Predictive Safety Filters for Contact Rich Quadruped Locomotion

Anonymous Authors

***Abstract*—Learning-based policies have enabled quadruped robots to perform increasingly complex, contact-rich locomotion. However, these policies provide no safety guarantees, making deployment risky when domain shifts introduce unseen obstacles. Even though large-scale offline safe learning can circumvent this, it is not practically feasible to cover all edge cases. Classical safety frameworks like Hamilton-Jacobi Reachability, Control Barrier Functions, and Model Predictive Shielding offer formal guarantees but face fundamental limitations for legged systems: the curse of dimensionality, difficulty constructing valid barrier functions, and overly conservative recovery policies. We propose a minimally invasive predictive safety filter that targets *likely* safety rather than worst-case guarantees. Our approach formulates a sampling-based receding-horizon optimization over foot contact locations, warm-started by a nominal input and an optional recovery policy, and is bootstrapped with a learned value function. By optimizing in the compact contact-location space, the filter redirects footsteps around obstacles while an underlying contact-conditioned policy trained offline maps contacts to joint-level behavior. We evaluate the filter in simulation on a Unitree Go2 across three scenarios: (i) large-obstacle navigation, (ii) dynamic obstacle avoidance, and (iii) cluttered-environment traversal. Lastly, we show that it substantially reduces safety violations while preserving goal-reaching performance.**

Keywords: *Legged Robots, Safety Filters, Optimization*

I. Introduction

Learning-based locomotion policies have made remarkable progress in enabling quadruped robots to traverse diverse terrain. When combined with contact-conditioned architectures that accept desired foot placements as input, these policies produce highly dynamic whole-body behaviors [1]. However, they are trained to optimize task performance and lack a mechanism to prevent violations of safety in the environment. This is especially problematic when deployment introduces obstacles or spatial constraints that were not present during training.

Rather than retraining the policy with safety objectives, which is impractical for all possible cases, a more modular solution is a *safety filter*: a lightweight online layer that intervenes only when a constraint violation is imminent. Hamilton-Jacobi (HJ) Reachability [2], [3] provides provable guarantees but scales exponentially with state dimension, making it intractable for full-order legged models [4], [5], [6]. Control Barrier Functions (CBFs) [7], [8] are more scalable but require hand-designed barrier functions that are difficult to construct for intermittent multi-contact dynamics [9], [10], [11]. Model Predictive Shielding (MPS) [12], [13], [14] addresses both issues via forward simulation, but existing MPS methods rely on conservative recovery controllers that are misaligned with the locomotion objective.

We propose a predictive safety filter that addresses MPS conservatism while remaining practical. Our key insight is that for a quadruped with a contact-conditioned policy, the natural optimization variable is not the full joint-torque vector but the *foot contact locations*: a compact, physically interpretable search space sufficient for whole-body collision avoidance. The contact-conditioned policy provides the mapping from contacts to joint-level behavior, so the filter need not solve the full inverse problem.

Concretely, at each control step, nominal contact locations are provided by a predefined planning module. A forward simulation checks whether executing these contacts will produce a safety violation within a receding horizon. If a violation is predicted, the contacts are passed to a warm-start sampling-based optimizer, which searches for alternative contact sequences that satisfy safety constraints while remaining close to the nominal behavior. The finite-horizon cost is bootstrapped with a learned value function, following the spirit of Dynamic MPS [15], POLO [16], and LOOP [17]. Full physics simulation serves as the dynamics model, avoiding the model-mismatch issues of reduced-order approaches.

To summarize, our contributions are:

- A minimally invasive predictive safety filter for quadruped locomotion that operates in the compact contact-location space, enabling whole-body collision avoidance without modifying the underlying policy.
- A sampling-based optimization scheme warm-started by a nominal input and an optional recovery policy, and is bootstrapped with a learned value function.
- Simulation results on a Unitree Go2 navigating three contact-rich scenarios of increasing complexity.

II. Related Works

We now review some of the predictive safety filters and position our approach relative to the state of the art.

