

Evaluation Metrics for Safe Reinforcement Learning

Lindsay Spoor^{*†}

Aske Plaat[†]

Thomas Moerland[†]

Abstract

Safe reinforcement learning (RL) is commonly formalized as a Constrained Markov Decision Process (CMDP), in which an agent maximizes expected reward while keeping its expected cumulative cost below a specified safety bound. Existing safe RL benchmarks predominantly report whether an algorithm is safe on average, following this expectation-based guarantee. We argue that this convention is insufficient to reliably characterize an algorithm’s true safety: it fails to capture how often and how severely the safety bound is violated, whether this holds consistently across tasks and safety bounds, and whether training-time behavior is representative of behavior of the final converged policy. Therefore, we introduce (i) *evaluation metrics* for safe RL that address each of these concerns and in addition allow for aggregation across tasks and safety bounds. We furthermore define (ii) a *safety tier system* to systematically categorize and compare algorithms in terms of safety and reliability at both training and for a final policy. Using this framework, we provide (iii) an *empirical safety evaluation* across multiple safety navigation tasks. Our results show that aggregate metrics, distributional reporting, and task- and safety bound-specific results each reveal information the other metrics cannot. We therefore recommend reporting all three jointly, rather than compressing this information into a single value, as is common practice. We provide *SafeRLEval*³, an open-source evaluation suite to support the reliable characterization of safety in future safe RL research.

1 Introduction

Reinforcement learning (RL) addresses sequential decision-making problems by enabling agents to learn from feedback in the form of rewards, with the goal of maximizing their long-term cumulative reward (Sutton et al., 2018). Despite their success on tasks without critical safety concerns (Mnih et al., 2015; Schulman et al., 2017; Haarnoja et al., 2018), agents deployed in safety-critical domains, such as autonomous driving, robotics, and power systems (Ramanujam et al., 2025), must often satisfy restrictive constraints on their behavior, in addition to maximizing task performance.

Safe RL provides a framework for such safety-critical tasks, in which the learning objective is extended to explicitly incorporate constraints. This is commonly formalized as a Constrained Markov Decision Process (CMDP) (Altman, 1999), in which the agent’s objective is to maximize expected reward subject to $\mathbb{E}[C] \leq d$, where C denotes the cumulative episodic cost and d the safety bound. Safe RL has been extensively studied over the last decade, leading to a variety of approaches (Garcia and Fernández, 2015), including Lagrangian (primal-dual) methods (Ray et al., 2019; Tessler et al., 2019; Stooke et al., 2020), trust-region methods (Zhang et al., 2020; Achiam et al., 2017), and increasing cost penalty coefficients in the objective (Zhang et al., 2022).

^{*}corresponding author: l.j.spoor@liacs.leidenuniv.nl

[†]Leiden Institute of Advanced Computer Science, Leiden University, Leiden, the Netherlands

³<https://github.com/lindsayspoor/SafeRLEval>

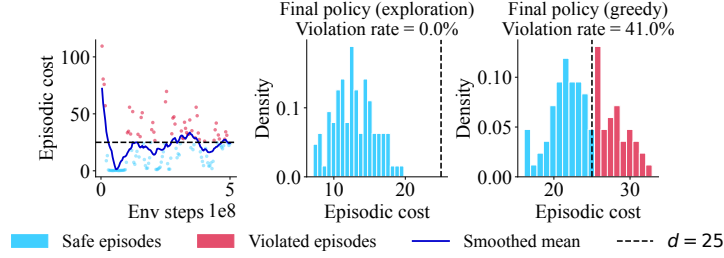


Figure 1: An example illustrating that the formal guarantee $\mathbb{E}[C] \leq d$ in the CMDP framework falls short of reliably ensuring safety, from a single-seed run of Lagrangian-based PPO, task: Safe Circle Point (Level 1), $d = 25$ (Tomilin et al., 2026). (left) The Monte Carlo estimate of $\mathbb{E}[C]$ is shown by the smoothed mean. All individual data points show single-episode costs. Even though the final policy with exploration noise shows that the full distribution of episodes lies below d (middle), greedy policy rollouts of the final policy show a distribution with a substantially higher violation rate (right).

