ROBUST ADAPTIVE MULTI-STEP PREDICTIVE SHIELDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning for safety-critical tasks requires policies that are both high-performing and safe throughout the learning process. While model-predictive shielding is a promising approach, existing methods are often computationally intractable for the high-dimensional, nonlinear systems where deep RL excels, as they typically rely on a patchwork of local models. We introduce RAMPS, a scalable shielding framework that overcomes this limitation by leveraging a learned, linear representation of the environment's dynamics. This model can range from a linear regression in the original state space to a more complex operator learned in a high-dimensional feature space. The key is that this linear structure enables a robust, look-ahead safety technique based on a *multi-step Control Barrier Function (CBF)*. By moving beyond myopic one-step formulations, RAMPS accounts for model error and control delays to provide reliable, real-time interventions. The resulting framework is minimally invasive, computationally efficient, and built upon robust control-theoretic foundations. Our experiments demonstrate that RAMPS significantly reduces safety violations compared to existing safe RL methods while maintaining high task performance in complex control environments.

1 INTRODUCTION

Deep reinforcement learning (RL) has achieved remarkable success in solving complex control problems, yet its deployment in safety-critical applications like autonomous vehicles and robotics remains a grand challenge Gu et al. (2022). A core requirement in these domains is not only that the final policy is safe, but that safety is maintained throughout the entire learning process. This problem of *safe exploration* has motivated a range of solutions, among which model-predictive shielding has emerged as a promising paradigm Jovanovi'c et al. (2020); Brunke et al. (2021).

Existing shielding frameworks present a difficult trade-off. On one hand, neural shields learn safety critics from data, offering flexibility but often requiring vast experience and failing to prevent violations during early training Bharadhwaj et al. (2021b); Dalal et al. (2018). On the other hand, symbolic shields provide formal, mathematical guarantees from the first interaction by analyzing an environment model Berkenkamp et al. (2017); Anderson et al. (2020); Wang & Zhu (2024). However, these methods have a critical limitation that has confined them to low-dimensional systems: they rely on explicitly partitioning the state space to construct a patchwork of local linear models. This approach suffers from the curse of dimensionality, rendering it computationally intractable for the complex, high-dimensional environments (> 10 dimensions) where modern deep RL excels.

This paper introduces RAMPS, a framework that bridges this critical gap by making formal shielding scalable to high-dimensional, nonlinear systems through a novel theoretical advance in safety certification. At the core of RAMPS is a new robust multi-step Control Barrier Function formulation that fundamentally changes how safety is guaranteed in discrete-time stochastic systems with model uncertainty. RAMPS achieves both theoretical soundness and practical scalability through a unified approach.

The key insight enabling RAMPS is the synergy between our robust multi-step CBF theory and the use of linear dynamics models. By representing the system dynamics through a single linear model, whether a linear regression in the original space or a learned operator in a high-dimensional feature space like the Deep Koopman OperatorShi & Meng (2022), we can efficiently propagate safety constraints multiple steps into the future while formally accounting for model error. Our

CBF formulation explicitly incorporates accumulated prediction error through a novel tightening mechanism, provides model-relative safety guarantees even with imperfect models. At each timestep, RAMPS's shield solves a comparatively-small Quadratic Program to find the minimally invasive safe action, with adaptive horizon selection that maximizes foresight while avoiding excessive conservatism.

Our contributions are threefold:

- We introduce RAMPS, a scalable shielding framework that provides probabilisitic safety guarantees, in high-dimensional, nonlinear systems by unifying robust CBF theory with learned linear dynamics representations.
- We develop a novel robust multi-step CBF formulation for discrete-time stochastic systems featuring accumulated error tightening and adaptive horizon selection, providing a principled solution to high relative-degree safety constraints under model uncertainty.
- We demonstrate that RAMPS significantly outperforms state-of-the-art safe RL methods, reducing safety violations by up to 90% and scaling to 348-dimensional environments, while maintaining competitive task performance across challenging high-dimensional control environments including quadrupedal locomotion.

2 RELATED WORK

Research in safe reinforcement learning (safe RL) can be categorized by *what kind of safety guarantees are provided* and *when those guarantees apply*. Safety is usually defined in two ways: (i) a *cost-based formulation*, where each action may incur some penalty and the long-term cost must remain below a threshold, or (ii) a *state-based formulation*, where specific regions of the state space are marked unsafe and must never be entered. Our work adopts the state-based view.

Worst-Case Guarantees. One line of work provides *deterministic safety guarantees* under a worst-case environment model, ensuring forward invariance by construction (Anderson et al., 2020; Gillula & Tomlin, 2012; Alshiekh et al., 2018; Zhu et al., 2019; Fulton & Platzer, 2019; Bacci et al., 2021). These approaches offer strong guarantees but require an explicit model and are limited to low-dimensional settings due to the computational cost of state-space partitioning. In contrast, RAMPS does not require a predefined model and remains tractable in high-dimensional systems.

Statistical Guarantees. Another family of methods offers *probabilistic or statistical safety guarantees*. These approaches build or learn an approximate dynamics model and optimize policies that are *likely* to be safe with respect to that model (Achiam et al., 2017; Liu et al., 2020; Yang et al., 2020; Ma et al., 2021; Zhang et al., 2020; Satija et al., 2020). While more scalable than worst-case methods, they typically allow safety violations during training. In contrast, RAMPS enforces hard constraints with respect to its learned model, reducing violations in practice.

Model-Predictive Shielding. A complementary paradigm is *model-predictive shielding* (MPS), where a shield monitors the agent's proposed action and intervenes only when it threatens safety (Wabersich & Zeilinger, 2018; Bastani, 2021; Anderson et al., 2020; 2023; Goodall & Belardinelli, 2023; Banerjee et al., 2024). Prior works differ in how they construct models and shields, but most struggle with scalability, particularly when moving beyond simple one-step predictions. Model-predictive shielding is closely related to **model predictive control (MPC)**: both use a model to roll out multi-step trajectories and solve constrained optimization problems. However, their objectives differ fundamentally. MPC optimizes long-horizon performance and effectively replaces the policy with its own control solution, whereas MPS acts purely as a *safety filter*: it retains the agent's action whenever it is safe, and otherwise solves a feasibility problem to return the closest safe alternative. This shifts the role of prediction from planning to minimal, targeted intervention, making shielding compatible with arbitrary RL policies while still enforcing hard safety guarantees.

Koopman Operators and Safety. Prior work has combined Koopman models with safety mechanisms, typically through *one-step* CBF filters. This includes Koopman-accelerated backup-CBF controllers, and neural or deep approaches that learn Koopman embeddings together with one-step CBF-QP filters or command governors Folkestad et al. (2020); Zinage & Bakolas (2022); Chen et al. (2024); Mitjans et al. (2024); Liang et al. (2025). Robust Koopman-MPC methods provide predictive control with error guarantees Mamakoukas et al. (2022); de Jong et al. (2024). However,

these methods either assume a known backup controller, rely on SMT-based CBF certification, or remain limited to one-step filtering and moderate-dimensional systems.

Cost-Based Safe RL. Cost-based methods enforce safety indirectly by shaping the reward with carefully designed cost signals and applying constrained optimization techniques (Achiam et al., 2017; Sootla et al., 2022a; Gu et al., 2024; Sootla et al., 2022b; Zhang et al., 2022; Yang et al., 2022). These approaches are flexible but inherently allow violations while the agent learns the cost structure. Compared to these methods, RAMPS enforces stricter state-based safety constraints, leading to fewer violations.

Safe RL methods balance a trade-off: *formal and symbolic methods* offer strong guarantees but do not scale, while *statistical and cost-based methods* scale but permit many violations early in training. Existing model-predictive shielding methods are typically limited to systems with state dimensions in the tens, as they rely on computationally expensive state-space partitioning or explicit nonlinear model propagation. RAMPS bridges this gap by combining a learned, linear model with a novel robust multi-step control barrier function, enabling scalable shielding with strong safety assurances in complex, high-dimensional environments, successfully operating on systems with over 300 state dimensions (where current formal techniques struggle above 10-dimensions), while maintaining real-time computational efficiency.

3 PRELIMINARIES

Safe Exploration. We model the environment as a Markov decision process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, r, P, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is a reward function, $P(x' \mid x, a)$ is a probabilistic transition function, and p_0 is an initial distribution over states. A *policy* π maps states to distributions over actions. The long-term return of a policy is $R(\pi) = \mathbb{E}_{s_i, a_i \sim \pi}\Big[\sum_{i=0}^{\infty} \gamma^i r(s_i, a_i)\Big]$. The goal of RL is to find an optimal policy $\pi^* = \arg\max_\pi R(\pi)$.

Most deep RL algorithms generate a sequence of policies $\pi_0, \pi_1, \ldots, \pi_N$ with $\pi_N \to \pi^*$. We refer to this sequence as a *learning process*. In *safe exploration*, the aim is to ensure that every intermediate policy remains safe with high probability. Formally, given a safety threshold δ and unsafe set $\mathcal{S}_U$, we require $\forall\, 1 \leq i \leq N,\ \Pr_{s\sim\pi_i}(s \in \mathcal{S}_U) \leq \delta$, while the final policy π_N maximizes reward among all safe policies. Following prior work (Anderson et al., 2023; Wang & Zhu, 2024), we do not require the initial policy π_0 to be safe, since no prior model of the environment is assumed.

Safety Specification. We adopt the common state-based notion of safety in safe RL. The unsafe set $\mathcal{S}_U$ is defined as a union of convex polyhedra over features of the state space (Anderson et al., 2023; Wang & Zhu, 2024). Equivalently, the safe set can be expressed as $\mathcal{S} \setminus \mathcal{S}_U = \bigcup_{i=1}^{M} \{\, s \in \mathcal{S} \mid G_i s \leq h_i \,\}$, for matrices G_i and vectors h_i. Unions of convex polyhedra are sufficient to approximate any compact safe set to arbitrary precision, and are widely used in model-predictive safety methods.

4 ROBUST ADAPTIVE MULTI-STEP PREDICTIVE SHIELDING

RAMPS, provides strong, real-time safety guarantees for reinforcement learning agents by integrating a learned, linear dynamics model with a robust, certificate-based safety shield. The framework is composed of three core components: (1) a learned linear dynamics model that provides a single, global representation of the environment's dynamics from data; (2) a Robust Control Barrier Function (CBF) that uses this model to certify safety and correct potentially unsafe actions online; and (3) a standard deep RL agent that learns a high-performance policy inside the protection of the shield.

The key requirement for the dynamics model is that it must be linear, as this structure enables the efficient, multi-step predictions required by the shield. This allows for a flexible range of modeling choices, from a simple linear regression operating in the original state space to a more complex Deep Koopman Operator (Shi & Meng, 2022) that learns a linear transition function in a high-dimensional feature space.

RAMPS operates in an iterative loop. The agent first collects a dataset of environment interactions. This data is used to train the linear dynamics model and a worst-case error bound, which in turn

parameterize the CBF shield. The RL agent is then trained, with every action being verified and potentially corrected by the shield to ensure safety. The newly collected, safe data is added back to the dataset, allowing the dynamics model and error bound to be periodically refined. This creates a cycle where a more accurate model leads to a less conservative shield, allowing the agent to explore more freely and learn a better policy. This is illustrated in Algorithm 1.

4.1 Safety Shielding with Multi-Step Robust Control Barrier Functions

We propose a safety shield designed to address a fundamental limitation of standard Control Barrier Functions (CBFs) when applied to discrete-time stochastic systems. Although one-step CBFs offer strong guarantees in continuous time, their discrete-time analogues may fail when a system's control inputs do not immediately affect the safety constraints; a challenge characterized by a relative degree greater than one. To resolve this issue, we construct a *multi-step robust CBF* by drawing upon principles from the theories of High-Order CBFs (HOCBFs; Tan et al. (2022)) and multi-step predictive control (Chriat & Sun, 2023). Our shield enforces safety over a variable prediction horizon H, which ensures that control authority is maintained despite such actuation delays. By adaptively selecting the largest feasible horizon at each timestep, the shield maximizes its predictive capability to eliminate "trap" states, which are configurations that appear safe in the short term but lead to inevitable future violations. This is accomplished while remaining minimally invasive to the actions proposed by the reinforcement learning agent's policy.