A. Predictive Safety Filters / Model Predictive Shielding

MPS forward-simulates candidate actions to check for violations and, if necessary, overrides the nominal input with a recovery policy [12], [13], [14]. It avoids the curse of dimensionality and does not require an analytical barrier function. The main drawback of existing MPS methods is the conservatism of recovery policies: standard recovery controllers are safe but misaligned with the locomotion objective. Dynamic MPS [15] addresses this by using a receding-horizon planner that accounts for long-term returns via a learned value function similar to [16], [17].

[18] optimizes over foot placements and step durations via signal temporal logic specifications within a reduced-order MPC for bipedal push recovery, demonstrating that the contact-location abstraction is effective for locomotion robustness; however, their reliance on a linear inverted pendulum model limits reasoning to CoM dynamics without whole-body

obstacle avoidance. [19] address scalability of HJ reachability and conservatism of MPS by training a game-theoretic fallback policy via adversarial self-play on the full 36-D quadruped state, but their predictive filter only performs a binary switch between the task policy and a pre-trained fallback at runtime via forward simulation. Additionally, it requires offline training to enforce safety objectives and cannot optimize actions online or adapt to novel configurations not seen during training. [20] uses nonlinear MPC as a safety filter for a learned control policy, but it assumes access to a given recoverable (viable) set, which is extremely challenging in general cases. In contrast, our work here relies on predictions under current policy and feasibility constraint checking, which removes the need for access to a recoverable set.

B. Positioning

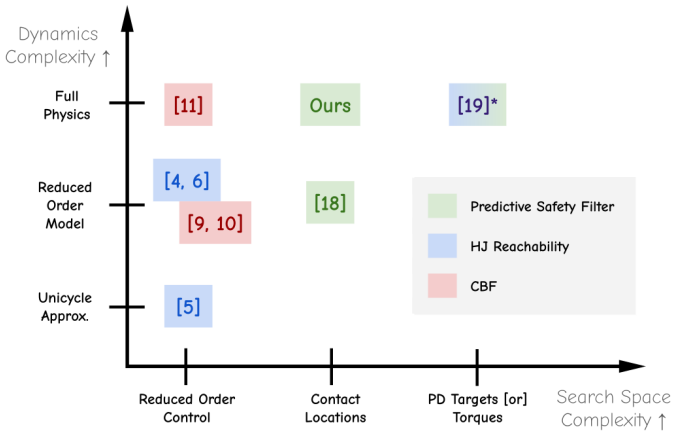


Fig. 1: Positioning of legged robot safety filter approaches by search space complexity and dynamics model fidelity. Our approach uniquely combines full-physics rollouts with optimization over contact locations. *[19] trains offline on full physics but only performs binary policy switching at runtime, without online optimization or adaptation to novel constraints.

See Fig 1 for an overview. Our approach builds on MPS while addressing its issues. Our key distinction is the choice of optimization variable: rather than base velocity commands (which cannot reason about limb placements) or joint torques (high-dimensional and expensive), we optimize over *foot contact locations*, leveraging the structure of contact-conditioned policies. We employ MPPI [21], a derivative-free sampler that naturally handles the non-smooth cost landscapes arising from contact dynamics.

III. Methodology

We now describe our predictive safety filter for whole-body collision avoidance on a Unitree Go2 quadruped equipped with a contact-conditioned locomotion policy [1].

A. General Framework

The safety filter sits *before* the contact-conditioned RL policy in the control pipeline; it is agnostic to the policy internals and only requires that the policy accept foot placements as input. At each control step:

1) Nominal contact locations are provided by a high-level planner, given the user's velocity command.
2) The contact-conditioned policy π takes the nominal contacts and proposes a sequence of PD targets over a horizon H.
3) A forward simulation rolls out these actions through the full pipeline (RL policy $\rightarrow$ PD controller $\rightarrow$ physics) and checks for violations.
4) If no violation is predicted, the nominal actions are executed directly.
5) If a violation is detected, new contact locations are optimized in parallel and fed to the RL policy:

$$\begin{aligned} \max_{p_0,\ldots,p_{H-1}} \quad & \mathbb{E}\left[\mathcal{R} = \sum_{t=0}^{H-1} \gamma^t r(s_t, a_t) + \gamma^H V(s_H)\right] \\ \text{s.t.} \quad & g(s_t, p_t) \leq 0, \\ & a_t = \pi(s_t, p_t), \\ & s_{t+1} = f(s_t, a_t), \quad \forall\, t = 0, \ldots, H-1. \end{aligned} \tag{1}$$

where $p_t \in \mathbb{R}^{3\times n_{\text{feet}}}$ denotes foot contact locations, a_t are PD targets from $\pi(s_t, p_t)$, $r(s_t, a_t)$ is the task reward, $V(s_H)$ is the learned value function bootstrapping the finite-horizon objective, g encodes safety constraints (Sec. III-C), and f is the full physics simulation based on Mujoco MJX [22]. In practice, we relax g into a soft penalty incorporated in the optimizer's cost (Sec III-D). An overview is shown in Fig. 2.

B. Choice of Optimization Variable

Existing safety filters for legged robots typically operate over base velocity commands: low-dimensional but unable to reason about limb placements or whole-body geometry. Optimizing directly over PD targets or joint torques is maximally expressive but yields a search space that is difficult to explore. We choose foot contact locations, motivated by two observations:

- **Sufficiency:** Whole-body collisions during locomotion are primarily determined by foot placements and how they shape the body's trajectory. Redirecting footsteps suffices to steer the robot around obstacles.
- **Compactness:** Contact locations are far more expressive than base velocity commands while remaining compact. The contact-conditioned policy maps contacts to joint-level behavior, so the optimizer need not solve the inverse problem.

C. Safety Constraints

We employ two complementary safety costs integrated as penalties into the optimizer.

1) Foot Contact Locations: We formulate a collision-avoidance cost using signed distance functions (SDFs) for obstacles approximated as geometric primitives. At each timestep t, the signed distance between foot i and obstacle j is:

$$\mathcal{D}_t^{ij} = \|p_t^i - c_t^j\| - r_j, \tag{2}$$

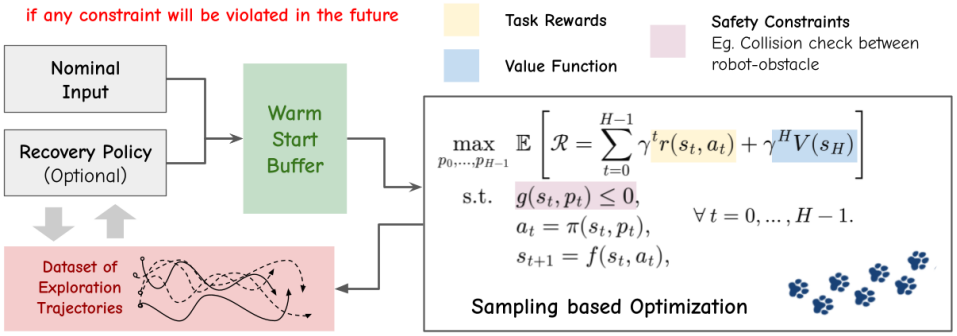


Fig. 2: Overview of the SafeMan safety filter. When a constraint violation is predicted, the sampling-based optimizer refines the contact-location sequence to maximize return. Candidate sequences are evaluated via parallel full-physics rollouts. If no violation is predicted, the nominal action is executed directly. Recovery policy training and collection of exploration dataset are optional extensions not evaluated in this work.

where c_t^j is the closest point on the obstacle and r_j is its margin radius. We convert $\mathcal{D}_t^{ij} \geq 0$ into the soft penalty $\sum_{i,j} \max(m - \mathcal{D}_t^{ij}, 0)^\alpha$, where $m > 0$ is a safety margin and $\alpha \geq 1$ controls steepness.

2) Whole-Body Collision: To capture collisions involving other body parts (legs, trunk), we query MuJoCo MJX's contact data during each parallel rollout, extracting a binary indicator $\mathbf{1}_{\text{col}}(s_t) \in \{0, 1\}$ for any contact between robot and obstacle. Trajectories with $\mathbf{1}_{\text{col}}(s_t) = 1$ at any t receive a large penalty, ensuring whole-body collisions are likely to be rejected. The combined safety cost is:

$$\mathcal{J}_{\text{safety}} = \sum_{t=0}^{H-1} \left(\sum_{i,j} \max(m - \mathcal{D}_t^{ij}, 0)^\alpha + M\, \mathbf{1}_{\text{col}}(s_t) \right), \quad (3)$$

where $M \ggg 0$ enforces rejection of colliding trajectories.