Existing safe RL benchmarks generally report whether an algorithm is safe *on average* (Ray et al., 2019; Ji et al., 2024). This convention follows directly from the CMDP formulation itself, with the standard formal guarantee, $\mathbb{E}[C] \leq d$, being expectation-based. Ray et al. (2019) proposed an evaluation protocol that distinguishes average performance during training from that of the final policy, alongside an average training regret capturing cumulative unsafe behavior incurred while learning. However, this protocol does not indicate how often, or how severely, a policy violates its constraints, nor does it distinguish between evaluating the final policy with exploration noise enabled and evaluating it as a greedy policy, i.e., with exploration noise disabled. Even though more recent benchmark suites (Ji et al., 2023, 2024; Tomilin et al., 2026; Ramanujam et al., 2025) have substantially expanded the range of available tasks, algorithms, and constraint types, the standard evaluation protocol has changed comparatively little. We argue that an expectation-based guarantee, with a measure of safety collapsed into a single aggregate value, remains insufficient to reliably characterize a method’s true safety.

Additionally, behavior observed during training is often subject to, and optimized under, exploration noise. However, the performance can differ for the behavior of a final, converged policy evaluated without exploration noise, since such a policy is commonly deployed deterministically in simulation. Figure 1 illustrates this on a representative single-seed training run of Lagrangian PPO (Tomilin et al., 2026): although all evaluated episodes of the converged final policy lie below the bound d when exploration noise is enabled, a substantial fraction of episodes violate d when the same policy is instead evaluated deterministically, showing a distributional shift between the two settings that, in this case, results in an even higher violation rate. Such shifts can occur because Lagrangian methods only constrain the expected cost under the stochastic policy used during training, while the deterministic policy, which simply takes the most likely action at each step, was never directly subject to this constraint.

This motivates the need to unpack the aggregate value of safety, typically reported as cost on average, into more informative measures, such as violation rate, violation magnitude, and cost deviation from the safety bound, both with and without exploration noise in the final policy. Moreover, current evaluation practices in safe RL have yet to adopt statistically rigorous aggregation of performance across a range of tasks, as well as across a range of safety bounds (Spoor et al., 2025). Agarwal et al. (2021) showed in the general RL literature that reporting single point estimates of performance, without accounting for variability across runs and tasks, can produce misleading comparisons, and proposed more statistically robust aggregate measures. Such aggregation across tasks, and in particular across safety bounds, remains largely absent from the safe RL literature. Moreover, Safe RL yet has to adopt an evaluation protocol that jointly reports these distributional properties alongside the single-value metrics of evaluated algorithms.

We therefore propose the following contributions:

1. **Evaluation metrics for safe RL** We introduce evaluation metrics for safe RL that go beyond reporting average reward and cost alone, focusing on violation frequency, violation magnitude, and cost deviation, each normalized to allow for aggregation across both tasks

and safety bounds. We provide an open-source evaluation suite at <https://github.com/lindsayspoor/SafeRLEval>.

2. **Safety tier system** We define a set of safety tiers, based on conditions tied to the proposed evaluation metrics, that allow existing safe RL algorithms to be systematically categorized and compared by their safety and reliability, both for any policy during training and for the final converged policy, rather than by average performance alone.
3. **Empirical safety evaluation** We provide a robust empirical evaluation across a variety of tasks, multiple safety bounds, a variety of popular safe RL algorithms, and 30 seeds. Alongside the standard practice of reporting reward and cost training curves, our evaluation additionally reports the episodic cost distribution over the entire course of training, for the final policy under exploration noise, and for the final greedy policy, both for individual tasks and safety bounds and aggregated across all such conditions.