Control Barrier Functions (CBFs) (Nagumo, 1942; Prajna & Jadbabaie, 2004; Wieland & Allgöwer, 2007; Ames et al., 2019) are a powerful tool for enforcing safety constraints in control systems by rendering a specific region of the state space forward invariant. In the **continuous-time** setting, for a system with dynamics $\dot{x} = f(x) + g(x)u$ and a safe set defined as $\mathcal{C} = \{x \in \mathbb{R}^n \mid h(x) \geq 0\}$, a function h is a CBF if there exists a class-$\mathcal{K}$ function α such that for all $x \in \mathcal{C}$, the condition $\sup_{u \in U} [L_f h(x) + L_g h(x) u + \alpha(h(x))] \geq 0$ holds. This Lie derivative condition ensures that for any state on the boundary of the safe set, there exists a control action that prevents the system from instantaneously exiting $\mathcal{C}$.

In contrast, for a **discrete-time** system $x_{k+1} = F(x_k, u_k)$, the condition is fundamentally different. A function h is a discrete CBF if for all $x_k \in \mathcal{C}$, there exists a control $u_k \in U$ such that $h(F(x_k, u_k)) \geq \lambda h(x_k)$, where $\lambda \in [0, 1]$ is a decay rate. The key distinction lies in their temporal nature: the continuous condition is infinitesimal, guaranteeing safety based on the instantaneous velocity of the system, while the discrete condition provides a guarantee over a finite time step, ensuring that the state at step $k + 1$ remains safe given the state at step k. This often makes the discrete condition more conservative, as it must account for the system's evolution over the entire sampling period. Reinforcement learning typically deals with discrete-time systems.

Linear Dynamics. The core of our shielding framework relies on a learned, linear dynamics model, as this structure is essential for performing the efficient, multi-step predictions needed for robust safety analysis. For systems with simple dynamics, this can be a direct linear model operating in the original state space. For more complex, non-linear environments, the state can be "lifted" via a learned, non-linear embedding into a higher-dimensional feature space (Shi & Meng, 2022). The fundamental principle is that within this lifted space, the intricate dynamics can be accurately captured by a simple linear transition, $z_{k+1} = Az_k + Bu_k + c$. This transformation from non-linear to linear dynamics is what enables the shield to efficiently propagate safety constraints far into the future, making the approach scalable to a wide range of complex systems.

Safe Set and Dynamics. Let the lifted state space be $\mathbb{R}^{n+d}$ ($d \geq 0$) with the discrete-time affine dynamics

$$z_{k+1} \;=\; Az_k + Bu_k + c + w_k,$$

where c is a learned constant offset representing the system's drift, and w_k is an additive model error satisfying $\|w_k\|_\infty \leq \varepsilon$. The admissible control set is $\mathcal{U} \subset \mathbb{R}^m$. We define a polyhedral safe set $\mathcal{C}$ as the intersection of half-spaces, such that

$$\mathcal{C} \;=\; \bigcap_{i=1}^{M} \{\, z \;\mid\; p_i^\top z + b_i \leq 0 \,\}.$$

For each face i of the polyhedron, we define a corresponding safety function $h_i(z)$ as $h_i(z) = -\big(p_i^\top z + b_i\big)$, which means the safe set can be expressed as $\mathcal{C} = \{\, z \;\mid\; h_i(z) \geq 0,\ \forall i\}$.

One-Step Robust CBF Condition. To guarantee safety under model uncertainty, we formulate a robust CBF condition similar to (Cosner et al., 2023). The safety requirement is that the true next state, $z_{k+1} = Az_k + Bu_k + c + w_k$, must remain in the safe set $\mathcal{C}$. This implies that for each face i, the condition $p_i^\top (Az_k + Bu_k + c + w_k) + b_i \leq 0$ must hold for bounded disturbances w_k.

To ensure this, we design the constraint based on the worst-case disturbance, which has a value of $\varepsilon\|p_i\|_1$. By incorporating this worst-case term, we arrive at the robust CBF condition: for any state $z \in \mathcal{C}$, there must exist a control input $u \in \mathcal{U}$ such that

$$p_i^\top (Az + Bu + c) + b_i \;\leq\; \lambda\big(p_i^\top z + b_i\big) \;-\; \varepsilon\|p_i\|_1, \quad \forall i, \tag{1}$$

where $\lambda \in (0, 1]$ is a decay parameter that governs the conservatism of the barrier condition. Values of λ close to 1 require the safety function $h_i(z)$ to remain nearly constant across timesteps, leading to stricter constraints and stronger invariance. Smaller values of λ relax this requirement by permitting $h_i(z)$ to decay over time, which can improve feasibility but reduces the safety margin. The term $-\varepsilon\|p_i\|_1$ provides an additional robust margin, ensuring safety under the worst-case model error.

Relative Degree. The relative degree of a safety constraint $h(z)$ under dynamics $z_{k+1} = f(z_k) + g(z_k)u_k$ is the smallest integer $r \geq 1$ such that the control input u_k appears explicitly in the r-step evolution of $h(z_k)$, i.e. through $\frac{\partial h(z_{k+r})}{\partial u_k} \neq 0$.

Multi-Step Robust CBF Condition. The one-step condition in equation 1 is insufficient for systems where the control input has a delayed effect on a safety constraint (i.e., relative degree $r > 1$) 4.2. To eliminate the *trap* states that arise in such systems, our method ensures that the safety condition is met at *every* intermediate timestep j over a chosen horizon H, for all $j \geq r_i$. For each such step j, we define the nominal reachable state under a control sequence $\mathbf{u} = (u_0, \ldots, u_{H-1})$ as

$$z_j(z, \mathbf{u}) \;=\; A^j z + \sum_{k=0}^{j-1} A^{j-1-k} B u_k + \sum_{k=0}^{j-1} A^k c,$$

where the final term represents the cumulative effect of the affine drift. The total accumulated error over this j-step horizon is bounded by a tightening term, $\mathcal{E}_j(p_i)$, which sums the worst-case error at each step:

$$\mathcal{E}_j(p_i) \;=\; \sum_{k=0}^{j-1} \varepsilon\|p_i^\top A^k\|_1.$$

This leads to a set of robust CBF conditions, one for each valid step j and face i:

$$p_i^\top z_j(z, \mathbf{u}) + b_i \;\leq\; \lambda^j \big(p_i^\top z + b_i\big) \;-\; \mathcal{E}_j(p_i). \tag{2}$$

Each of these inequalities is linear with respect to the full control sequence $\mathbf{u}$. We aggregate all such constraints into a single system of linear inequalities, $G\mathbf{u} \leq h$, which guarantees that any feasible control sequence maintains the system within the safe set $\mathcal{C}$ throughout the entire horizon.

Minimally Invasive Action Selection. For a horizon H, the shield solves a Quadratic Program (QP) to find a safe control sequence that is minimally invasive to the RL agent's intended action, a_π. The primary objective is to find a control sequence $\mathbf{u} = (u_0, \ldots, u_{H-1})$ that minimizes the deviation of the first action, u_0, from the agent's proposal:

$$\begin{aligned} \min_{\mathbf{u}} \quad & \|u_0 - a_\pi\|_2^2 \\ \text{s.t.} \quad & G\mathbf{u} \leq h, \quad \text{(representing all constraints from equation 2)}, \\ & u_k \in \mathcal{U}, \quad k = 0, \ldots, H-1. \end{aligned} \tag{3}$$

Following the receding horizon principle, only the first action of the solution, u_0, is applied to the system. The subsequent actions, $u_{1:H-1}$, are optimized to ensure a feasible trajectory exists but are discarded, preserving flexibility at the next timestep.

Adaptive Horizon Selection and Safety Guarantee. At each timestep, we select the horizon H via a bounded binary search within $[H_{\min}, H_{\max}]$, where $H_{\min}$ is the maximum relative degree among active constraints. Candidate horizons are tested by solving the QP in equation 3: feasible horizons remain candidates while the search continues toward larger values, and infeasible ones shrink the range. The largest feasible horizon H^* determines the minimally invasive action u_0.

If no feasible horizon is found, a backup policy $u_{\text{backup}}(z)$ (A.4) is applied. Otherwise, the chosen action u_0 guarantees forward invariance: under disturbances $\|w_k\|_\infty \le \varepsilon$, the closed-loop system satisfies $z_k \in \mathcal{C}, \quad \forall k \le H$.

4.2 Analysis of the Shielding Framework

The efficacy of our framework stems from the powerful synergy between a learned linear dynamics model and the multi-step robust CBF shield. Each component is designed to address a fundamental challenge in safe control, and their integration yields a solution that is formally sound, robust to model error, and computationally tractable.

Synergy of a Linear Model and Multi-Step Shielding. The foundational element of our approach is the use of a linear dynamics model. This structure is the key enabler for our multi-step shield; it allows safety constraints, defined as simple polyhedra, to be accurately and efficiently propagated through time. Unlike methods that rely on repeated local linearizations or computationally expensive nonlinear propagation, our approach maintains tractability even over extended prediction horizons. This synergy is critical: the linear model makes multi-step prediction feasible, and the multi-step prediction is what gives the shield its foresight and power.

Robustness to Model Error. A core design principle of our framework is that it does not assume a perfect dynamics model. Instead, it achieves robustness by formally accounting for model error. The shield's safety guarantee is not based on the model's nominal prediction alone, but on a worst-case analysis that considers the maximum possible deviation. The robust tightening term, $\mathcal{E}_j(p_i)$, is derived from a data-driven error bound ε, effectively creating a *tube* of uncertainty around the predicted trajectory. By ensuring this entire tube remains within the safe set, the shield remains effective even when the learned linear model is an imperfect approximation of the true, complex dynamics. This allows the framework to work well even with simple models like linear regression, as it plans for their inherent inaccuracies.

Illustrative Example: Resolving High Relative-Degree Traps in Pendulum. The multi-step CBF framework also addresses traps in systems where the safety constraint depends on a state that the control input does not influence in a single step. Consider the pendulum environment with state $z = (\theta, \omega)$, representing angle and angular velocity. Its dynamics can be written in affine form as $z_{k+1} = Az_k + Bu_k + c(z_k)$, with $A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{3\Delta t}{m\ell^2} \end{bmatrix}, \quad c(z_k) = \begin{bmatrix} 0 \\ \frac{g\Delta t}{2\ell}\sin(\theta_k) \end{bmatrix}.$

Suppose we impose a safety constraint on the angle, $p^\top z + b = \theta + \delta \le 0$, with normal $p = [1\ 0]^\top$. The influence of the control input in one step is determined by $p^\top B$, which evaluates to 0. Thus, a one-step CBF cannot act directly on θ to prevent it from exceeding the bound. This creates a *relative-degree trap*: the shield has no immediate authority over the constrained variable.

In contrast, our multi-step formulation evaluates terms such as $p^\top A^{k-1}B$. For the pendulum, $p^\top AB = [1\ 0] \begin{bmatrix} \frac{3\Delta t^2}{m\ell^2} \\ \frac{3\Delta t}{m\ell^2} \end{bmatrix} = \frac{3\Delta t^2}{m\ell^2} \neq 0$. This non-zero term indicates that the control input does affect θ, but only after two steps. By enforcing constraints over a horizon $H \ge r$ (here, $r = 2$), our framework ensures that the control authority is accounted for, thereby resolving the trap. The affine term $c(z_k)$ shifts the dynamics but does not alter the relative-degree analysis. This mirrors the role of High-Order Control Barrier Functions in continuous-time systems (Tan et al., 2022; Chriat & Sun, 2023).

4.2.1 Conditional Safety Guarantees

The safety guarantee provided by our framework is a probabilistic certificate, which is standard for systems with learned dynamics. The argument is twofold: we first establish a deterministic guarantee of safety relative to our learned model and its error bound, and then connect this guarantee to the true physical system with a probabilistic bound.

Guarantee Relative to the Learned Model. Let the true, unknown, discrete-time dynamics of the system be governed by the function $F : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$, such that the true next state is $s_{k+1} = F(s_k, u_k)$. Our framework learns a linear model, which we denote as $\hat{F}$, that approximates these dynamics in a lifted space:

$$\hat{z}_{k+1} = \hat{F}(z_k, u_k) = Az_k + Bu_k + c.$$

The residual dynamics, or one-step prediction error, is the difference between the true evolution of the lifted state and the model's prediction, denoted by $w_k = z_{k+1} - \hat{z}_{k+1}$. Our shield is constructed using the model $\hat{F}$ and a worst-case bound on this error, $\|w_k\|_\infty \leq \varepsilon$. This leads to the following guarantee.

