlled analysis of prediction, decision-making, and failure recovery.

S8.3 Autoverse Simulation Setup

Controlled validation was conducted in the Autoverse simulator with virtual opponents. Autoverse provides high-fidelity vehicle dynamics simulation, while the virtual opponent vehicles enable multi-vehicle interaction scenarios. This setup enables systematic evaluation of the refined agent under controlled initial conditions, track configurations, opponent behaviors, and physical-limit conditions.

The ego vehicle is running in the Autoverse simulator. The baseline simulation experiments are conducted on the Yas Marina North Circuit, consistent with the real-vehicle deployment setting. Additional cross-circuit generalization experiments are performed on different track layouts, with detailed racetrack information described in Section S10.4. Virtual opponents are implemented as vectorized interactive agents responding to the ego vehicle in real time. Fig. A15 provides a conceptual visualization of the environment provided by the Autoverse simulator with virtual opponents.

Fig. A15 Illustration of the environment of Autoverse simulator with virtual opponents.

Adversarial interaction scenarios are constructed to evaluate interactive performance under physical constraints. Each interaction scenario is initialized with the opponent vehicles placed around the ego vehicle. The default interaction duration is 20 s.

Opponent vehicles are initialized by sampling the relative longitudinal distance, lateral position, and initial speed with respect to the ego vehicle. After scenario initialization, the opponent behaviors are generated by policies trained from interaction

demonstrations using imitation learning rather than fixed scripted rules. Each opponent policy takes the local track context, ego–opponent relative states, and recent motion history as inputs, and produces interactive driving actions in real time. The resulting virtual opponents reproduce representative racing behaviors such as holding line, defending, yielding, and competing during overtaking. Scenario variations are introduced by sampling the local track segment, ego–opponent initial relative position, speed difference, and opponent behavior condition.

For each method, scenarios are generated using the same scenario generation setting to ensure the repeatable and fair comparison. A trial is terminated when any of the following conditions occurs:

- the evaluation horizon is reached;
- collision occurs;
- the ego vehicle violates track boundaries;
- the ego vehicle enters an unrecoverable unstable state.

In the simulation environment, the ego agent uses the same decision, planning, world-model rollout, and physical-limit-constrained control interfaces as in the real-vehicle stack. The simulation environment provides ego and opponent states, track information, collision status, boundary violation status, and dynamic-response variables. This makes it possible to evaluate both cognitive-side performance, such as interaction prediction and decision success, and physical-side performance, such as acceleration-envelope utilization, tire slip, tracking performance, and maneuvering stability.

Simulation testing enables controlled comparison between the proposed method and baselines, ablation of world-model-based future-aware reasoning, and reproduction of representative real-racing cases for failure analysis and recovery validation.

S8.4 Baseline Methods

The proposed world-model-driven method is compared with three baselines: behavioral cloning, soft actor–critic, and a rule-based planner.

Behavioral cloning.

The behavioral cloning baseline learns a direct mapping from world-state representation to racing decisions using supervised learning from demonstration data. Given a policy state s_t corresponding to the racing world state w_t, the BC policy is trained by minimizing

$$L_{\mathrm{BC}} = \mathbb{E}_{(s_t, a_t^{\mathrm{e}})\sim\mathcal{D}} \left[\left\| \pi_\theta(s_t) - a_t^{\mathrm{e}} \right\|^2 \right], \tag{A120}$$

where a_t^{e} denotes the demonstration action. This baseline evaluates whether supervised imitation alone can provide sufficient interaction reasoning and generalization under adversarial racing conditions.

Soft actor–critic.

The SAC baseline trains a model-free reinforcement-learning policy using the same state and action interface as the proposed decision policy. It optimizes racing reward

through interaction with the simulator but does not use world-model-based parallel future evaluation for online candidate scoring. This baseline evaluates the performance of a policy-centric reinforcement-learning approach without explicit predictive world-model reasoning during deployment.

Rule-based planner.

The rule-based planner corresponds to the state-machine-based competition stack. It selects racing behaviors according to manually designed logic, such as following, overtaking, defending, and speed adjustment based on relative opponent position and track context. This baseline provides a real-deployment reference for comparing the refined world-model-driven policy against the original competition stack.

S8.5 Evaluation Metrics

For the benchmark comparison, all baselines are assessed using the same interaction and safety metrics. The downstream controller and actuation interfaces are kept consistent when applicable. To evaluate whether each method can complete competitive interaction scenarios safely and efficiently, the success rate, safe mileage, average speed, and overtaking frequency are adopted as evaluation metrics.