2 Safe reinforcement learning

Safe RL commonly uses the formal framework of a Constrained Markov Decision Process (CMDP) (Altman, 1999), an extension of the Markov Decision Process (MDP) (Sutton et al., 2018; Bellman, 1957), in which we not only want to maximize desired behavior, but also want to minimize undesired, unsafe behavior. A CMDP is defined as the tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R, P_0, \gamma, \mathcal{C})$, where $\mathcal{S}$ and $\mathcal{A}$ are the set of all possible states and actions, respectively, P is the transition dynamics distribution $P : \mathcal{S} \times \mathcal{A} \rightarrow \Delta(\mathcal{S})$, R is the reward function $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$, $P_0 \in \Delta(\mathcal{S})$ is the initial state distribution, and $\gamma \in [0, 1)$ is the discount factor. A set of m cost functions $\mathcal{C} := \{C_1, \dots, C_m\}$ with associated safety bounds $\mathcal{D} := \{d_1, \dots, d_m\}$ maps transition tuples to costs, $\mathcal{C} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}^m$. Actions are selected from a policy $\pi : \mathcal{S} \rightarrow \Delta(\mathcal{A})$, $s \mapsto \pi(\cdot|s)$. For a trajectory $\tau = (s_t, a_t, s_{t+1}, \dots)$, the discounted return and discounted cost are given by $R(\tau) = \sum_{t=0}^T \gamma^t R(s_t, a_t, s_{t+1})$ and $C_i(\tau) = \sum_{t=0}^T \gamma^t C_i(s_t, a_t, s_{t+1})$, respectively.

The objective of a CMDP is to find a policy that maximizes expected discounted return while keeping each expected discounted cost within its safety bound:

$$\begin{aligned} \max_{\pi \in \Pi} \quad & \mathbb{E}_{\tau \sim \pi, P} [R(\tau)] \\ \text{s.t.} \quad & \mathbb{E}_{\tau \sim \pi, P} [C_i(\tau)] \leq d_i, \quad i = 1, \dots, m, \end{aligned} \tag{1}$$

where Π denotes the set of policies. Unlike standard MDPs (Bellman, 1957), policy search under a CMDP must additionally ensure that each policy update remains feasible with respect to its constraints, i.e., that it stays within the feasible set $\Pi_{\mathcal{C}} := \{\pi \in \Pi : \mathbb{E}_{\tau \sim \pi} [C_i(\tau)] \leq d_i \forall i\}$, rather than optimizing over Π alone. For a single cost function ($m = 1$), we write this constraint as $\mathbb{E}[C] \leq d$, with C the cumulative episodic cost and d the safety bound.

In this study, we evaluate several widely used safe RL algorithms, introduced below. A large family of safe RL algorithms address the constrained optimization problem in Eq. 1 by modifying the underlying objective function itself. Lagrangian relaxation (Ray et al., 2019; Tessler et al., 2019; Stooke et al., 2020) penalizes the reward objective in proportion to the constraint violation, with the Lagrange multiplier learned jointly with the policy via dual gradient ascent. Under convexity conditions, this guarantees feasibility only asymptotically, offering no guarantee for any individual policy iterate. P3O (Zhang et al., 2022) similarly penalizes the objective, but instead progressively increases a fixed-form penalty coefficient in response to observed violations, avoiding the instability of a learned Lagrange multiplier at the loss of a formal convergence guarantee. An alternative family of methods instead restricts the policy update itself. FOCOPS (Zhang et al., 2020) solves this constrained update via a first-order approximation. This approach, in approximation, enforces feasibility at every policy iteration.

3 Evaluation metrics

We first characterize an algorithm’s safety within a single fixed condition: a single task m , safety bound d , algorithm a , and seed k . Each of the introduced metrics can be used to evaluate a fixed policy π_j at policy iterate $j \in \{0, \dots, J\}$, where J denotes the final policy iterate, resulting in the converged policy π_{final} . Instance i denotes a single evaluation episode out of N episodes evaluated

under policy π_j , with N episodes forming a Monte Carlo estimate of the expectation in the CMDP definition from Eq. 1. We then describe how these metrics are aggregated across all policy iterates, tasks, and safety bounds, and furthermore introduce a safety tier system based on the resulting metrics, enabling a generalized comparison of algorithms.

Exploration noise Exploration noise is an inherent part of the training process itself, making it the relevant condition for characterizing safety while the policy is still being learned, whereas a converged policy is typically deployed deterministically in practice, making the greedy evaluation the relevant condition for characterizing deployment-time safety. All of our proposed metrics can be evaluated under two distinct policy conditions: with exploration noise enabled, or with a greedy policy, with exploration noise disabled. Comparing them directly isolates the effect that exploration noise itself has on the safety of an algorithm. Throughout this paper, we report metrics evaluated with exploration noise enabled when characterizing behavior over the course of training, and metrics evaluated under both a greedy and an exploration-based policy when characterizing the final policy, for direct comparison.