Theorem 1 (Conditional Model-Relative Forward Invariance)**.** *Given the learned dynamics model $\hat{F}$ and an error bound ε, if at every timestep k the multi-step robust CBF problem defined by the constraints in equation 2 is feasible, and the true residual dynamics satisfy $\|w_k\|_\infty \leq \varepsilon$, then the state of the system z_k is guaranteed to remain within the safe set $\mathcal{C}$ for all $k \geq 0$ (Blanchini, 1999).*

Proof. The proof is by construction and induction. At any state $z_k \in \mathcal{C}$, the feasibility of the QP in equation 3 implies the existence of a control sequence $\mathbf{u}$ that satisfies the robust multi-step CBF condition in equation 2. This condition, by its formulation, ensures that all intermediate states $z_{k+1}, \ldots, z_{k+H}$ remain within $\mathcal{C}$ for any possible realization of the error sequence where each $\|w_j\|_\infty \leq \varepsilon$. By applying the first action u_0 of this sequence, the resulting state z_{k+1} is guaranteed to be in $\mathcal{C}$. The argument then applies recursively at timestep $k+1$, as long as the condition stays feasible at $k+1$ □

While Theorem 1 establishes conditional recursive feasibility under the assumption that the QP remains feasible at every timestep, this requirement is standard but difficult to analytically guarantee in practical safe-control or safe-RL settings, especially when the dynamics model is learned. Prior model-based shielding and safe-exploration methods similarly rely on stepwise feasibility assumptions in their theoretical guarantees, while noting that infinite-horizon feasibility cannot be fully certified in practice and is instead supported empirically (Wang & Zhu, 2024; Anderson et al., 2023; Banerjee et al., 2024; Wachi et al., 2023). Consistent with this common limitation, we find that the QP in our framework is feasible in over 98% of timesteps, indicating that the theoretical assumption is well-satisfied in practice.

Probabilistic Connection to the Physical System. The deterministic guarantee of Theorem 1 is conditioned on the validity of the error bound ε. In practice, ε is estimated empirically from a finite hold-out validation dataset, D_{val}, as the maximum observed one-step prediction error. The connection between this empirical bound and the true, underlying error distribution is necessarily probabilistic, but the bound is maintained with high probability. Theorem 2 formalizes this connection. The proof is given in Appendix A.2.

Theorem 2 (High-Probability Model Accuracy)**.** *Let $\epsilon_1, \ldots, \epsilon_N$ be a set of i.i.d. sampled model errors from our learned model $\hat{F}$. Assume that the probability of any two samples being equal is zero. Choose a quantile $0 < q < 1$ and let ε be the $\lceil qN \rceil$'th smallest value among $\epsilon_1, \ldots, \epsilon_N$. Then*

$$Pr[\|F(s_k, u_k) - \hat{F}(s_k, u_k)\|_\infty > \varepsilon] \leq 1 - q + \frac{1}{(2N)^{1/3}} + \frac{1}{4(2^{1/3})N^{2/3}}.$$

Corollary 1 (Probabilistic Forward Invariance over Finite Horizon)**.** *Let $\delta = 1 - q + \frac{1}{(2N)^{1/3}} + \frac{1}{4(2^{1/3})N^{2/3}}$ be the failure probability of the empirical error bound ϵ from Theorem 2. If the multi-step robust CBF problem (Eq. 3) is feasible at every timestep $k \in \{0, \ldots, K-1\}$ over a finite horizon of K steps, then the true system state z_k remains within the safe set $\mathcal{C}$ for all $k \in \{0, \ldots, K\}$ with probability $P \geq 1 - K\delta$.*

Proof Sketch. By Theorem 2, $\Pr(||w_k||_\infty \leq \epsilon) \geq 1 - \delta$ for each timestep k. By union bound over K timesteps, $\Pr(\forall k \in \{0, \ldots, K-1\} : ||w_k||_\infty \leq \epsilon) \geq 1 - K\delta$. Conditioning Theorem 1's forward invariance on this high-probability event yields the result. □

5 Experimental Evaluation

We conduct experiments to evaluate RAMPS on a suite of challenging control tasks. Our evaluation is designed to answer three primary research questions:

1. **Safety Analysis**: Does RAMPS reduce safety violations more effectively than state-of-the-art safe RL algorithms?

2. **Safety-Performance Tradeoff**: Does the minimally invasive nature of RAMPS allow the agent to learn a high-performing policy?
3. **Role of Model Expressiveness** Does improved representational power of the learned dynamics model enhance the shielding performance of RAMPS?

Environments. We evaluate our method on five challenging environments. **Pendulum** is a classic low-dimensional control task. **SafeHopper**, **SafeCheetah**, **SafeAnt** and **SafeHumanoid** are high-dimensional locomotion tasks from the Safety-Gymnasium benchmark (Ji et al., 2023). **SafeHumanoid** is a challenging benchmarks due to their high-dimensional state (348) and action spaces (17) and the complex, unstable dynamics of legged locomotion, where sophisticated coordination is required to prevent falling.

Baselines. We compare RAMPS against two classes of baselines. First, we consider state-of-the-art Constrained Markov Decision Process (CMDP) algorithms that optimize for reward while treating safety as a constraint: **PPOSaute** (Sootla et al., 2022a), **P3O** (Zhang et al., 2022), and **CUP** (Yang et al., 2022). We use the implementations from the OmniSafe-RL library (Ji et al., 2024). We compare against these methods because, unlike many symbolic approaches, they are capable of operating in the high-dimensional environments we consider. We discuss additional baselines in Appendix A.5.3.

Second, we compare against methods architecturally similar to RAMPS, which also learn a dynamics model for shielding. We selected **SPICE** (Anderson et al., 2023), which learns a simple linear model; we refer to this as **SPICE + L**. To provide a direct comparison of modeling techniques, we also implemented **SPICE + K**, a variant where we replace the original linear model with our learned Koopman operator. We found that while **SPICE + L** failed to scale to the high-dimensional SafeHopper and SafeCheetah environments, **SPICE + K** was able to produce a stable model. We attempted comparisons with other relevant MPS/MPC techniques - DMPS (Banerjee et al., 2024), VELM (Wang & Zhu, 2024), MASE (Wachi et al., 2023), and Conservative Safety Critics (Bharadhwaj et al., 2021b), but these methods failed to achieve stable training on the high-dimensional locomotion tasks, accumulating over 1000 violations within the first 20-30k environment interactions. More details are in Appendix A.5.3.

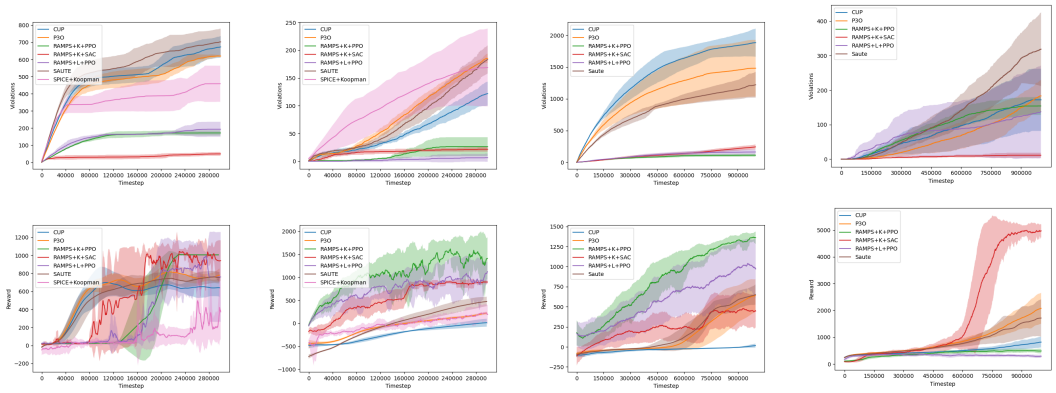
5.1 EXPERIMENTAL SETUP

Implementation Details. To analyze the impact of the learned dynamics model, we evaluate two versions of our RAMPS framework: **RAMPS + L**, which uses a simple linear model learned via regularized regression in the original state space, and **RAMPS + K**, which uses the Deep Koopman Operator. The underlying policy for RAMPS variants is trained with PPO and SAC. For all baselines, we add a penalty reward of -100 and terminate the episode upon a safety violation to provide a clear learning signal. For the CMDP baselines, the cost is 1 for a violation and 0 otherwise. We also ran CMDP baselines using only the sparse violation cost (cost = 1 for a violation, 0 otherwise) without the -100 episode-termination penalty; under this protocol the CMDP baselines failed to learn a safe or performant policy within our training budget. For all experiments involving RAMPS and SPICE, we use a maximum prediction horizon of $H_{max} = 5$, which is justified in A.3. We use OSQP Stellato et al. (2020) to solve the quadratic program.

5.2 RESULTS AND ANALYSIS

Safety Analysis. As detailed in Table 1, all variants of RAMPS demonstrates a substantial reduction in cumulative safety violations compared to various baselines across different environments. This effect is particularly pronounced in high-dimensional tasks like **SafeHopper**, **SafeCheetah**, **SafeAnt** and **SafeHumanoid**, where RAMPS variants typically exhibit significantly fewer violations than other methods. The violation curves in Figure 1, 6 visually reinforce these findings; while other methods often show a continued accumulation of violations during training, the curves for RAMPS variants tend to flatten much earlier in the training phase, indicating the shield's success in mitigating unsafe actions. This suggests that our multi-step shielding approach provides robust safety assurances, especially where optimization-based or other model-predictive methods may face challenges.

The effectiveness of RAMPS stems from its robust shielding framework, rather than solely relying on the underlying dynamics model's accuracy. A comparison between RAMPS + K and SPICE + K, both utilizing Koopman dynamics, reveals that RAMPS consistently achieves superior safety performance.



(a) Hopper (b) Cheetah (c) Ant (d) Humanoid

Figure 1: Cumulative Safety violations (top row in each subfigure) and episodic reward (bottom row) for all high-dimensional environments.

Algorithm	Pendulum	SafeHopper	SafeCheetah	SafeAnt	SafeHumanoid
SauteRL	91 ± 22	703 ± 78	183 ± 25	1221 ± 203	319 ± 106
CUP	184 ± 225	673 ± 63	122 ± 22	1883 ± 221	172 ± 90
P3O	173 ± 166	620 ± 6	185 ± 8	1481 ± 446	183 ± 45
SPICE + L	495 ± 128	Failed	Failed	Failed	Failed
SPICE + K	87 ± 8	459 ± 105	169 ± 70	Failed	Failed
RAMPS + L + PPO	69 ± 6	193 ± 44	$\mathbf{7 \pm 7}$	162 ± 42	137 ± 134
RAMPS + K + PPO	53 ± 6	172 ± 15	26 ± 17	$\mathbf{111 \pm 23}$	154 ± 25
RAMPS + K + SAC	$\mathbf{25 \pm 26}$	$\mathbf{49 \pm 10}$	21 ± 4	242 ± 38	$\mathbf{11 \pm 7}$

Table 1: Cumulative safety violations during training. Failed indicates that training quit or the agent never completed a safe episode. L = Linear Regression baseline; K = Koopman Dynamics model.

This difference highlights RAMPS's ability to operate effectively even with an imperfect model, due to its explicit accounting for model error through robust multi-step predictions. While methods like SPICE typically require a highly accurate model with minimal error to ensure stable performance, RAMPS's design allows it to maintain safety guarantees across a broader range of model accuracies. Furthermore, the real-time operation of the RAMPS shield is highly efficient. As detailed in Table 2, the mean per-step computation time ranges from just 0.23 ms for Pendulum to 0.40 ms for the high-dimensional Ant environment, suggesting feasibility for real-time control loops.

The Safety-Performance Tradeoff. A critical aspect of safe RL is balancing stringent safety with high task performance. The reward curves in Figure 1 illustrate that RAMPS effectively navigates this tradeoff. Across most environments, RAMPS achieves strong safety while obtaining competitive, and often superior, task rewards compared to the baselines. This suggests that the shield provides necessary interventions without being overly conservative, allowing the policy to explore and exploit high-reward regions.

Policy-Agnostic Shielding. We evaluate RAMPS with both PPO (on-policy) and SAC (off-policy) to highlight that the shield operates independently of the underlying RL algorithm. SAC is generally more reliable, especially in high-dimensional settings such as **SafeHumanoid**, while PPO performs competitively and even surpasses SAC on **SafeAnt**. These results indicate that RAMPS is compatible with multiple learning paradigms and scales effectively to challenging continuous-control tasks.

The PPO instability observed on Humanoid is not a shield-specific failure but a known limitation of on-policy methods under action modification. Prior work shows that even simple invalid-action masking, structurally analogous to shielding because the executed action differs from the policy's proposal, can cause PPO's KL divergence to spike and training to collapse (Huang & Ontañón, 2020; Hou et al., 2023). Similar sensitivity has been documented in safe-RL algorithms such as CPO and

primal–dual CMDP methods, where constraint-induced distribution shift destabilizes updates without additional safeguards (Achiam et al., 2017; Paternain et al., 2019; Ding et al., 2020). In contrast, off-policy approaches exhibit greater robustness to distribution mismatch (Haarnoja et al., 2018; Liu et al., 2022).