Beyond the benchmark comparison, additional metrics are used to evaluate agent performance and evolution in physical terms. Tire slips, tracking errors, and maneuvering stability are adopted to assess limit-aware control near the dynamics boundary. The average speed, lap time, and reachable acceleration states are used to characterize the evolution of the agent across refinement stages.

S8.6 Real-World Case Reproduction

Two representative real-racing cases are reproduced in simulation for controlled validation: a wheel-to-wheel overtaking case and a spin case. The details are introduced in Supplementary Section S10.2 and S10.3.

These case reproductions connect simulation evaluation with real-racing events. They allow the validation pipeline to test not only average performance, but also whether the world-model-centric agent can predict, diagnose, and correct the critical interaction and physical-limit failure modes.

S9. Safety Monitoring and Intervention Logic

The safety monitoring layer supervises the localization and ego-state estimation subsystem during real-vehicle deployment. Its purpose is to detect sensor dropout, inconsistent velocity measurements, and abnormal estimation residuals before they propagate to world-state construction, future-aware reasoning, and downstream control. The safety layer does not replace the world model or controller; instead, it provides an independent intervention mechanism for real-time deployment under high-speed racing conditions.

The output safety mode is defined as

$$\text{Mode} \in \{\textbf{Normal}, \textbf{Degradation}, \textbf{Emergency stop}\}. \tag{A121}$$

In **Normal** mode, the autonomy stack operates without additional restrictions. In **Degradation** mode, the ego vehicle switches to a reduced-speed operating mode with conservative planning and control limits. In **Emergency stop** mode, the ego vehicle applies immediate braking while maintaining the current heading as far as possible. The thresholds and timing parameters are platform-dependent and are therefore configured according to sensor characteristics and deployment conditions. Fig. A16 illustrates the overall safety logic.

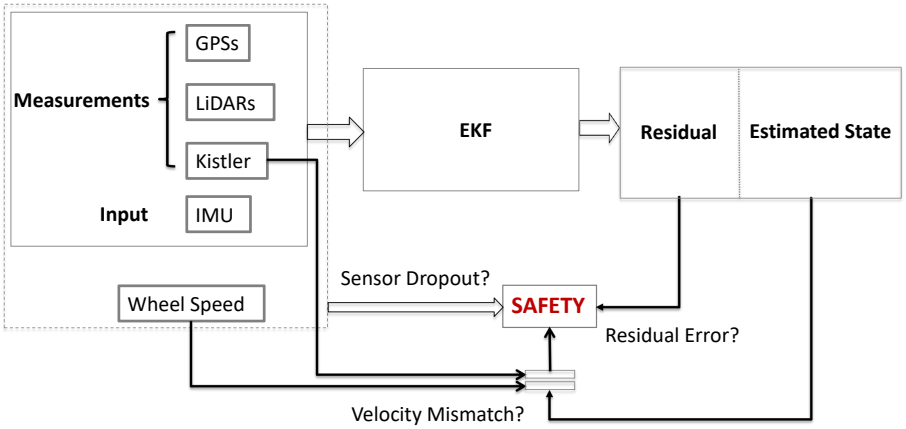


Fig. A16 Safety monitoring framework for localization and ego-state estimation.

S9.1 Sensor Dropout Logic

Each sensor stream is associated with a binary availability flag

$$s \in \{0, 1\}, \tag{A122}$$

where $s = 1$ indicates a valid and up-to-date signal and $s = 0$ indicates dropout or timeout. The position-source flag is defined as

$$s_{\text{pos}} = s_{\text{GNSS}} \vee s_{\text{LiDAR}}, \tag{A123}$$

which indicates whether at least one absolute-position source is available.

The dropout-based safety mode is

$$\text{Mode} = \begin{cases} \textbf{Emergency stop}, & \neg s_{\text{IMU}} \vee \neg s_{\text{pos}} \vee \neg s_{\text{whl}}, \\ \textbf{Degradation}, & s_{\text{IMU}} \wedge s_{\text{pos}} \wedge \neg s_{\text{Kis}}, \\ \textbf{Normal}, & s_{\text{IMU}} \wedge s_{\text{pos}} \wedge s_{\text{Kis}}. \end{cases} \tag{A124}$$

Here s_{IMU}, s_{GNSS}, s_{LiDAR}, s_{whl}, and s_{Kis} denote the availability of IMU, GNSS, LiDAR localization, wheel-speed, and Kistler velocity signals, respectively. The logic enforces

that the ego vehicle cannot continue normal operation without inertial sensing, at least one absolute-position source, and wheel-speed information.