3.1 Metrics for a fixed policy

As a standard performance metric already reported by prior benchmarks, the **average reward and cost** are described as in Eq. 2.

$$\bar{R}_{(m,d,a,k)}^j = \frac{1}{N} \sum_{i=1}^N r_{i,(m,d,a,k)}, \quad \bar{C}_{(m,d,a,k)}^j = \frac{1}{N} \sum_{i=1}^N c_{i,(m,d,a,k)}, \quad (2)$$

where $r_{i,(m,d,a,k)}$ and $c_{i,(m,d,a,k)}$ are the reward and cost of rollout episode i under policy iterate j , for seed $k \in \{1, \dots, K\}$, task m , safety bound d , and algorithm a .

We propose the **violation rate** as a metric that indicates how often an algorithm violates constraints, which is the fraction of episodes on task m with bound d in which algorithm a exceeded the safety bound d , as described by Eq. 3.

$$V_{(m,d,a,k)}^j = \frac{1}{N} \sum_{i=1}^N \mathbb{1}[c_{i,(m,d,a,k)} > d], \quad (3)$$

where $\mathbb{1}[\cdot]$ is the indicator function.

Additionally, we introduce the **mean normalized cost deviation** as the metric that captures how far algorithm a 's average cost sits above (> 0 , unsafe) or below (< 0 , safe) the bound, scaled by the bound itself, as in Eq. 4.

$$D_{\text{norm},(m,d,a,k)}^j = \frac{\bar{C}_{(m,d,a,k)}^j - d}{d}. \quad (4)$$

Unlike Ray et al. (2019), who normalize deviation from the bound relative to a naive, unconstrained baseline algorithm like PPO (Schulman et al., 2017), we normalize directly by the safety bound d itself. Because an unconstrained baseline typically incurs costs far above d , the deviation term would be large, and the normalized scores of safe RL algorithms would vanish. We instead treat d as the natural unit of measurement for cost deviation, centering D_{norm} around 0 for $\bar{C} = d$.

Among the episodes in which algorithm a violated safety bound d , the **normalized violation magnitude** then captures how much they overshot the bound on average, relative to the bound itself, following Eq. 5.

$$D_{\text{norm}^+, (m,d,a,k)}^j = \begin{cases} \frac{1}{N_{a,k}^+} \sum_{i=1}^N (c_{i,(m,d,a,k)} - d)^+ & \text{if } N_{a,k}^+ > 0, \\ 0 & \text{if } N_{a,k}^+ = 0, \end{cases} \quad (5)$$

where $N_{a,k}^+ = \sum_{i=1}^N \mathbb{1}[c_{i,(m,d,a,k)} > d]$ is the number of violating rollout episodes for algorithm a under seed k , and $(x)^+ := \max(x, 0)$ denotes the positive part of x .

Along with D_{norm} from Eq. 4, which reflects how far the algorithm's cost sits from the bound on average, over all episodes, D_{norm^+} in Eq. 5 is restricted to the set of violating episodes only. This

distinction matters because two algorithms with an identical average cost can exhibit very different violation profiles. An algorithm may violate the bound frequently but only slightly on each occasion, or violate rarely but catastrophically when it does.

3.2 Aggregation across policy iterates

Each metric $X_{(m,d,a,k)}^j \in \{\bar{R}_{(m,d,a,k)}^j, \bar{C}_{(m,d,a,k)}^j, V_{(m,d,a,k)}^j, D_{\text{norm},(m,d,a,k)}^j, D_{\text{norm}^+, (m,d,a,k)}^j\}$ is defined for a fixed policy π_j at policy iterate j . To evaluate an algorithm’s performance across the entire course of training, i.e., aggregated over all policy iterates, we take the average across all iterates, as shown in Eq. 6. Since this is meant to capture behavior during the training process, we evaluate policy π_j with exploration noise enabled for this aggregation.

$$X_{(m,d,a,k)} = \frac{1}{J} \sum_{j=0}^J X_{(m,d,a,k)}^j. \quad (6)$$

3.3 Aggregation across tasks and safety bounds

To characterize an algorithm’s overall safety performance, each per-seed metric is first averaged over seeds $k \in \{1, \dots, K\}$, and then aggregated over the full set of task-bound pairs $(m, d) \in \mathcal{P}$. Following Agarwal et al. (2021), we report the interquartile mean (IQM) across $\mathcal{P}$, taking the mean of the values that fall between the 25th and 75th percentile. We denote this as $\text{IQM}(X_a^j)$ for a fixed policy π_j , and as $\text{IQM}(X_a)$ for the metric aggregated over the entire course of training, as defined in Eq. 6.