Empirical Feasibility of Multi-step constraints A crucial element of our framework's reliability is the practical feasibility of the multi-step QP. We analyzed the action selection distribution, and the results (detailed in Appendix A.5.5, Table 3) confirm our shield is highly robust. For the complex locomotion tasks, the backup policy was invoked in **less than 2% of all timesteps**, and for Pendulum and SafeHumanoid, it was never used at all. This demonstrates that our primary shield consistently finds a feasible, safe solution, validating the empirical stability of our approach and showing that the conditional guarantee of Theorem 1 is almost always active.

Role of Model Expressiveness. The choice of dynamics model within RAMPS can influence this balance between safety and reward, particularly in environments with complex dynamics. While both RAMPS + L (simple linear model) and RAMPS + K (Koopman model) offer significant safety improvements, the more expressive Koopman model generally supports better reward performance. This is observed in all environments, but particularly in **SafeCheetah**, where RAMPS + L achieves extremely low violations but shows lower reward accumulation compared to RAMPS + K. As shown in Appendix A.6, this is an outcome of the simpler linear model leading to a more conservative shield (due to larger estimated error bounds), resulting in interventions with larger deviations from the neural action. This hinder the agent's ability to learn a policy that maximizes the reward. In contrast, the more accurate Koopman model allows for a less conservative, yet still provably safe, shield, thereby improving the overall safety-performance balance.

Ablation Analysis. We performed an extensive ablation analysis, detailed in Appendix A.3, to validate the design principles of the RAMPS framework. These studies confirm that robust, multi-step shielding is a co-designed system requiring a careful balance of competing factors. Our most critical finding is that explicit **robustness to model error is the essential component for safety**; removing the error-aware tightening term proved catastrophic, leading to continuous safety violations regardless of other hyperparameter settings (Figure 2).

Furthermore, the ablations justify our hyperparameter choices by exploring key trade-offs. The prediction horizon H must be long enough to resolve high relative-degree traps but short enough to avoid compounding model error (Figure 4). The CBF decay rate λ must be permissive enough to ensure the underlying QP remains feasible, as an overly conservative setting harms both safety and reward (Figure 3). Finally, we show that a high-confidence error bound (99th percentile) is a prerequisite for achieving both safety and high reward, as it creates a more stable learning environment (Figure 5). We further evaluate RAMPS under *multi-dimensional safety constraints* to demonstrate scalability A.7. In the SAFEHUMANOID benchmark, we simultaneously constrain the 3 coordinate and 18 joint angular velocities (a 21-dimensional safety set). RAMPS accumulates only **256 violations**, whereas CMDP-based baselines exceed **3000 violations** and fail to learn a safe policy, reflected by their steadily increasing violation curves. Additonally, RAMPS is the only method that attains a high task reward of **5,000**, while CMDP baselines plateau near **500**. Together, these results show that RAMPS maintains safety even under high-dimensional constraints without sacrificing performance. Collectively, these results validate our methodology and demonstrate that effective shielding arises from a calibrated synthesis of all framework components.

6 CONCLUSION

We present RAMPS, a scalable model-predictive shielding framework that enables safe policy learning for complex, high-dimensional systems. The core of our approach is the synergy between a learned, linear dynamics model and a robust, multi-step safety shield. By leveraging a linear representation, which can range from a simple regression to a more complex latent model like the koopman operator, RAMPS remains computationally tractable. Its multi-step, adaptive-horizon Control Barrier Function provides strong foresight to prevent safety violations, even when the learned model is an imperfect approximation of the true dynamics. Experiments on a suite of challenging environments demonstrate the efficacy of RAMPS, showing it can dramatically reduce safety violations while maintaining high task performance. Its ability to learn a reliable safety model from a few samples makes it particularly well-suited for deployment of reinforcement learning agents in safety-critical applications.

Reproducibility Statement

We are committed to ensuring the reproducibility of our work. To this end, we provide the following resources. **Source Code:** The complete implementation of the RAMPS framework, including the Koopman dynamics model, the multi-step CBF shield, and all training scripts used to generate our results, is available as supplementary material. **Theoretical Foundations:** The mathematical formulation of our multi-step robust CBF, including the derivation of the safety constraints and the robust tightening term, is detailed in Section 4. The probabilistic safety guarantees relative to the learned model are established in 4. **Experimental Details:** Our experimental setup, including environment descriptions, safety specifications, and baseline implementations, is described in Section 5 and Section A. Furthermore, our extensive ablation studies, detailed in Appendix A.3, provide a clear analysis of hyperparameter sensitivity and justify our final configuration choices. We believe these resources provide a clear and complete path for reproducing our findings and building upon our work.

References

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 22–31. JMLR.org, 2017.

Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In Sheila A. McIlraith and Kilian Q. Weinberger (eds.), *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 2669–2678. AAAI Press, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17211.

Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European control conference (ECC)*, pp. 3420–3431. Ieee, 2019.

Greg Anderson, Abhinav Verma, Isil Dillig, and Swarat Chaudhuri. Neurosymbolic reinforcement learning with formally verified exploration. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6172–6183. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/448d5eda79895153938a8431919f4c9f-Paper.pdf.

Greg Anderson, Swarat Chaudhuri, and Isil Dillig. Guiding safe exploration with weakest preconditions. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://iclr.cc/virtual/2023/poster/12258.

Edoardo Bacci, Mirco Giacobbe, and David Parker. Verifying reinforcement learning up to infinity. In Zhi-Hua Zhou (ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 2154–2160. International Joint Conferences on Artificial Intelligence Organization, 8 2021. doi: 10.24963/ijcai.2021/297. URL https://doi.org/10.24963/ijcai.2021/297. Main Track.

Arko Banerjee, Kia Rahmani, Joydeep Biswas, and Isil Dillig. Dynamic model predictive shielding for provably safe reinforcement learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=x2zY4hZcmg.

Osbert Bastani. Safe reinforcement learning with nonlinear dynamics via model predictive shielding. In *2021 American Control Conference (ACC)*, pp. 3488–3494, 2021. doi: 10.23919/ACC50511.2021.9483182.

Felix Berkenkamp, Matteo Turchetta, Angela Schoellig, and Andreas Krause. Safe model-based reinforcement learning with stability guarantees. *Advances in neural information processing systems*, 30, 2017.

Homanga Bharadhwaj, Aviral Kumar, Nicholas Rhinehart, Sergey Levine, Florian Shkurti, and Animesh Garg. Conservative safety critics for exploration. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=iaO86DUuKi.

Homanga Bharadhwaj, Aviral Kumar, Nicholas Rhinehart, Sergey Levine, Florian Shkurti, and Animesh Garg. Conservative safety critics for exploration. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021b. URL https://openreview.net/forum?id=iaO86DUuKi.

Franco Blanchini. Set invariance in control. *Automatica*, 35(11):1747–1767, 1999.

Lukas Brunke, Melissa Greeff, Adam W. Hall, Zhaocong Yuan, Siqi Zhou, Jacopo Panerati, and Angela P. Schoellig. Safe learning in robotics: From learning-based control to safe reinforcement learning. *ArXiv*, abs/2108.06266, 2021. URL https://arxiv.org/pdf/2108.06266.pdf.

Hao Chen, Xiangkun He, Shuo Cheng, and Chen Lv. Deep koopman operator-informed safety command governor for autonomous vehicles. *IEEE/ASME Transactions on Mechatronics*, 29(5): 3568–3578, 2024.

Alaa Eddine Chriat and Chuangchuang Sun. On the optimality, stability, and feasibility of control barrier functions: An adaptive learning-based approach. *IEEE Robotics and Automation Letters*, 8 (11):7865–7872, 2023.

Ryan K Cosner, Preston Culbertson, Andrew J Taylor, and Aaron D Ames. Robust safety under stochastic uncertainty with discrete-time control barrier functions. *arXiv preprint arXiv:2302.07469*, 2023.

Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerík, Todd Hester, Cosmin Paduraru, and Yuval Tassa. Safe exploration in continuous action spaces. *CoRR*, abs/1801.08757, 2018. URL http://arxiv.org/abs/1801.08757.

Thomas de Jong, Valentina Breschi, Maarten Schoukens, and Mircea Lazar. Koopman data-driven predictive control with robust stability and recursive feasibility guarantees. In *2024 IEEE 63rd Conference on Decision and Control (CDC)*, pp. 140–145. IEEE, 2024.

Dongsheng Ding, Kaiqing Zhang, Tamer Basar, and Mihailo Jovanovic. Natural policy gradient primal-dual method for constrained markov decision processes. *Advances in Neural Information Processing Systems*, 33:8378–8390, 2020.

Carl Folkestad, Yuxiao Chen, Aaron D Ames, and Joel W Burdick. Data-driven safety-critical control: Synthesizing control barrier functions with koopman operators. *IEEE Control Systems Letters*, 5 (6):2012–2017, 2020.

Nathan Fulton and André Platzer. Verifiably safe off-model reinforcement learning. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pp. 413–430. Springer, 2019.

Jeremy H. Gillula and Claire J. Tomlin. Guaranteed safe online learning via reachability: tracking a ground target using a quadrotor. In *IEEE International Conference on Robotics and Automation, ICRA 2012, 14-18 May, 2012, St. Paul, Minnesota, USA*, pp. 2723–2730, 2012. doi: 10.1109/ICRA.2012.6225136. URL https://doi.org/10.1109/ICRA.2012.6225136.

Alexander W. Goodall and Francesco Belardinelli. Approximate model-based shielding for safe reinforcement learning. In Kobi Gal, Ann Nowé, Grzegorz J. Nalepa, Roy Fairstein, and Roxana Radulescu (eds.), *ECAI 2023 - 26th European Conference on Artificial Intelligence, September 30 - October 4, 2023, Kraków, Poland - Including 12th Conference on Prestigious Applications of Intelligent Systems (PAIS 2023)*, volume 372 of *Frontiers in Artificial Intelligence and Applications*, pp. 883–890. IOS Press, 2023. doi: 10.3233/FAIA230357. URL https://doi.org/10.3233/FAIA230357.

Shangding Gu, Longyu Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, Yaodong Yang, and Alois Knoll. A review of safe reinforcement learning: Methods, theory and applications. *ArXiv*, abs/2205.10330, 2022. URL https://api.semanticscholar.org/CorpusId:248965265.

Shangding Gu, Bilgehan Sel, Yuhao Ding, Lu Wang, Qingwei Lin, Ming Jin, and Alois Knoll. Balance reward and safety optimization for safe reinforcement learning: A perspective of gradient manipulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 21099–21106, 2024.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/haarnoja18b.html.

Yueqi Hou, Xiaolong Liang, Jiaqiang Zhang, Qisong Yang, Aiwu Yang, and Ning Wang. Exploring the use of invalid action masking in reinforcement learning: A comparative study of on-policy and off-policy algorithms in real-time strategy games. *Applied Sciences*, 13(14):8283, 2023.

Shengyi Huang and Santiago Ontañón. A closer look at invalid action masking in policy gradient algorithms. *arXiv preprint arXiv:2006.14171*, 2020.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Josef Dai, and Yaodong Yang. Safety gymnasium: A unified safe reinforcement learning benchmark. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=WZmlxIuIGR.

Jiaming Ji, Jiayi Zhou, Borong Zhang, Juntao Dai, Xuehai Pan, Ruiyang Sun, Weidong Huang, Yiran Geng, Mickel Liu, and Yaodong Yang. Omnisafe: An infrastructure for accelerating safe reinforcement learning research. *Journal of Machine Learning Research*, 25(285):1–6, 2024. URL http://jmlr.org/papers/v25/23-0681.html.

Mihailo R. Jovanovi'c, Dongsheng Ding, Xiaohan Wei, Zhuoran Yang, and Zhaoran Wang. Provably efficient safe exploration via primal-dual policy optimization. In *International Conference on Artificial Intelligence and Statistics*, 2020. URL https://api.semanticscholar.org/CorpusId:211677570.

Kaier Liang, Guang Yang, Mingyu Cai, and Cristian-Ioan Vasile. Safe navigation in dynamic environments using data-driven koopman operators and conformal prediction. *arXiv preprint arXiv:2504.00352*, 2025.

Zuxin Liu, Hongyi Zhou, Baiming Chen, Sicheng Zhong, Martial Hebert, and Ding Zhao. Constrained model-based reinforcement learning with robust cross-entropy method, 2020.