S9.2 Velocity Consistency Logic

Velocity consistency is evaluated using redundant velocity sources. The mismatch terms are defined as

$$e_{kw} \triangleq \left| v_x^{\mathrm{Kis}} - v^{\mathrm{whl}} \right|, \qquad e_{ke} \triangleq \left| v_x^{\mathrm{Kis}} - v^{\mathrm{est}} \right|, \tag{A125}$$

where v_x^{Kis} is the Kistler longitudinal speed, v^{whl} is the average front-wheel longitudinal speed, and

$$v^{\mathrm{est}} = \sqrt{(v_x^{\mathrm{est}})^2 + (v_y^{\mathrm{est}})^2} \tag{A126}$$

is the estimator-derived speed magnitude.

A window-based Boolean comparison operator is used to avoid reacting to isolated spikes. For a signal e and threshold δ, the statement

$$e \overset{\mathcal{W},\rho}{>} \delta \tag{A127}$$

is true when the proportion of samples in sliding window $\mathcal{W}$ satisfying $e > \delta$ exceeds the ratio ρ.

The velocity-consistency safety mode is

$$\mathrm{Mode} = \begin{cases} \textbf{Emergency stop}, & e_{kw} \overset{\mathcal{W},\rho}{>} \delta_{kw} \vee e_{ke} \overset{\mathcal{W},\rho}{>} \delta_{ke}, \\ \textbf{Normal}, & \text{otherwise.} \end{cases} \tag{A128}$$

This check detects large disagreements among independent velocity estimates, which may indicate wheel-speed slip, external velocity sensor failure, or state-estimation inconsistency.

S9.3 Residual-Based Consistency Logic

Residual-based consistency checks compare incoming sensor measurements with predicted observations from the estimator. Let r_i denote the i-th measurement residual. Residual indicators are defined as

$$\begin{aligned} \mathcal{D}_{\mathrm{res}} &= \exists\, i \in \mathcal{I} : |r_i| \overset{\mathcal{W},\rho}{>} \delta_{d,i}, & \mathcal{C}_{\mathrm{Kis}} &= \exists\, i \in \mathcal{I}_{\mathrm{Kis}} : |r_i| \overset{\mathcal{W},\rho}{>} \delta_{c,i}, \\ \mathcal{C}_{\mathrm{GNSS}} &= \exists\, i \in \mathcal{I}_{\mathrm{GNSS}} : |r_i| \overset{\mathcal{W},\rho}{>} \delta_{c,i}, & \mathcal{C}_{\mathrm{LiDAR}} &= \exists\, i \in \mathcal{I}_{\mathrm{LiDAR}} : |r_i| \overset{\mathcal{W},\rho}{>} \delta_{c,i}. \end{aligned} \tag{A129}$$

Here $\mathcal{I}$ denotes all residual channels, while $\mathcal{I}_{\mathrm{Kis}}$, $\mathcal{I}_{\mathrm{GNSS}}$, and $\mathcal{I}_{\mathrm{LiDAR}}$ denote modality-specific residual subsets. The thresholds $\delta_{d,i}$ and $\delta_{c,i}$ represent degradation and critical residual thresholds, respectively.

The residual-based safety mode is

$$\text{Mode} = \begin{cases} \textbf{Emergency stop}, & \mathcal{C}_{\text{Kis}} \vee (\mathcal{C}_{\text{GNSS}} \wedge \mathcal{C}_{\text{LiDAR}}), \\ \textbf{Degradation}, & \mathcal{D}_{\text{res}}, \\ \textbf{Normal}, & \text{otherwise.} \end{cases} \tag{A130}$$

An emergency stop is triggered when critical residuals indicate Kistler failure, or when both GNSS and LiDAR position residuals violate their critical thresholds simultaneously. The latter condition implements mutual backup between the two absolute-position sources and avoids emergency intervention from a single-source anomaly alone.

S9.4 Intervention Priority and Deployment Use

When multiple safety checks are active, the final safety mode is selected according to the most conservative triggered mode:

$$\text{Mode}_{\text{final}} = \max_{\text{priority}} \left(\text{Mode}_{\text{dropout}}, \text{Mode}_{\text{velocity}}, \text{Mode}_{\text{residual}}\right), \tag{A131}$$

where the priority order is

$$\textbf{Emergency stop} > \textbf{Degradation} > \textbf{Normal}. \tag{A132}$$

The selected safety mode is then passed to the planning and control stack. In Normal mode, the system follows the standard world-model-driven perception–prediction–decision–control loop. In Degradation mode, the planner and controller use reduced speed limits and conservative physical-limit envelopes. In Emergency stop mode, the control module overrides the selected racing trajectory and applies a braking maneuver.