Safety tiers While the level of detail in the introduced metrics is essential for a reliable evaluation, it can also make it difficult to directly compare algorithms. Therefore, we define thresholds on each $X \in \{V_a, D_{\text{norm},a}, D_{\text{norm}^+,a}\}$, separating algorithms into safety tiers from level 0 to 4, with increasingly stringent safety requirements, following Table 1. We argue that any algorithm should be assessed on these tiers under two settings: over the course of training, using $\text{IQM}(X_a^{\text{expl}})$, where the superscript expl denotes evaluation with exploration noise enabled; and at the final, greedy policy, using $\text{IQM}(X_a^{\text{final,greedy}})$. This means, for example, that an algorithm may achieve tier 1 during training, but tier 2 at the final greedy policy.

Table 1: Safety tiers based on thresholds for the aggregated metrics $\text{IQM}(D_{\text{norm}})$, $\text{IQM}(V)$, and $\text{IQM}(D_{\text{norm}^+})$, for a single algorithm a .

Tier	Label	Conditions
0	Unsafe	–
1	Minimal	$\text{IQM}(D_{\text{norm}}) \leq 0$
2	Moderate	$\text{IQM}(D_{\text{norm}}) \leq 0$ and $\text{IQM}(V) \leq 0.5$
3	Substantial	$\text{IQM}(D_{\text{norm}}) \leq 0$ and $\text{IQM}(V) \leq 0.1$ and $\text{IQM}(D_{\text{norm}^+}) \leq 0.1$
4	Strict	$\text{IQM}(D_{\text{norm}}) \leq 0$ and $\text{IQM}(V) = 0$ and $\text{IQM}(D_{\text{norm}^+}) = 0$

$\text{IQM}(D_{\text{norm}}) \leq 0$ is required across all tiers from 1 upward, since it directly reflects an algorithm being safe on average. We therefore treat this as the minimal requirement for tier 1. However, $\text{IQM}(D_{\text{norm}}) \leq 0$ alone does not guarantee infrequent violations. It is possible for the average cost to sit below the safety bound while the majority of episodes still violate it, with only large undershoots in a minority of episodes pushing the mean below the bound. We therefore impose an additional threshold for tier 2 at $\text{IQM}(V) \leq 0.5$, such that the majority of episodes are not allowed to violate the safety bound. Because moving directly from tier 2 to strict safety would leave too large a gap between a moderately-safe algorithm and one that is fully constraint-satisfying, we introduce an intermediate tier 3, for which we set $\text{IQM}(V) \leq 0.1$, and additionally set $\text{IQM}(D_{\text{norm}^+}) \leq 0.1$, so that, even under this low violation rate, the average overshoot on the episodes that do violate remains within 10% of the safety bound. Tier 4 characterizes strict safety: the average cost is not allowed to exceed the safety bound, and no constraint violations are permitted at all. From $\text{IQM}(V) = 0$, it then naturally follows that $\text{IQM}(D_{\text{norm}^+})$ is 0 as well, since there are no violating episodes over which to compute an overshoot.

Aggregate cost distribution The metrics introduced in this section summarize an algorithm’s safety only at the single threshold where cost equals the bound. To capture an algorithm’s performance across the entire range of possible deviations from the safety bound, we recommend reporting the empirical cumulative distribution function (CDF) of the normalized episodic cost deviation. This provides an informative visualization of constraint adherence as a performance profile of an algorithm, and offers a visual way to compare algorithms across the entire set of episodes, rather than compressing this information into a single value, as recommended by Agarwal et al. (2021). For algorithm a , the aggregate empirical CDF evaluated at threshold κ is given by Eq. 7.

$$\text{CDF}_a^j(\kappa) = \frac{\sum_{(m,d) \in \mathcal{P}} \sum_{k=1}^K \sum_{i=1}^N \mathbb{1}\left[\frac{c_{i,(m,d,a,k)} - d}{d} \leq \kappa\right]}{|\mathcal{P}| \cdot K \cdot N}, \quad (7)$$

where $c_{i,(m,d,a,k)}$ is the cost of the i -th episode, evaluated under policy π_j , N is the number of evaluated episodes per seed, and $\mathbb{1}[\cdot]$ is the indicator function. As with the other metrics introduced in this section, the CDF can be computed for a fixed policy π_j at any training iterate j , for the final policy evaluated with exploration noise enabled or under a greedy policy, or aggregated across the entire course of training. This allows the effect of exploration noise on the full distribution of constraint adherence to be visualized directly, both over the course of training and at the final policy.