D. Sampling-Based Optimization

We adopt MPPI [21], which maintains a Gaussian over contact-location sequences centered at the nominal warm-start. At each iteration, K perturbations are sampled, rolled out through the full simulation pipeline, and the mean is updated via exponentially-weighted averaging:

$$p_t^* = \frac{\sum_{k=1}^K w_k\, p_t^{(k)}}{\sum_{k=1}^K w_k}, \quad w_k = \exp\left(\frac{1}{\lambda}\left(\mathcal{R}^{(k)} - \mathcal{J}_{\text{safety}}^{(k)}\right)\right), \quad (4)$$

where $\mathcal{R}^{(k)}$ is the return and $\mathcal{J}_{\text{safety}}^{(k)}$ the safety penalty for the k-th sample. MPPI is derivative-free, parallelizable on a GPU, and naturally handles discontinuities in cost due to contact events.

IV. Results

We evaluate the filter on a Unitree Go2 in MuJoCo MJX across three scenarios. All experiments use a planning frequency of 2 Hz, a horizon of $H = 5$ footsteps, $K = 512$ parallel rollouts, and $N = 1$ MPPI iteration per planning cycle on a single NVIDIA RTX 3090. Each planning cycle completes in $\sim$500 ms, with the MJX rollouts accounting for most of the time. The MPPI temperature is $\lambda = 0.1$, the SDF margin $m = 0.15$m, and the penalty exponent $\alpha = 2$.

A. Qualitative Evaluation

We first demonstrate the filter's behavior in two scenarios that highlight its core capabilities.

a) Scenario 1: Large obstacle navigation: The robot must navigate past a large obstacle occluding a significant portion of the workspace. As shown in Fig. 3a, the SDF cost guides the sampler toward contact sequences that route the robot around the obstacle, producing a smooth detour while maintaining a stable gait.

b) Scenario 2: Dynamic obstacle avoidance.: A moving obstacle crosses the robot's path. Fig. 3b shows the filter modifying planned contacts on-the-fly to steer clear, demonstrating reactivity to time-varying constraints within the receding horizon.

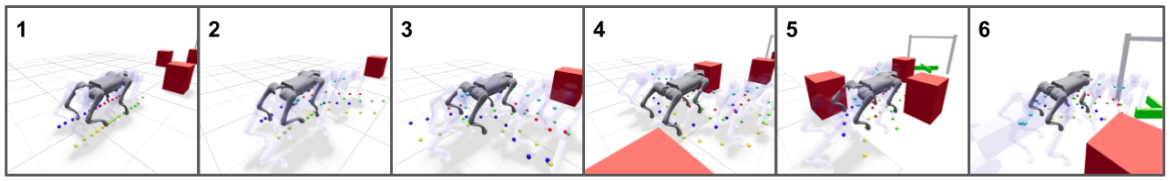
B. Quantitative Evaluation: Static Cluttered Environment

a) Scenario 3: Dense cluttered scene.: The Go2 navigates $\sim$50 obstacles in four zones of increasing density. We compare the unfiltered nominal policy against two MPPI configurations selected from a hyperparameter sweep over the safety penalty weight M. Table I summarizes the results.