Zuxin Liu, Zhepeng Cen, Vladislav Isenbaev, Wei Liu, Steven Wu, Bo Li, and Ding Zhao. Constrained variational policy optimization for safe reinforcement learning. In *International Conference on Machine Learning*, pp. 13644–13668. PMLR, 2022.

Yecheng Jason Ma, Andrew Shen, Osbert Bastani, and Dinesh Jayaraman. Conservative and adaptive penalty for model-based safe reinforcement learning, 2021. URL https://arxiv.org/abs/2112.07701.

Giorgos Mamakoukas, Stefano Di Cairano, and Abraham P Vinod. Robust model predictive control with data-driven koopman operators. In *2022 American Control Conference (ACC)*, pp. 3885–3892. IEEE, 2022.

Marc Mitjans, Liangting Wu, and Roberto Tron. Learning deep koopman operators with convex stability constraints. *arXiv preprint arXiv:2404.15978*, 2024.

Mitio Nagumo. Über die lage der integralkurven gewöhnlicher differentialgleichungen. *Proceedings of the physico-mathematical society of Japan. 3rd Series*, 24:551–559, 1942.

Santiago Paternain, Miguel Calvo-Fullana, Luiz FO Chamon, and Alejandro Ribeiro. Safe policies for reinforcement learning via primal-dual methods. arxiv e-prints, art. *arXiv preprint arXiv:1911.09101*, 2019.

Stephen Prajna and Ali Jadbabaie. Safety verification of hybrid systems using barrier certificates. In *International workshop on hybrid systems: Computation and control*, pp. 477–492. Springer, 2004.

Harsh Satija, Philip Amortila, and Joelle Pineau. Constrained markov decision processes via backward value functions. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Haojie Shi and Max Q. H. Meng. Deep koopman operator with control for nonlinear systems, 2022. URL https://arxiv.org/abs/2202.08004.

Aivar Sootla, Alexander I Cowen-Rivers, Taher Jafferjee, Ziyan Wang, David H Mguni, Jun Wang, and Haitham Ammar. Sauté rl: Almost surely safe reinforcement learning using state augmentation. In *International Conference on Machine Learning*, pp. 20423–20443. PMLR, 2022a.

Aivar Sootla, Alexander I Cowen-Rivers, Jun Wang, and Haitham Bou Ammar. Effects of safety state augmentation on safe exploration. *arXiv preprint arXiv:2206.02675*, 2022b.

B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd. OSQP: an operator splitting solver for quadratic programs. *Mathematical Programming Computation*, 12(4):637–672, 2020. doi: 10.1007/s12532-020-00179-2. URL https://doi.org/10.1007/s12532-020-00179-2.

Xiao Tan, Wenceslao Shaw Cortez, and Dimos V. Dimarogonas. High-order barrier functions: Robustness, safety, and performance-critical control. *IEEE Transactions on Automatic Control*, 67 (6):3021–3028, 2022. doi: 10.1109/TAC.2021.3089639.

Kim P Wabersich and Melanie N Zeilinger. Linear model predictive safety certification for learning-based control. In *2018 IEEE Conference on Decision and Control (CDC)*, pp. 7130–7135. IEEE, 2018.

Akifumi Wachi, Wataru Hashimoto, Xun Shen, and Kazumune Hashimoto. Safe exploration in reinforcement learning: A generalized formulation and algorithms. *Advances in Neural Information Processing Systems*, 36:29252–29272, 2023.

Yuning Wang and He Zhu. *Safe Exploration in Reinforcement Learning by Reachability Analysis over Learned Models*, pp. 232–255. 07 2024. ISBN 978-3-031-65632-3. doi: 10.1007/978-3-031-65633-0_11.

Peter Wieland and Frank Allgöwer. Constructive safety using control barrier functions. *IFAC Proceedings Volumes*, 40(12):462–467, 2007.

Long Yang, Jiaming Ji, Juntao Dai, Linrui Zhang, Binbin Zhou, Pengfei Li, Yaodong Yang, and Gang Pan. Constrained update projection approach to safe policy optimization. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 9111–9124. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/3ba7560b4c3e66d760fbdd472cf4a5a9-Paper-Conference.pdf.

Tsung-Yen Yang, Justinian Rosca, Karthik Narasimhan, and Peter J Ramadge. Projection-based constrained policy optimization. *arXiv preprint arXiv:2010.03152*, 2020.

Linrui Zhang, Li Shen, Long Yang, Shixiang Chen, Xueqian Wang, Bo Yuan, and Dacheng Tao. Penalized proximal policy optimization for safe reinforcement learning. In Lud De Raedt (ed.), *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pp. 3744–3750. International Joint Conferences on Artificial Intelligence Organization, 7 2022. doi: 10.24963/ijcai.2022/520. URL https://doi.org/10.24963/ijcai.2022/520. Main Track.

Yiming Zhang, Quan Vuong, and Keith Ross. First order constrained optimization in policy space. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 15338–15349. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/af5d5ef24881f3c3049a7b9bfe74d58b-Paper.pdf.

He Zhu, Zikang Xiong, Stephen Magill, and Suresh Jagannathan. An inductive synthesis framework for verifiable reinforcement learning. In *ACM Conference on Programming Language Design and Implementation (SIGPLAN)*, 2019.

Vrushabh Zinage and Efstathios Bakolas. Neural koopman control barrier functions for safety-critical control of unknown nonlinear systems. *arXiv preprint arXiv:2209.07685*, 2022.

A APPENDIX

Algorithm 1 RAMPS

Input: Policy π, environment $\mathcal{E}$, initial datasets D, D_{val}, model-update period T_{model}, policy-update period T_π

Train initial linear model $\hat{F} = (A, B, c)$ on D; compute validation errors $\{\varepsilon_i\}$ on D_{val}
Set error bound $\varepsilon \leftarrow \text{quantile}(\{\varepsilon_i\}, q)$
Precompute matrices for all horizons up to $H_{\max}$ (matrix powers A^k, constraint templates $G(H)$, accumulators M_h)
loop
  $a_\pi \leftarrow \pi(s)$
  Observe lifted state z from s
  $u_{best} \leftarrow$ None; best_H $\leftarrow$ None
  $H_{\text{lo}} \leftarrow H_{\min}$; $H_{\text{hi}} \leftarrow H_{\max}$
  while $H_{\text{lo}} \le H_{\text{hi}}$ **do**
    $H_{\text{mid}} \leftarrow \lfloor (H_{\text{lo}} + H_{\text{hi}})/2 \rfloor$
    Build constraint matrices $G(H_{\text{mid}})$ and $h(H_{\text{mid}})$ using (A, B, c), ε, and precomputed $\{A^k\}$
    Solve $\text{QP}_{H_{\text{mid}}}$: $\min_{u_{0:H-1}} \|u_0 - a_\pi\|_2^2$
      s.t. $G(H_{\text{mid}})u \le h(H_{\text{mid}}), \quad u_k \in \mathcal{U}$
    if $\text{QP}_{H_{\text{mid}}}$ feasible **then**
      $u_{best} \leftarrow$ solution; best_H $\leftarrow H_{\text{mid}}$
      $H_{\text{lo}} \leftarrow H_{\text{mid}} + 1$
    else
      $H_{\text{hi}} \leftarrow H_{\text{mid}} - 1$
    end if
  end while
  if $u_{best} =$ None **then**
    Apply backup action $u \leftarrow u_{\text{backup}}(z)$
  else
    Apply shielded action $u \leftarrow u_{best}[0]$
  end if
  Execute u in environment, store transition in D
  Periodically refit (A, B, c) and recompute ε and QP precomputations (every T_{model} steps)
  Periodically update policy π from D (every T_π steps)
end loop

A.1 COMPUTATIONAL COMPLEXITY ANALYSIS

The computational efficiency of our shielding framework is critical for its real-time applicability. The architecture of our method is designed to front-load the most intensive computations into a single, state-independent pre-computation phase, leaving the per-timestep solve phase remarkably lightweight. We analyze the complexity of these two phases below, defining s as the state dimension, u as the action dimension, H as the maximum prediction horizon, and m as the number of faces in the safety polyhedron.

One-Time Pre-computation Cost. The computationally intensive construction of the QP's state-independent components is performed in a pre-computation phase, which is executed only when the underlying Koopman dynamics model is updated. This pre-computes and caches all components of the QP that are independent of the current state z_k. The construction of the QP constraint matrices dominates the complexity of this phase.

- **Matrix Power and Affine Term Pre-computation:** Calculating the powers of the state matrix A up to A^H requires $\mathbf{O}(\mathbf{H} \cdot \mathbf{s^3})$ operations. The cumulative affine terms are subsequently computed in $\mathbf{O}(\mathbf{H} \cdot \mathbf{s^2})$.
- **Constraint Matrix Construction:** The primary cost lies in constructing the matrices for the full-horizon QP. The constraint matrix G_{all} has dimensions $(N_c \times Hu)$, where the number of

constraints $N_c \leq mH$. The vectorized right-hand-side matrices, M_h and v_h, are constructed with a complexity of approximately $\mathbf{O}(\mathbf{m} \cdot \mathbf{H} \cdot \mathbf{s^2})$.

The dominant term arises from the matrix power calculation, making the total complexity of the pre-computation phase $\mathbf{O}(\mathbf{H} \cdot \mathbf{s^3} + \mathbf{m} \cdot \mathbf{H} \cdot \mathbf{s^2})$. This cost is incurred only once per model update, not at every control step.

Real-Time Solve Cost. The shield is executed at each timestep and is designed for high-frequency operation. Its complexity is significantly lower due to the extensive pre-computation.

- **State-Dependent Calculation:** The only significant state-dependent computation is the calculation of the right-hand-side vector h_{all}. Leveraging the pre-computed matrices, this is reduced to a single matrix-vector product, $h_{\text{all}} = M_h z_k + v_h$, which has a complexity of $\mathbf{O}(\mathbf{m} \cdot \mathbf{H} \cdot \mathbf{s})$.
- **Binary Search and QP Solution:** The binary search for the largest feasible horizon performs $O(\log H)$ iterations. Within each iteration, we update the QP's bounds and solve it. The 'update' operation is linear in the number of constraints, $O(N_c)$. Crucially, the 'solve' call is warm-started from the previous iteration's solution, making its average-case complexity, which we denote $T_{\text{qp\_warm}}$, substantially lower than solving from scratch.

Therefore, the total real-time complexity of the adaptive horizon selection is approximately $\mathbf{O}(\mathbf{m} \cdot \mathbf{H} \cdot \mathbf{s} + \log(\mathbf{H}) \cdot \mathbf{T}_{\text{qp\_warm}})$. This low polynomial complexity ensures that the shield can operate efficiently in real-time control loops.