4 Results

For a detailed description of our experimental setup, we refer to Appendix Section A.1. The main focus of this paper is on the results of each metric aggregated over all tasks and bounds, shown in Table 2. Based on these aggregated metrics, we classify each algorithm into one of the safety tiers from Table 1, both over the entire course of training and at the final greedy policy. We furthermore present results on the aggregated metrics evaluated on final policies with exploration noise on and off (greedy) in Table A.2 in Appendix A.2.1. Results for single tasks and bounds are reported in Appendix A.2.2, along with their training curves and episodic cost distributions.

The results in Table 2 show that all three safe RL algorithms reach a higher safety tier for a final greedy policy than over the entire course of training with exploration noise, which is expected for the selection of studied algorithms, since all of them only guarantee constraint satisfaction asymptotically or within a local trust region, rather than at every individual policy iterate encountered during training. These results also show that FOCOPS is marginally safer than PPO-Lag in terms of violation rate over training, yet FOCOPS is assigned tier 0 for training while PPO-Lag reaches tier 2. This is because FOCOPS’s $\text{IQM}(D_{\text{norm}})$ over training is 0.06, narrowly above the threshold of $\text{IQM}(D_{\text{norm}}) \leq 0$ required for tier 1 and above. Nevertheless, FOCOPS achieves the highest $\text{IQM}(\bar{R})$ at both training and for a final greedy policy, while simultaneously achieving the lowest $\text{IQM}(D_{\text{norm}})$, $\text{IQM}(V)$, and $\text{IQM}(D_{\text{norm}}^+)$ for a final greedy policy. PPO-Lag, by contrast, is the most conservative algorithm during training, with $\text{IQM}(D_{\text{norm}}^+) = 0.25$, but achieves a substantially lower $\text{IQM}(\bar{R})$ than FOCOPS.

Table 2: Safety and performance evaluation across algorithms, aggregated over all $K = 30$ seeds for all $(m, d) \in \mathcal{P}$ pairs, IQM reported on each metric, for both over the course of all policy iterates (Train) and final *greedy* policy (Final) settings. CIs for this table, computed via stratified bootstrapping, are reported in Appendix Table A.3. **Bold text** indicates best result in that specific column, excluding PPO, as this is not a safe RL algorithm. Lower is better for each metric except for $\text{IQM}(\bar{R})$. PPO-Lag reaches tier 2 for training, even though its $\text{IQM}(V)$ is slightly higher than for FOCOPS, which sits in tier 0 due to $\text{IQM}(D_{\text{norm}}) = 0.06$ for training.

Algorithm	$\text{IQM}(\bar{R})$		$\text{IQM}(\bar{C})$		$\text{IQM}(D_{\text{norm}})$		$\text{IQM}(V)$		$\text{IQM}(D_{\text{norm}}^+)$		Tier	
	Train	Final	Train	Final	Train	Final	Train	Final	Train	Final	Train	Final
PPO	115.63	68.17	145.98	83.60	4.77	2.20	1.00	0.94	4.79	2.29	0	0
PPO-Lag	77.72	54.18	26.62	18.14	0.00	−0.13	0.43	0.34	0.25	0.50	2	2
P3O	90.41	57.80	45.69	23.76	0.55	−0.03	0.70	0.42	0.81	0.53	0	2
FOCOPS	108.76	74.26	30.39	17.09	0.06	−0.19	0.41	0.29	0.34	0.48	0	2

Figure 2 shows the empirical CDFs computed by Eq. 7, for three settings: over the entire course of training, for the final policy evaluated with exploration noise enabled, and for the final, greedy policy. The ordering of the curves at $\kappa = 0$ matches the ordering of the violation rate reported in Table 2. Note, however, that the values do not correspond one-to-one: Table 2 reports $\text{IQM}(V)$, whereas the CDF is computed over all instances. The overall shape of each curve to the right of $\kappa = 0$ reflects the severity of constraint violations across all conditions, rather than a single-value statistic. A curve that rises steeply immediately to the right of $\kappa = 0$ indicates that, on the episodes where the safety bound is exceeded, overshoot tends to be small, whereas a curve with a longer tail extending further to the right instead reflects episodes with more severe violations, consistent with a higher $D_{\text{norm}+}$.

