MPPI with Control Barrier Functions for F1/10: Robust Safety Under Real-World Uncertainty

Vaibhav Thakkar
University of Pennsylvania
Philadelphia, USA
vaithak@seas.upenn.edu

Rhythm Chandak
University of Pennsylvania
Philadelphia, USA
rhythm@seas.upenn.edu

Junfei Zhan
University of Pennsylvania
Philadelphia, USA
zjf2024@seas.upenn.edu

Ethan Yu
University of Pennsylvania
Philadelphia, USA
ethanyu@seas.upenn.edu

I. Introduction

Autonomous racing requires robust planners that are both safe and capable of real-time computation under uncertainty. Standard Model Predictive Path Integral (MPPI) controllers struggle with enforcing safety constraints under unmodeled dynamics and disturbances, typically requiring powerful GPUs for real-time performance [1]. However, for resource-limited platforms such as the F1/10, MPPI's computational demands are prohibitive. To overcome these limitations, we propose the integration of discrete-time Control Barrier Functions (CBFs) into MPPI, referred to as Shield-MPPI, enabling rigorous safety guarantees and robust performance even under disturbances and model uncertainties [2].

II. Related Work

MPPI, as introduced by Williams et al. [1], provides a sampling-based approach advantageous in handling nonconvex optimization landscapes encountered in real-world autonomous driving. However, standard MPPI lacks explicit safety constraint handling mechanisms, often resulting in unsafe trajectories when disturbances occur. Variants such as risk-aware MPPI [3], covariance-steering MPPI [4], and robust MPPI [5] mitigate uncertainty impacts but fail to provide strict safety guarantees. Conversely, CBFs offer rigorous safety assurances by enforcing forward invariance of a predefined safe set [6], [7]. The recently introduced Shield-MPPI integrates these concepts effectively, providing a computationally feasible and theoretically sound solution for robust autonomous control [2].

III. Proposed Methodology

We propose to implement Shield-MPPI on the F1/10 autonomous racing platform. The vehicle will be modeled using a discrete-time dynamic bicycle model. Track boundary adherence is enforced using a discrete-time CBF defined as:

$$h(x) = w_T^2 - e_y^2 \tag{1}$$

where w_T is half track width and e_y is lateral displacement from the track centerline.

The Shield-MPPI consists of two primary components:

1) Cost Augmentation: Trajectory costs are augmented with a CBF-based penalty:

$$C_{\text{cbf}}(x_k, x_{k-1}) = C \cdot \max\{-h(x_k) + \alpha h(x_{k-1}), 0\} \tag{2}$$

This strongly penalizes trajectories nearing safety violations.

2) Control Filtering: A local gradient-based optimization corrects nominal MPPI-generated control actions to ensure immediate safety:

$$v_{0:N}^{\text{safe}} = \arg\max_{v_{0:N}} \sum_{k=0}^{N} \min\{h(x_{k+1}) - \alpha h(x_k), 0\} \tag{3}$$

Experiments will explicitly introduce model uncertainties (e.g., incorrect friction parameters), external disturbances (such as random perturbations), and sensor noise, evaluating Shield-MPPI's robustness rigorously.

IV. Evaluation Plan

Performance evaluation will focus on three primary dimensions to rigorously assess the capabilities of Shield-MPPI against the standard MPPI approach on the F1/10 platform:

- **Safety**: Safety will be assessed by measuring collision frequency, adherence to track boundaries, and the system's recovery ability following disturbances or deviations. Specifically, we will evaluate:
  - The frequency and severity of boundary violations or collisions during high-speed centerline-following tasks.
  - The effectiveness in avoiding collisions with stationary obstacles placed at known track points.
  - The capability of detecting and avoiding collisions with stationary obstacles placed at unknown positions on the track (if time permits).
  - Performance and collision avoidance with dynamic obstacles introduced on the track (if time permits).
- **Efficiency**: Efficiency will be evaluated by comparing Shield-MPPI's average lap times, maximum sustainable speeds, and consistency in achieving high-speed consecutive laps without collisions against standard MPPI. Fig. 2 will illustrate the comparative performance improvements anticipated.
- **Computational Feasibility**: We will test real-time computational performance of Shield-MPPI versus standard MPPI, specifically analyzing:

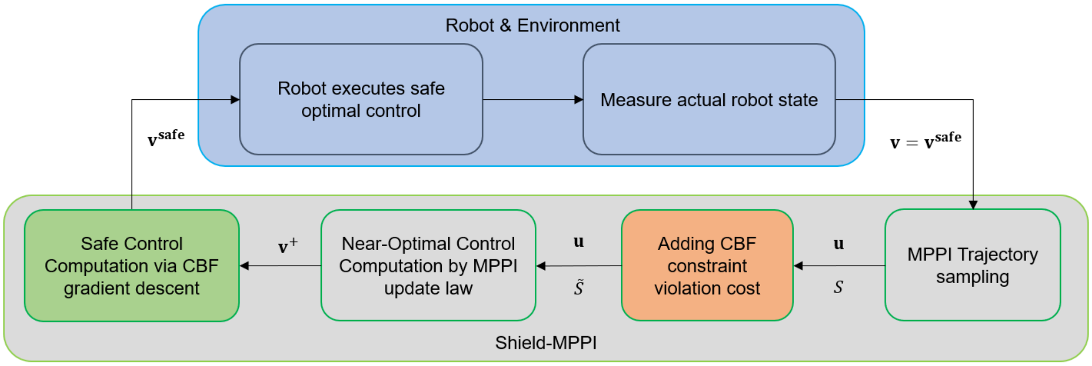

Fig. 1. Shield-MPPI Control Architecture [2].

- Real-time performance, achievable control loop frequencies, and computational feasibility using CPU-only versus GPU-accelerated implementations.
- Scalability in maintaining robust safety and control performance under varying computational resource constraints.

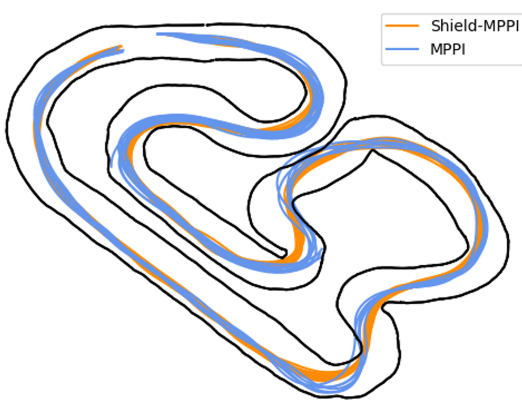

Fig. 2. performance comparison of Shield-MPPI vs standard MPPI [2].

V. Timeline and Plan

The following timeline outlines key milestones from April 7 to May 10 for implementing and evaluating Shield-MPPI on the F1/10 platform:

- **April 7–April 11:** Literature review, detailed problem formulation, and preliminary design of discrete-time dynamic bicycle model.
- **April 12–April 16:** Develop and validate discrete-time dynamic bicycle model; initial implementation of baseline standard MPPI controller.
- **April 17–April 21:** Complete MPPI baseline controller validation; implement and test Control Barrier Function integration within MPPI (cost augmentation and barrier condition enforcement).
- **April 22–April 26:** Integrate gradient-based safety filter (local control optimization); initial Shield-MPPI integration; extensive debugging and parameter tuning in simulation.
- **April 27–April 29:** Conduct extensive simulation experiments, robustness checks against model uncertainty, disturbances, and noise; perform comparative evaluation against baseline MPPI.
- **April 30:** Demonstration day—showcase the implemented Shield-MPPI controller on the F1/10 platform.
- **May 1–May 5:** Further validation, refinements based on demonstration feedback, additional experiments if necessary.
- **May 6–May 10:** Final analysis, documentation, and preparation of comprehensive results for report submission.

VI. Conclusion

Integrating Control Barrier Functions with MPPI control presents a promising pathway towards achieving robust, safe autonomy on the F1/10 racing platform. The proposed Shield-MPPI method provides rigorous safety assurances alongside performance advantages, particularly suited for real-time operations under uncertainty.

References

[1] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, “Information-theoretic model predictive control: Theory and applications to autonomous driving,” *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1603–1622, 2018.

[2] J. Yin, C. Dawson, C. Fan, and P. Tsiotras, “Shield model predictive path integral: A computationally efficient robust mpc approach using control barrier functions,” *IEEE Robotics and Automation Letters*, vol. 8, no. 11, pp. 7106–7113, 2023.

[3] J. Yin, Z. Zhang, and P. Tsiotras, “Risk-aware model predictive path integral control using conditional value-at-risk,” *arXiv preprint arXiv:2209.12842*, 2022.

[4] J. Yin, Z. Zhang, E. Theodorou, and P. Tsiotras, “Trajectory distribution control for model predictive path integral control using covariance steering,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022, pp. 1478–1484.

[5] M. S. Gandhi, B. Vlahov, J. Gibson, G. Williams, and E. A. Theodorou, “Robust model predictive path integral control: Analysis and performance guarantees,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1423–1430, 2021.

[6] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, “Control barrier functions: Theory and applications,” in *European Control Conference (ECC)*, 2019, pp. 3420–3431.

[7] M. Ahmadi, A. Singletary, J. W. Burdick, and A. D. Ames, “Safe policy synthesis in multi-agent pomdps via discrete-time barrier functions,” in *IEEE Conference on Decision and Control (CDC)*, 2019, pp. 4797–4803.