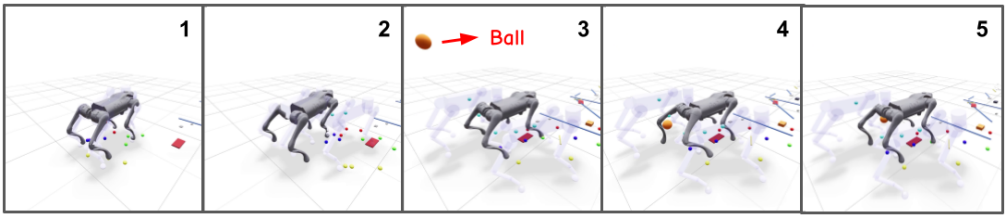
Method	Track. Cost ↓	Collisions ↓	Path Eff. ↑	Speed ↑
Nominal	$--$	80.7 ± 8.1	0.98 ± 0.01	0.12 ± 0.13
MPPI†	1.36 ± 0.08	55.5 ± 5.7	0.72 ± 0.03	0.05 ± 0.01
MPPI‡	1.46 ± 0.21	$\mathbf{49.7 \pm 2.2}$	0.68 ± 0.04	0.05 ± 0.01

TABLE I: Go2 cluttered-scene results (3 runs). $^\dagger M = 10^3$ (best path efficiency); $^\ddagger M = 10^4$ (best collision reduction). Collisions = whole-body contact events per run. Path efficiency = straight-line / actual path length. Speed in m/s.

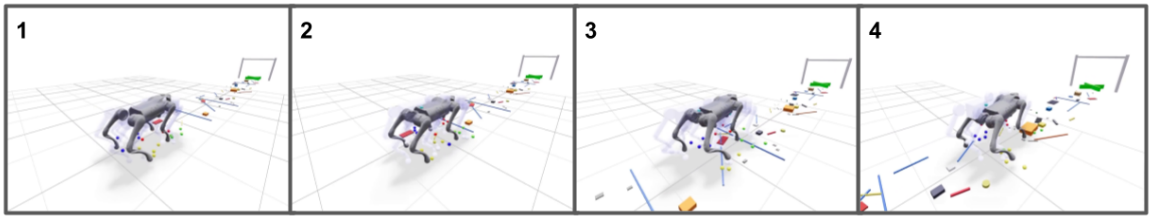
The nominal policy achieves high path efficiency (0.98) only because it walks straight through obstacles, incurring 80.7 collisions per run. Both MPPI configurations substantially reduce collisions: MPPI‡ achieves a 38% reduction (49.7 vs. 80.7) while MPPI† reduces collisions by 31% with more direct traversal. The trade-off is controlled by M: higher values yield safer but less direct paths. The speed reduction ($\sim$0.05 m/s vs. 0.12 m/s) reflects conservative replanning in dense regions, a reasonable trade-off since the nominal policy maintains speed only by ignoring obstacles.



(a) Large obstacle: the filter replans contacts (colored dots) to route around the obstacle. The transparent robot shows the planner's predicted trajectory.



(b) Dynamic obstacle: a ball approaches (frame 2) and the filter reactively shifts planned contacts to steer clear.



(c) Cluttered scene ($\sim$50 obstacles): the filter aggressively redirects contacts, trading path directness for collision avoidance.

Fig. 3: Qualitative results across three scenarios. Colored dots indicate planned foot-contact locations; the transparent robot shows the planner's predicted rollout. Keyframes progress left to right.

Moderate tracking costs indicate the filter modifies behavior only where necessary, consistent with the minimally invasive design. Remaining collisions occur predominantly in the densest traversal zones, where obstacles enter the planning horizon between the 2 Hz planning cycles, and the finite sample count $K = 512$ cannot always discover collision-free contact sequences in a single optimization pass.

V. Conclusion

We presented a minimally invasive predictive safety filter for contact-rich quadruped locomotion. By optimizing over foot contact locations rather than base velocities or joint torques, our approach exploits the structure of contact-conditioned policies to achieve whole-body collision avoidance in a compact search space. Full-physics rollouts via MuJoCo MJX avoid the model-mismatch issues of reduced-order approaches, and MPPI provides a derivative-free optimizer well-suited to the non-smooth cost landscapes of contact dynamics. We do not provide formal safety certificates; the sampling-based formulation offers high-probability constraint satisfaction that improves with the number of samples K and MPPI iterations.

Our simulation results on the Unitree Go2 demonstrate collision reductions of up to 38% compared to the unfiltered policy while preserving goal-reaching capability, across scenarios ranging from single large obstacles to dense cluttered environments and dynamic obstacles.