The curves for all three safe RL algorithms shift visibly to the left when moving from training to the final policy, i.e., toward $\kappa = 0$ and below, consistent with the improvement in safety tier from training to the final policy observed in Table 2. Notably, within the final policy, the curve for the policy with exploration noise consistently lies to the left of the curve for the greedy policy, across all four algorithms, indicating fewer and less severe constraint violations under exploration noise than under a greedy policy. This is consistent with the results shown in Table A.2 in Appendix A.2.1. The CMDP guarantee $\mathbb{E}[C] \leq d$ constrains the expected cost under the policy *with* exploration noise during training, but the greedy policy was never itself directly subject to this constraint. Because this pattern holds consistently across all tasks and safety bounds, these results point to a general limitation of relying on evaluations under a final policy with exploration noise as a good indication of the performance of a final greedy policy.

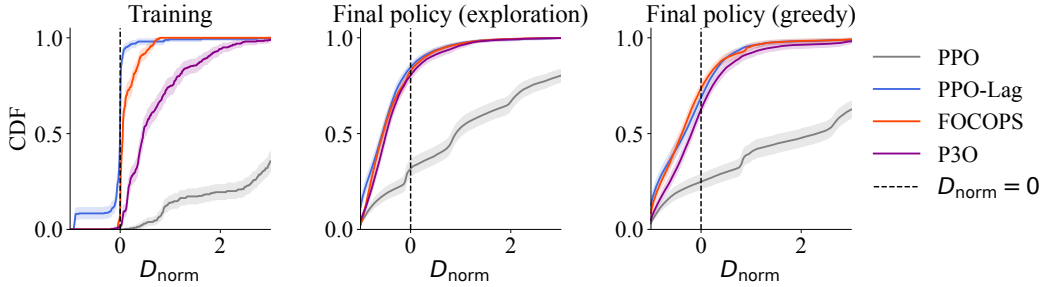


Figure 2: CDFs following Eq. 7, over the entire range of training (left), for a final policy with exploration noise on (middle), and for a final greedy policy (right). Threshold $\kappa = 0$ is indicated by the dashed vertical line at $D_{\text{norm}} = 0$. Shaded regions indicate the 95% CIs computed via stratified bootstrap. The x -axis is cut off at $D_{\text{norm}} = 2$ for visual purposes. For a visualization across the entire range of the x -axis, see Figure A.1 in Appendix A.2.1. Unlike the aggregate metrics in Table 2, the CDFs reveal the full shape of the underlying violation distribution across all thresholds, not just a single summary value. The CDF curves for the final policy with exploration noise enabled cross $\kappa = 0$ at a higher density value than those for the greedy final policy. This indicates that, for the algorithms shown, lower costs and fewer violations occur under exploration noise than under a greedy policy.

5 Discussion

While our proposed evaluation protocol provides a robust set of metrics that allow for a broadened view of safety and reliability, as well as aggregation across multiple tasks and bounds, we would like to point out that the classification into safety tiers can, in some cases, be sharply determined by a single condition. A small, consistent overshoot of bound d can shift an algorithm by multiple tiers: for instance, in Table 2, FOCOPS’s training $\text{IQM}(D_{\text{norm}})$ is only marginally above zero, yet this alone places it two tiers below PPO-Lag, despite FOCOPS achieving a comparable and even lower training $\text{IQM}(V)$. This illustrates that the thresholds introduced in Table 1 can be sensitive to small differences near a boundary, and that the assigned safety tiers should be interpreted alongside the underlying metric values rather than as a standalone summary.

Additionally, aggregating results across tasks and safety bounds using the IQM is only meaningful for metrics that are directly comparable across tasks and safety bounds with potentially very different scales. V is a rate bounded in $[0, 1]$ by construction, and D_{norm} and $D_{\text{norm}+}$ are explicitly normalized

by the safety bound d , making all three comparable across tasks and bounds. By contrast, $\bar{R}$ and $\bar{C}$ are not normalized and can vary substantially in scale across tasks, which is why our safety tiers are conditioned only on V , D_{norm} and D_{norm^+} .