A.2 Proof of Theorem 2

We restate the theorem for convenience:

Theorem. *Let $\epsilon_1, \ldots, \epsilon_N$ be a set of i.i.d. sampled model errors from our learned model $\hat{F}$. Assume that the probability of any two samples being equal is zero. Choose a quantile $0 < q < 1$ and let ε be the $\lceil qN \rceil$'th smallest value among $\epsilon_1, \ldots, \epsilon_N$. Then*

$$Pr[\|F(s_k, u_k) - \hat{F}(s_k, u_k)\|_\infty > \varepsilon] \leq 1 - q + \frac{1}{(2N)^{1/3}} + \frac{1}{4(2^{1/3})N^{2/3}}.$$

Proof. Let E be a random variable defining the errors in the learned model so that $\epsilon_1, \ldots, \epsilon_N$ are i.i.d. samples from E. Let ε be a conservative q-quantile of these errors (that is, $\varepsilon = \epsilon_{(\lceil qN \rceil)}$ where $\epsilon_{(i)}$ is the i'th order statistic of the sampled errors). Define a random variable X such that

$$X = \begin{cases} 1 & E > \epsilon \\ 0 & otherwise \end{cases}.$$

Then X is a Bernoulli random variable with success probability $P[E > \varepsilon]$ so that in particular $\mathbb{E}[X] = P[E > \varepsilon]$ and $\text{Var}[X] = P[E > \varepsilon]\,(1 - P[E > \varepsilon])$. We can now view our error samples $\epsilon_1, \ldots, \epsilon_n$ in terms of X. By construction, exactly $\lceil qN \rceil$ of the error samples are less than or equal to ε, so we compute the sample mean of X

$$\hat{\mu}_X = \frac{N - \lceil qN \rceil}{N} = \frac{\lfloor N - qN \rfloor}{N} \leq 1 - q.$$

Applying Chebyshev's inequality to $\hat{\mu}_X$, we find that for any positive c

$$P\left[|\hat{\mu}_X - P[E > \varepsilon]| \geq c\right] \leq \frac{P[E > \varepsilon]\,(1 - P[E > \varepsilon])}{Nc^2}$$

Notice that for all $0 \leq p \leq 1$ we have $p(1-p) \leq 1/4$ so that

$$P\left[|\hat{\mu}_X - P[E > \varepsilon]| \geq c\right] \leq \frac{1}{4Nc^2} \implies P\left[P[E < \varepsilon] - \hat{\mu}_X \geq c\right] \leq \frac{1}{4Nc^2}.$$

Plugging in, $\hat{\mu}_X \leq 1 - q$ we find

$$P\left[P\left[E > \varepsilon\right] \geq 1 - q + c\right] \leq \frac{1}{4Nc^2} \implies P[E > \varepsilon] \leq 1 - q + c + \frac{1}{4Nc^2}.$$

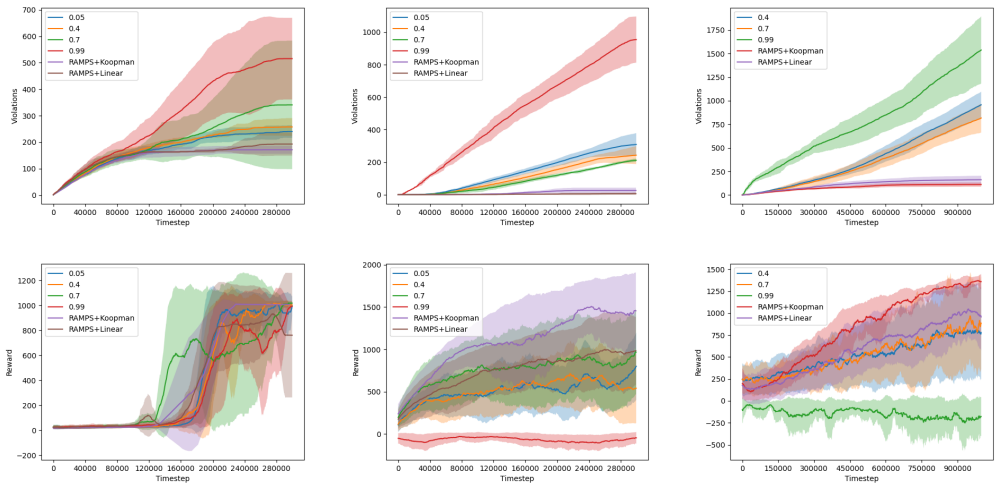
Since this bound holds for any positive c, we set $c = (2N)^{-1/3}$, which minimizes the value of the right-hand side. Plugging in this value of c, we find the bound

$$P[E > \varepsilon] \leq 1 - q + \frac{1}{(2N)^{1/3}} + \frac{1}{4(2^{1/3})N^{2/3}}.$$

□

A.3 Ablations

A.3.1 Exploring the effect of the Robustness term



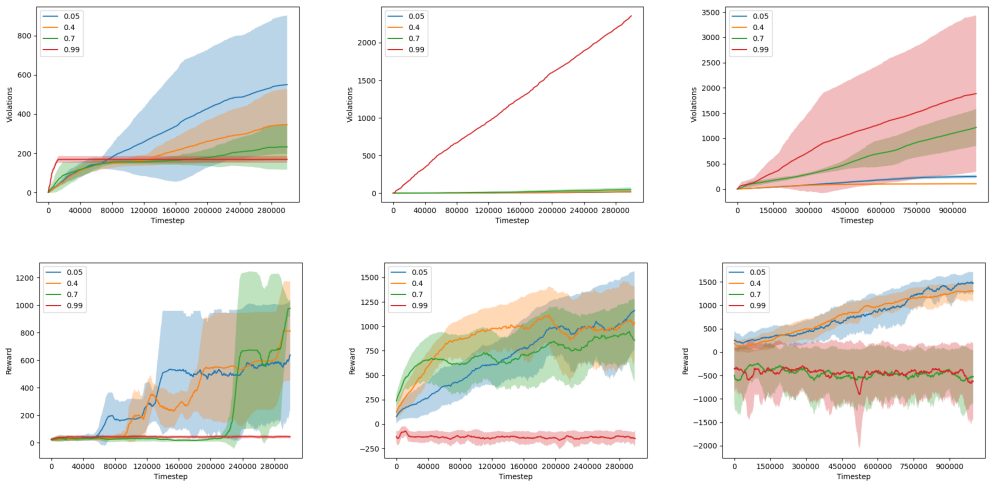
(a) Hopper (b) Cheetah (c) Ant

Figure 2: The critical role of the robustness term in ensuring safety. This figure compares the cumulative safety violations of the full RAMPS framework against a non-robust version that operates without the error-aware tightening term. While the non-robust shield continuously accumulates violations across all settings for the decay rate λ, the full RAMPS framework successfully learns to operate safely, evidenced by the flattening of its violation curves.

To isolate the contribution of our robust formulation, we performed a critical ablation study, presented in Figure 2. In this experiment, the shield operated without its error-aware tightening term. For direct comparison, each plot includes the performance of the full RAMPS framework using both the learned Koopman and a baseline Linear model.

While all non-robust configurations continuously accumulate safety violations, the full RAMPS framework's violation curve consistently flattens, demonstrating its ability to learn to operate safely. This failure of the ablated models occurs because they operate on an overly optimistic view of the dynamics; they consistently certify actions that are safe within their flawed model but lead to catastrophic failures in the physical system. This result provides definitive evidence that while the multi-step CBF is a necessary structure, the explicit robustness to model uncertainty is the essential component that enables our framework to achieve strong safety guarantees.

Furthermore, the reward curves reveal that ineffective shielding directly harms task performance. The most conservative non-robust setting ($\lambda = 0.99$), which also suffers from high violations, yields the worst reward, often collapsing to negative values. This occurs because its overly strict constraints lead to frequent QP infeasibility, forcing the agent to rely on a simple backup policy that is not designed for task progression. In contrast, the full RAMPS framework not only achieves the best safety but also learns the highest-performing reward policy. This demonstrates that the shield is not overly aggressive; rather, by being robust and minimally invasive, it creates a stable learning environment that allows the agent to safely explore and find a more optimal policy.



(a) Hopper (b) Cheetah (c) Ant

Figure 3: The impact of CBF conservatism (λ) on safety and performance. Moderately conservative decay rates (e.g., $\lambda = 0.4$, $\lambda = 0.7$) achieve the best balance of low safety violations and high task reward. The most conservative setting ($\lambda = 0.99$) paradoxically performs the worst due to frequent QP infeasibility.

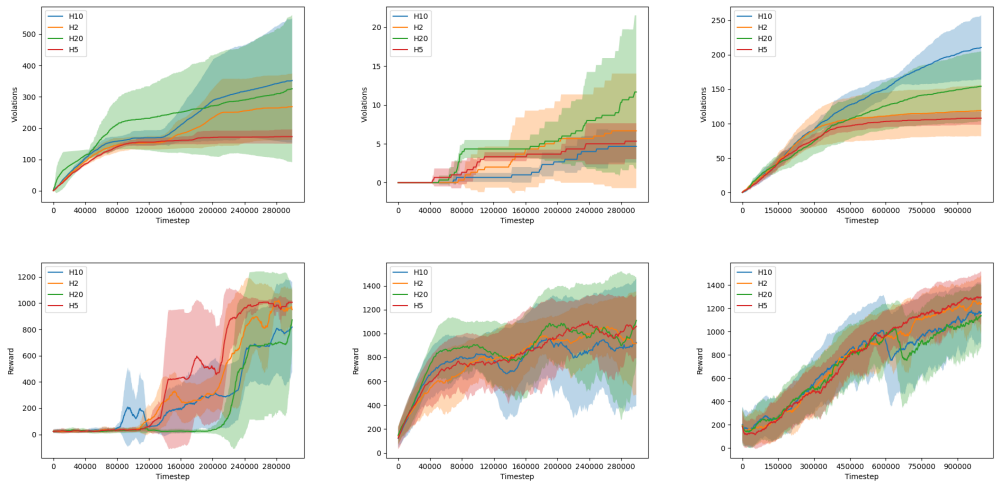
A.3.2 The Effect of CBF λ Conservatism

A key hyperparameter in our framework is the CBF decay rate λ, which dictates the conservatism of the shield. Figure 3 explores this parameter's effect on both safety and reward performance when using the full, robust RAMPS framework. The results reveal a critical trade-off between constraint strictness and the feasibility of the shielding problem, a trade-off that varies with the complexity of the environment dynamics.

In the **Hopper** environment, the most conservative setting ($\lambda = 0.99$) results in a *safe failure*; the lowest violation count but also near-zero reward. This occurs because the strict requirement to preserve 99% of the safety margin makes the QP problem frequently infeasible, forcing the agent to over-rely on a passive backup policy that prevents task progression. Conversely, a setting of $\lambda = 0.7$ achieves the best performance in both safety and reward, indicating that the learned model is sufficiently accurate to consistently find feasible solutions under this demanding safety requirement.

The **Cheetah** environment paints a similar but distinct picture. Here, the $\lambda = 0.99$ setting again results in the worst performance, but leads to high violations and low reward, suggesting that when the primary QP fails, the simple backup policy is insufficient to manage Cheetah's unstable dynamics. The best results are achieved with more permissive values ($\lambda \in \{0.05, 0.4\}$), suggesting that for more complex systems, the shield requires greater flexibility to ensure the underlying optimization problem remains feasible.

This trend is further emphasized in the highly unstable **Ant** environment. As with Cheetah, the $\lambda = 0.99$ setting is catastrophically poor in both safety and reward. A setting of $\lambda = 0.7$ is also too restrictive, hampering the agent's ability to learn. The best overall performance is achieved with $\lambda = 0.4$, which provides a strong safety guarantee while allowing enough flexibility for the agent to learn a high-reward policy. A highly permissive setting like $\lambda = 0.05$ enables good policy learning but at the cost of higher safety violations. Ultimately, these results show that λ is a critical tuning parameter, with the optimal value becoming more permissive as the inherent instability of the environment increases.



(a) Hopper (b) Cheetah (c) Ant

Figure 4: The trade-off between foresight and model reliability for the prediction horizon H. A moderate horizon ($H = 5$) achieves the best balance of low violations and high reward, avoiding both the myopic failures of a short horizon ($H = 2$) and the unreliable predictions of a long horizon ($H = 20$).

A.3.3 The Effect of Prediction Horizon

The selection of the prediction horizon H presents a critical trade-off between predictive foresight and the reliability of the learned dynamics model. Figure 4 explores this trade-off, revealing that the optimal choice for H is environment-dependent but follows a clear pattern.

A long horizon ($H = 20$) consistently leads to poor performance in both safety and reward. This occurs because the multi-step predictions of the Koopman model become increasingly inaccurate as errors compound over the extended rollout. The shield is forced to make decisions based on this unreliable information, leading to suboptimal or unsafe interventions.

Conversely, a short horizon ($H = 2$) is also suboptimal. While the model is accurate over this brief window, the limited lookahead is insufficient to resolve the high relative-degree traps present in the dynamics, a core challenge this paper aims to address. The shield becomes myopic, failing to prevent safety violations that are inevitable several steps in the future.

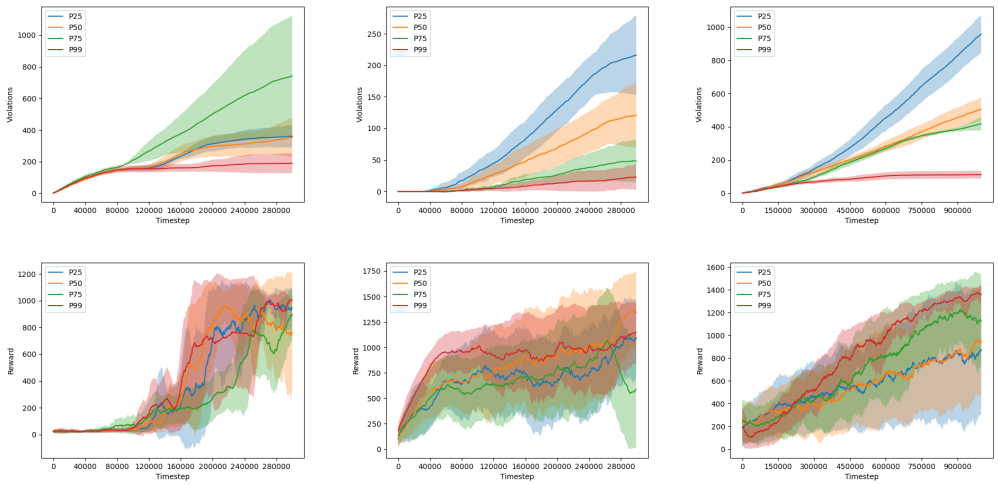
The results show that a moderate horizon ($H = 5$) provides the optimal balance. It is long enough to provide the necessary foresight to handle control delays and traps, yet short enough that the learned model's predictions remain reliable. This empirical finding validates our choice of $H = 5$ for the main experiments presented in this paper.