This safety monitoring layer provides an independent deployment safeguard for the high-frequency localization and state-estimation subsystem. It helps prevent sensor or estimator failures from propagating into incorrect world-state construction and unsafe high-speed execution.

S10. Additional Validation and Ablation Details

This section summarizes the validation and ablation settings used to support the main results. The details focus on four parts that are directly evaluated in the main manuscript: world-model-based future-aware reasoning, real-world overtaking case reproduction, physical-limit-aware adaptation in the spin case, and cross-circuit generalization.

S10.1 Ablation of World-Model-Based Future-Aware Reasoning

The proposed method is compared with an ablated variant that removes recursive world-model rollouts during online trajectory evaluation. In the ablated variant, candidate trajectories are generated using the same decision policy and multimodal planner, but the final trajectory is selected using only the current observed world state.

$$\tau^*_{t,\mathrm{abl}} = \arg\max_{\tau_t^k \in \mathcal{T}_t} S_{\mathrm{current}}(\tau_t^k, w_t). \tag{A133}$$

where S_{current} scores candidate trajectories through speed score and the driving safety field. In contrast, the full method generates multimodal candidate trajectories and selects the final trajectory using predicted future world states, as described in Eq. (A70).

This ablation isolates the effect of predictive reasoning. Comparing the full method with this variant shows whether world-model rollouts improve the agent's ability to anticipate opponent motion, avoid unsafe overtaking decisions, and select trajectories that remain executable under downstream physical-limit-constrained control.

S10.2 Real-World Overtaking Case Reproduction

The real-world overtaking case is reproduced to evaluate whether the world model captures short-horizon ego–opponent interaction evolution during wheel-to-wheel racing. The reproduced scenario uses the observed interaction geometry, ego trajectory, opponent trajectory, and track context as reference conditions. The analysis compares realized ego–opponent trajectories with world-model-predicted trajectories, and evaluates whether future-aware reasoning selects a maneuver that remains collision-free, physically feasible, and competitive.

This case supports the qualitative validation of the predictive reasoning process. The key comparison is whether the world-model-predicted interaction remains consistent with the realized wheel-to-wheel racing interaction, and whether the selected trajectory avoids the unsafe decision observed in the no-world-model ablation.

S10.3 Physical-limit-aware adaptation and Spin-Case Reproduction

The spin case observed during real-world racing is reproduced to evaluate physical-limit-aware adaptation and failure recovery. The reproduced scenario includes the high-curvature corner, aggressive acceleration command, high lateral load, and tire-state condition associated with the observed instability.

The comparison includes four settings:

1. baseline control without additional limit degradation;
2. thermal-adaptive limit degradation;
3. world-model-prediction-deviation-based limit degradation;
4. synergistic degradation combining thermal and world-model-based signals.

The evaluated quantities include slip angle, yaw response, trajectory tracking behavior, and corner-exit performance. This validation assesses whether the adaptive

physical-limit mechanism reduces the effective feasible-motion boundary before instability develops, thereby preventing or mitigating the spin failure mode observed during the race.

S10.4 Cross-Circuit Generalization Protocol

Cross-circuit generalization is conducted by deploying the agent on target circuits without additional training. The source circuit is the Yas Marina North Circuit, which has a total length of 3002 m and contains 8 corner segments. The target circuits include the Yas Marina Circuit, Autonodrome, and Suzuka Circuit, with total lengths of 5284 m, 6117 m, and 5809 m, respectively. They contain 16, 16, and 18 corner segments, respectively. These circuits differ in total length, curvature distribution, corner density, and the arrangement of high-speed straights and consecutive cornering segments, thereby providing diverse track geometries for evaluating cross-circuit transfer.

The same world-model interface, decision policy, multimodal planner, and physical-limit-constrained controller were used across all target circuits. For each circuit, random interaction scenarios were generated by sampling ego–opponent initial relative positions, speed differences, and opponent behavior conditions from the same predefined ranges. The evaluation reports the average speed, maximum speed, overtaking frequency, and success rate in random interaction scenarios and collision-related results. This protocol assesses whether the world model and policy preserve interaction capability and limit-aware execution under changes in track geometry and interaction contexts.