Current limitations include the 2 Hz planning frequency, which limits solution quality within the operating time window; reactivity to fast-moving obstacles; and reliance on known obstacle geometry for the SDF cost. In future work, we plan to (i) deploy the filter on hardware and evaluate sim-to-real transfer, (ii) increase the planning rate through better optimization or warm-starting strategies, and (iii) develop a principled online value function and recovery policy training scheme, as in online context this remains an open problem.

References

[1] S. Omar and M. Khadiv, "Learning to act through contact: A unified view of multi-task robot learning," *Learning for Dynamics and Control (L4DC)*, 2026.

[2] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin, "Hamilton-jacobi reachability: A brief overview and recent advances," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 2242–2253, 2017.

[3] I. M. Mitchell, "A toolbox of level set methods," 2005.

[4] S. Yang, H. Chen, L. Zhang, Z. Cao, P. M. Wensing, Y. Liu, J. Pang, and W. Zhang, "Reachability-based push recovery for humanoid robots with variable-height inverted pendulum," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3054–3060, 2021.

[5] J. Borquez, S. Peng, Y. Chen, Q. Nguyen, and S. Bansal, "Hamilton-jacobi reachability analysis for hybrid systems with controlled and forced transitions," *arXiv preprint arXiv:2309.10893*, 2023.

[6] X. Xia, J. J. Choi, A. Agrawal, K. Sreenath, C. J. Tomlin, and S. Bansal, "Gait switching and enhanced stabilization of walking robots with deep learning-based reachability: A case study on two-link walker," in *2024 IEEE 63rd Conference on Decision and Control (CDC)*, pp. 3402–3409, 2024.

[7] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2017.

[8] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European Control Conference (ECC)*, pp. 3420–3431, 2019.
[9] R. Grandia, A. J. Taylor, A. D. Ames, and M. Hutter, "Multi-layered safety for legged robots via control barrier functions and model predictive control," 2021.
[10] M. H. Cohen, T. G. Molnar, and A. D. Ames, "Safety-critical control for autonomous systems: Control barrier functions via reduced-order models," *Annual Reviews in Control*, vol. 57, p. 100947, 2024.
[11] R. M. Bena, G. Bahati, B. Werner, R. K. Cosner, L. Yang, and A. D. Ames, "Geometry-aware predictive safety filters on humanoids: From poisson safety functions to cbf constrained mpc," 2025.
[12] M. Alshiekh, R. Bloem, R. Ehlers, B. Könighofer, S. Niekum, and U. Topcu, "Safe reinforcement learning via shielding," *Proceedings of the AAAI Conference on Artificial Intelligence*, undefined.
[13] O. Bastani, "Safe reinforcement learning with nonlinear dynamics via model predictive shielding," in *American Control Conference*, 2021.
[14] W. Zhang and O. Bastani, "Mamps: Safe multi-agent reinforcement learning via model predictive shielding," in *arXiv.org*, 2019.
[15] A. Banerjee, K. Rahmani, J. Biswas, and I. Dillig, "Dynamic model predictive shielding for provably safe reinforcement learning," in *Neural Information Processing Systems*, 2024.
[16] K. Lowrey, A. Rajeswaran, S. Kakade, E. Todorov, and I. Mordatch, "Plan online, learn offline: Efficient learning and exploration via model-based control," 2019.
[17] H. Sikchi, W. Zhou, and D. Held, "Learning off-policy with online planning," 2021.
[18] Z. Gu, Y. Zhao, Y. Chen, R. Guo, J. K. Leestma, G. S. Sawicki, and Y. Zhao, "Robust-locomotion-by-logic: Perturbation-resilient bipedal locomotion via signal temporal logic guided model predictive control," 2024.
[19] D. P. Nguyen, K.-C. Hsu, W. Yu, J. Tan, and J. F. Fisac, "Gameplay filters: Robust zero-shot safety through adversarial imagination," 2025.
[20] X. Pua and M. Khadiv, "Safe learning of locomotion skills from mpc," in *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*, pp. 459–466, IEEE, 2024.
[21] G. Williams, A. Aldrich, and E. Theodorou, "Model predictive path integral control using covariance variable importance sampling," 2015.
[22] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, IEEE, 2012.