A.3.4 The Effect of Error Bound Confidence

The robust tightening term in our framework is calibrated using an error bound, ϵ, derived from a hold-out validation set. The confidence of this bound is a critical bhyperparameter, which we control by taking a percentile of the absolute one-step prediction errors. Figure 5 explores the impact of this choice on safety and reward.

The results show a clear and direct correlation between the confidence of the error bound and the safety of the resulting shield. A lower percentile (e.g., 25 or 50) provides an error bound that is too optimistic; it underestimates the true model error, leading to a high number of safety violations. As the confidence increases, the shield becomes more conservative and effective, with the 99th percentile (99) achieving the best safety performance by a significant margin.

Crucially, this improved safety directly enables better reward performance. By providing a more reliable and stable training environment, the 99th percentile configuration allows the RL agent to



(a) Hopper (b) Cheetah (c) Ant

Figure 5: The impact of error bound confidence (percentile) on performance. A higher percentile (99), corresponding to a more conservative error bound, achieves the best safety (lowest violations) and enables the agent to learn a higher-reward policy.

explore more effectively and learn a higher-reward policy. Less confident settings, while seemingly more permissive, lead to catastrophic failures that terminate episodes, ultimately harming the agent's ability to learn the task. This demonstrates that a sufficiently conservative, high-confidence error bound is not a hindrance to performance but is in fact a prerequisite for achieving both safety and high reward in complex environments.

A.3.5 Summary of Ablation Studies

Taken together, our ablation studies validate the core design principles of the RAMPS framework and illuminate the critical trade-offs inherent in learning-based predictive safety. The results consistently demonstrate that achieving robust safety is not a matter of maximizing any single parameter, but of finding a carefully calibrated balance between competing factors.

The most crucial finding is that explicit *robustness to model error is the essential component for safety*. As shown in Figure 2, removing the error-aware tightening term is catastrophic; the shield becomes overly optimistic and fails to prevent a continuous accumulation of violations, regardless of other hyperparameter settings. This confirms that the ability to reason about its own model's uncertainty is a prerequisite for the shield's success.

Building upon this robust foundation, the remaining hyperparameters tune the balance between safety and performance. The prediction horizon H must balance foresight against model reliability; a moderate horizon (Figure 4) is optimal, as short horizons are too myopic to handle control delays, while long horizons suffer from compounding prediction errors. Similarly, the CBF decay rate λ must balance constraint strictness against QP feasibility (Figure 3), where an overly conservative setting can paradoxically harm both safety and reward by causing frequent reliance on a simple backup policy. Finally, our analysis of the error bound confidence (Figure 5) resolves a key trade-off, showing that a more conservative, high-confidence error bound (99th percentile) does not hamper performance but instead enables it by creating a more stable learning environment. Collectively, these results show that RAMPS is a co-designed system where each component is critical for achieving both high performance and strong safety guarantees.

A.4 Active Recovery Backup Policy

In the rare event that the primary multi-step QP is infeasible, a deterministic backup policy is invoked. The use of such a policy is a standard practice in model-predictive shielding to ensure the agent

can always take an action. The active recovery approach detailed here is therefore **not a novel contribution of this work**, but rather follows a common pattern established in prior literature. Similar geometric recovery strategies are employed in other prominent shielding frameworks, such as SPICE Anderson et al. (2023), DMPS Banerjee et al. (2024), and MASE Wachi et al. (2023). For completeness, we describe our specific implementation of this widely-used technique below.

The policy, detailed in Algorithm 2, identifies the single most critical safety constraint, which is the one the agent is closest to violating; and solves a secondary, lightweight QP. The objective of this QP is to find a control action that maximally steers the system away from that constraint's boundary, leveraging the inward-pointing normal vector of the safe set.

Algorithm 2 Active Recovery Backup Policy (following e.g., Anderson et al. (2023); Banerjee et al. (2024))

Input: Current latent state z
Initialize: Minimum barrier value $h_{\min} \leftarrow \infty$, critical normal vector $p^* \leftarrow$ null
for all polyhedron face (p_i, b_i) in the definition of the safe set $\mathcal{C}$ **do**
  Compute barrier value $h_i(z) \leftarrow -(p_i^\top z + b_i)$
  if $h_i(z) < h_{\min}$ **then**
    $h_{\min} \leftarrow h_i(z)$
    $p^* \leftarrow p_i$
  end if
end for
if p^* is null **then**
  {This occurs only if the state is not within any defined polyhedron.}
  return $\mathbf{0}$
end if
Define QP cost vector $q \leftarrow (p^*)^\top B$
Solve the following QP for the recovery action u_{backup}:

$$\begin{aligned} \min_{u} \quad & q^\top u \\ \text{s.t.} \quad & u \in \mathcal{U} \quad \text{(action bounds)} \end{aligned}$$

return u_{backup} if QP is solved, else return $\mathbf{0}$.

Limitations of Backup Policy. This recovery strategy is designed for computational efficiency, which necessitates several trade-offs common to such backup controllers. First, it is **non-robust**, relying on the nominal dynamics matrix B without accounting for model error. Second, it is **myopic**, operating as a one-step greedy controller without the foresight of a multi-step planner. Third, it focuses on the **single most critical constraint**, which may be insufficient when multiple constraints are nearly violated. Despite these inherent limitations, it provides a more principled fallback than a simple passive policy.

A.5 Experimental Details

We evaluate our approach on five distinct environments, ranging from classic control tasks to more complex locomotion challenges, to demonstrate its efficacy across varying dimensionalities and dynamics.

Pendulum In this classic control benchmark, the goal is to swing up and stabilize an inverted pendulum. The environment has a 2-dimensional state space (encoding the pendulum's angle and angular velocity) and a 1-dimensional action space (torque). A state is considered unsafe if the pendulum's angle $|\theta|$ exceeds 0.4 radians. We allow all baselines 200000 environment interactions.

SafeHopper To test our method on more complex dynamics, we use the SafeHopper environment. This task involves controlling a two-legged robot, presenting a higher-dimensional challenge with an 11-dimensional state space and a 3-dimensional action space. Safety is defined by constraints on

the robot's velocity ($-0.37315 \leq v \leq 0.37315$.) to ensure stable hopping. We allow all baselines 300000 environment interactions.

SafeCheetah The SafeCheetah environment is another complex benchmark. The agent must control a planar cheetah-like robot to run forward. This environment features a 17-dimensional state space and a 6-dimensional action space. Similar to SafeHopper, the safety specifications impose constraints on the robot's velocity ($-2.8795 \leq v \leq 2.8795$.) to prevent unstable or dangerous movements. We allow all baselines 300000 environment interactions.

SafeAnt We further increase the complexity with the SafeAnt environment, which involves controlling a quadrupedal robot. The agent must learn to walk forward in a high-dimensional state space of 105 dimensions, with an 8-dimensional action space. Safety is defined by constraints on the robot's velocity ($-2.3475 \leq v \leq 2.3475$.) We allow all baselines 1000000 environment interactions.

SafeHumanoid We also test RAMPS on SafeHumanoid, a highly challenging benchmark with a 348-dimensional state space and 17-dimensional action space. The agent must learn to coordinate full-body locomotion while remaining within prescribed safety limits, defined by velocity constraints ($-2.3475 \leq v \leq 2.3475$). As with the other environments, each baseline receives 1000000 environment interactions. Due to its dimensionality and instability, SafeHumanoid is known to be difficult for safe-RL algorithms, making it a strong stress test for both the learned dynamics and the shielding mechanism.

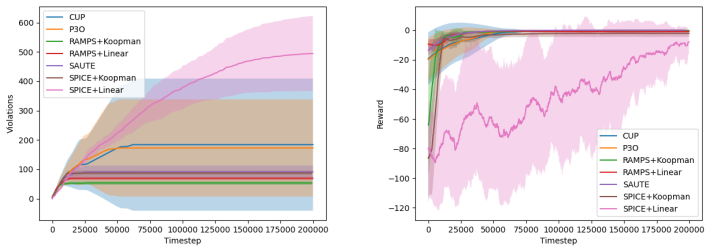


Figure 6: Safety violations (top) and episodic reward (bottom) for Pendulum.

Pendulum is a low-dimensional, easy-to-control system where safety constraints and dynamics pose minimal difficulty. As expected, all methods achieve near-perfect safety and high reward. We include the results here for completeness; the main paper focuses on higher-dimensional domains where RAMPS' multi-step shielding and scalable model-based prediction are essential.

A.5.1 Deep Koopman Operator Implementation

For all experiments involving a learned latent dynamics model (RAMPS + K and SPICE + K), we use the Deep Koopman Operator framework introduced by Shi & Meng Shi & Meng (2022). The central idea of this approach is to represent the nonlinear system dynamics with a linear model in a suitably constructed higher-dimensional *lifted* space. This is achieved by augmenting the original state with learned features that allow the dynamics to evolve linearly in the expanded coordinate system.

Lifted State Representation. Let x_k denote the original system state at time k. A deep encoder network $g_\theta(\cdot)$ generates additional latent coordinates, producing the lifted state

$$z_k = \begin{bmatrix} x_k \\ g_\theta(x_k) \end{bmatrix}.$$

In this lifted space, the system evolution is modeled by a linear dynamical system:

$$z_{k+1} = \mathbf{A} z_k + \mathbf{B} u_k,$$

where u_k is the control input and the matrices $\mathbf{A}$ and $\mathbf{B}$ are learned jointly with the encoder.

Network Architecture. The encoder is realized as a three-layer multilayer perceptron (MLP) with two hidden layers of 256 units each. Hidden layers use SiLU activations and the final layer uses a Tanh activation. The dimensionality of the lifted latent features is environment-specific, chosen to balance model accuracy and computational tractability: we use 2 for Pendulum, 22 for SafeHopper, 34 for SafeCheetah, and 100 for both SafeAnt and SafeHumanoid. These dimensions were selected empirically based on prediction accuracy and shield feasibility profiles.

Training Objective. The Koopman model is trained end-to-end using the loss function proposed by Shi & Meng Shi & Meng (2022), consisting of two complementary components:

1. **State Prediction Loss:** Measures the discrepancy between the true next state x_{k+1} and the predicted next state extracted from $z'_{k+1} = \mathbf{A}z_k + \mathbf{B}u_k$. This encourages accurate one-step predictions in the original state space.
2. **Linearity Loss:** Enforces consistency between the learned feature representation and linear evolution under the Koopman operator. Specifically, it penalizes the difference between the predicted lifted features $g_\theta(x'_{k+1})$ and the lifted features of the true next state $g_\theta(x_{k+1})$.

Together, these losses encourage g_θ to discover a set of basis functions that linearize the system dynamics, effectively serving as a global regularizer and promoting long-horizon predictive stability.

Constraint Handling in the Lifted Space. Safety constraints are defined in the original state space via polyhedral sets. Rather than imposing constraints directly on the learned latent coordinates, we preserve the original safety specifications by *zero-padding* the additional Koopman features. That is, the lifted constraint set is constructed by embedding the original constraint polytope into the higher-dimensional space as:

$$\mathcal{C}_{\text{lifted}} = \mathcal{C} \times \{0\}^{d_{\text{latent}}}.$$

This approach ensures that safety checks and subsequent CBF/QP computations remain well-defined without introducing arbitrary or unverified restrictions on the latent variables. While conservative, this choice has two benefits: (i) it guarantees compatibility with the underlying safety definitions, and (ii) it avoids imposing structural constraints on the learned features whose semantics are not directly interpretable. Investigating learned or data-driven safety projections in the latent space is an interesting direction for future work.

Summary. In combination, these design choices allow the Deep Koopman Operator to model high-dimensional, nonlinear systems with a compact linear approximation in the lifted space, enabling fast multi-step predictions and efficient computation of safety certificates needed for RAMPS and SPICE. Despite its simplicity, the model is sufficiently expressive for the complex locomotion domains we consider, while maintaining the computational tractability required for real-time shielding.

A.5.2 Error Bound Calibration

The dynamics model and its corresponding error bound, ϵ, are periodically recalibrated throughout training to adapt to newly collected data. An initial model is trained after the first 10,000 environment steps. Subsequently, the model is fine-tuned at progressively doubling intervals (e.g., at 20,000, 40,000, and 80,000 steps). At each training stage, the error bound is calibrated using a hold-out validation set comprising 20% of all collected experience. To maintain computational efficiency, the size of this validation set is capped at a maximum of 100,000 samples.

A.5.3 Analysis of Baseline Failures

As noted in Table 1, the SPICE+L baseline failed on all high-dimensional locomotion tasks (**SafeHopper**, **SafeCheetah**, and **SafeAnt**). This is attributable to the representational limits of a simple linear model in capturing the complex, nonlinear dynamics of these environments. The resulting model exhibited large prediction errors, which, within SPICE's one-step shielding formulation, rendered the safety QP persistently infeasible. This forced the agent to over-rely on its backup policy, preventing it from learning the task.

More critically, **SPICE+K** also failed on the most complex environment, **SafeAnt**. This failure occurred even when using the *exact same* pre-trained Deep Koopman Operator that was successful for

our method, RAMPS + K. This finding isolates the failure to SPICE's underlying shielding technique. Its myopic, one-step approach is not sufficiently robust to the larger, yet unavoidable, prediction errors of a learned model in such a high-dimensional space. In contrast, our multi-step formulation is explicitly designed to tolerate these errors, explaining the significant performance difference.

We note explicitly that **DMPS** (Banerjee et al., 2024), **VELM** (Wang & Zhu, 2024) and **MASE** (Wachi et al., 2023) are model-predictive shielding (MPS) techniques which operate using a safety predicate or safe/unsafe-state specification rather than relying on dense cost or reward shaping. As such they are directly related to SPICE and RAMPS in design intent: all attempt to find safe actions via online planning under a safety specification. Despite operating with this stronger safety interface, the publicly available implementations of DMPS and VELM, and our re-implementation of MASE (Wachi et al., 2023), exhibited rapid violation growth and frequent infeasible/timeout planner returns on the MuJoCo locomotion benchmarks (Hopper, Cheetah, Ant), accumulating `>1000` violations in the first 20–30k environment interactions. We also tested the **Conservative Safety Critics** approach (Bharadhwaj et al., 2021a), which similarly failed to train stably in these setting. Because these failure modes made them impractical and unstable as baselines for our main comparisons, we exclude them from the final baseline table. For transparency and validation, we will release all code and scripts for these baselines along with our framework implementation.

A.5.4 Analysis of Computational Efficiency

Environment	RAMPS	SPICE
Pendulum	0.2289 ± 0.01	0.4061 ± 0.0063
Hopper	0.2822 ± 0.04	0.5996 ± 0.0624
Cheetah	0.3234 ± 0.03	0.5244 ± 0.0418
Ant	0.4038 ± 0.08	2.4820 ± 1.8329
Humanoid	0.5061 ± 0.02	3.7105 ± 1.2512

Table 2: Shield Computation Time Analysis. This table shows the mean and standard deviation of the per-step execution time of the safety shield (Policy action proposal, state space lifting via Koopman, constraint assembly and QP solving) across all training episodes for each environment. The consistently low average times (all under 0.5 ms) confirm the real-time feasibility of the RAMPS framework.

The primary design goal of the RAMPS framework is to make predictive shielding computationally tractable for real-time applications. The timing results in Table 2 confirm the success of this approach. Across all tested environments, the mean per-step computation time for the shield is remarkably low, remaining well under half a millisecond.

Notably, the computation time scales gracefully with the complexity of the environment. For the simple **Pendulum** environment, the mean solve time is just 0.2289 ms. For the high-dimensional and dynamically complex **Ant** environment, this time increases to only 0.4038 ms. This sub-millisecond performance demonstrates that the extensive pre-computation phase is effective, leaving the online QP solve lightweight and suitable for high-frequency control loops. Furthermore, the low standard deviation across all environments indicates that the solver's performance is consistent and predictable, a critical feature for reliable real-time systems. We use OSQP Stellato et al. (2020) for solving the QP.

A.5.5 Analysis of Action Type Distribution

Table 3 details the ratio of actions selected by the primary shield, the original RL agent's policy (Neural), and the fallback backup policy. This analysis reveals how the shield's behavior adapts to the complexity of the environment.

The shield is the dominant actor in all environments, indicating its critical role in maintaining safety. This is most pronounced in the highly unstable **Ant** environment, where the shield intervenes in over 96% of steps, indicating that the RL agent rarely proposes a provably safe action on its own. It is critical to note that this high intervention rate does not prevent the agent from learning a high-reward policy. This is a direct result of the shield's minimally invasive objective function, which finds the

ENVIRONMENT	SHIELD RATIO (%)	NEURAL RATIO (%)	BACKUP RATIO (%)
PENDULUM	74.72 ± 31.48	25.28 ± 31.48	0.00 ± 0.00
HOPPER	82.06 ± 4.11	17.08 ± 3.79	0.86 ± 0.93
CHEETAH	81.13 ± 4.55	17.27 ± 4.25	1.60 ± 0.59
ANT	96.50 ± 1.39	2.45 ± 1.25	1.05 ± 0.81
HUMANOID	96.28 ± 1.39	3.81 ± 0.81	0.00 ± 0.00

Table 3: This table shows the per-episode average ratio of actions selected by the Shield, the original RL agent (Neural), and the Backup policy for results in Figure 1 and Table 1. The results show that the shield is highly active but allows the agent more freedom in less complex environments. The extremely low reliance on the Backup policy across all environments confirms the robustness and high feasibility rate of the primary multi-step QP shield.

closest possible safe action to the agent's original proposal. Consequently, many interventions are slight corrections that nudge the agent back towards safety without fundamentally disrupting its learned behavior.

In the moderately complex **Cheetah** and **Hopper** environments, the shield remains the primary actor but is significantly less invasive, allowing the agent's neural policy to act directly approximately 17% of the time. This suggests that for these dynamics, the RL agent is better able to learn a policy that aligns with the safety constraints. The **Pendulum** environment shows the most interesting behavior; while the shield is active for 75% of the steps on average, the extremely high standard deviation (31.48%) suggests a bimodal behavior where the agent learns to operate safely for long periods before requiring periods of heavy intervention.

Finally, a crucial indicator of the primary shield's robustness is the extremely low reliance on the backup policy. For all locomotion tasks, the backup policy is invoked only 1% of the time, and for Pendulum, it is never used at all. This demonstrates that the multi-step, adaptive-horizon QP is consistently able to find a feasible, provably safe solution, rarely needing to resort to its simpler fallback mechanism.

A.6 ANALYSIS OF SHIELD INTERVENTION

To better understand the trade-off between reward maximization and safety interventions, we analyze the average per-step action deviation ($\|u_{\text{shielded}} - u_{\text{agent}}\|$) during training in the Cheetah environment (Fig. 7).

We observe that **RAMPS+L** (linear model) produces consistently higher action deviations, indicating that the shield intervenes more aggressively to maintain safety. This stronger intervention translates into more reliable shielding performance, but it also limits the agent's ability to explore freely, resulting in lower asymptotic reward.

By contrast, **RAMPS+K** (Koopman model) exhibits substantially smaller action deviations throughout training. This reduced level of intervention reflects the shield's greater *invasiveness efficiency*: the agent is allowed to execute its intended actions more faithfully, leading to improved reward performance while still respecting safety constraints. In other words, RAMPS+K achieves a better balance between enforcing safety and preserving the agent's autonomy.

A.7 ANALYSIS OF MULTI-DIMENSIONAL CONSTRAINTS ON HUMANOID

To evaluate RAMPS under realistic multi-dimensional safety conditions, we conducted an additional experiment on SAFEHUMANOID with a 348-dimensional state space and imposed a 21-dimensional polyhedral safety constraint set. Specifically, we constrained all coordinate velocities (state indices 23–25) to lie in $[-2.3475,\ 2.3475]$ and all angular velocities (state indices 28–45) to lie in $[-20,\ 20]$. Figure 8b shows that RAMPS+SAC achieves high reward (approximately 5000) and continues improving throughout training, while CMDP baselines (P3O, CUP, PPO-Saute) plateau early and fail to make progress.

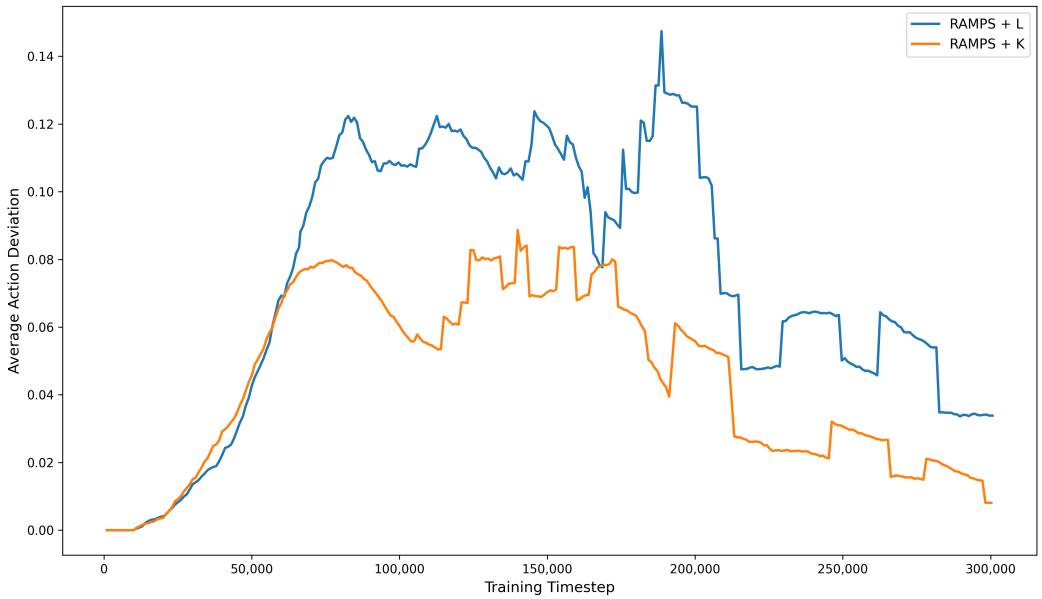
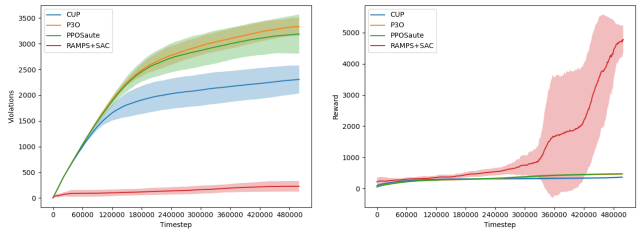


Figure 7: Average per-step action deviation ($\|u_{\text{shielded}} - u_{\text{agent}}\|$) during training on the Cheetah environment. Both variants show increasing intervention early in training as the agent explores unsafe behaviors, with RAMPS+K consistently yielding smaller deviations (less invasive interventions) and decaying sooner as the policy and model improve.



(a) Cumulative safety violations . (b) Episode reward .

Figure 8: RAMPS evaluation on Humanoid with constraints on 21 dimensions (3 coordinate velocity constraints and 18 angular velocity constraints): (a) cumulative safety violations over training; (b) episode reward over training.

Figure 8a also demonstrates the safety behavior: RAMPS accumulates only about 256 cumulative violations over 500,000 steps, and reducing violations to 0 after 400,000 steps. CMDP baselines never learn to be safe, as evidenced by their increasing violation curves. These results confirm that RAMPS scales effectively to high-dimensional, coupled safety constraints and maintains both strong safety and high performance in settings where CMDP methods fail completely.

A.8 Limitations and Future Work

Our framework operates under a common assumption in online learning: no a priori model of the environment is available. Consequently, the initial policy, π_0, begins exploring without any prior knowledge of the environment, which can lead to safety violations while the dynamics model is being learned. These have also been reported in SPICE Anderson et al. (2023), VELM Wang & Zhu (2024) and DMPS Banerjee et al. (2024). A promising direction for future work is to mitigate these *cold start* violations by pre-training the Koopman model on relevant offline datasets, thereby enabling a safer initial policy. Furthermore, while our method provides a high-confidence probabilistic safety certificate, it does not offer a hard, worst-case guarantee on the number of violations. Future research

could focus on bridging this gap by employing formal verification techniques, such as abstract interpretation, to compute a rigorous upper bound on the number of potential safety violations throughout the learning process.