Limitations Our empirical evaluation is limited to four tasks and three safety bounds. While a broader evaluation across additional task domains (Tomilin et al., 2025; Ramanujam et al., 2025) and bounds would further strengthen the generality of our findings, the selected conditions were chosen to capture a broad range of diversity in constraint types and task complexity, providing a clear empirical basis for the results presented in this study, with approximately 2,880 GPU-hours on NVIDIA L4 GPUs with 24GB of memory. However, our results are limited to algorithms that permit constraint violations during training and only guarantee constraint adherence asymptotically or approximately. Stricter safe RL methods that disallow constraint violations even during training (Alshiekh et al., 2018; Dalal et al., 2018; As et al., 2025; Cheng et al., 2019) are yet to be evaluated using our proposed framework. We therefore provide an open-source evaluation suite that enables the community to adopt our proposed evaluation metrics and apply them to other safe RL algorithms and task domains, which would help further extending the broadened view of safety and reliability that this work provides.

Future Work For safe real-world deployment, the gap between simulation-based and real-world deployment performance should ideally be negligible. Our results, however, show a substantial gap between a final policy with and without exploration noise for the algorithms evaluated in this work, even within simulation alone. Future work could compare the evaluation of simulated greedy policy rollouts against real-world deployment performance, using the distributions of V , D_{norm} , and D_{norm^+} introduced here to quantify this gap directly.

The results in Appendix A.2.2 reveal that the gap between exploration and greedy final-policy performance, observed in the aggregate results in Section 4, is not uniform across tasks: the episodic cost distributions in Figures A.7-A.10 show a substantial difference between exploration and greedy policies for Safe Circle Point and Safe Goal Point, but considerable overlap for Safe Button Point and Safe Push Point. This suggests that the gap between final policies with exploration noise on and off may itself be task-dependent, and could be further explored in future studies.

We argue that a natural extension of this work is to incorporate the proposed evaluation metrics directly into algorithm design, rather than using them only for evaluation purposes. For example, an algorithm’s training objective could explicitly minimize V , D_{norm} or D_{norm^+} . This would allow safe RL algorithms to be optimized directly against the same safety and reliability criteria used to evaluate them.

6 Conclusion

In this work, we introduced a comprehensive safe RL evaluation framework, including the violation rate (V), mean normalized cost deviation (D_{norm}), and normalized violation magnitude (D_{norm^+}), together with a safety tier system and a distributional view of constraint adherence, directly addressing the insufficiency of the expectation-based guarantee $\mathbb{E}[C] \leq d$ illustrated in Figure 1. Our evaluation study showed that no algorithm dominates uniformly across all conditions and both training and for a final policy: FOCOPS achieves the strongest overall balance of reward and safety at a final greedy policy, while PPO-Lag trades off reward for a lower violation magnitude over the course of training. We further found that, across all evaluated algorithms, the final policy is consistently less safe when evaluated greedily than under exploration noise, pointing to a general limitation of relying on evaluations under a final policy with exploration noise as a good indication of the performance of a final greedy policy. Additionally, our results show that aggregate safety tiers, the CDF, and results per task and safety bound each reveal information that is crucial to a complete evaluation of an algorithm’s safety and reliability. We therefore recommend reporting all three jointly, rather than mean cost or mean reward alone, and provide an open-source evaluation suite to provide the tools to assess safe RL algorithms by their reliability, not just their average performance.

References

- Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. 2017. Constrained Policy Optimization. doi:10.48550/arXiv.1705.10528 arXiv:1705.10528.
- Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. 2021. Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems* 34 (2021), 29304–29320.
- Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. 2018. Safe reinforcement learning via shielding. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.
- Eitan Altman. 1999. *Constrained Markov Decision Processes*. Routledge, New York. doi:10.1201/9781315140223
- Yarden As, Lenart Treven, Carmelo Sferrazza, Stelian Coros, and Andreas Krause. 2025. Actsafes: Active exploration with safety constraints for reinforcement learning. In *International Conference on Learning Representations*, Vol. 2025. 29752–29776.
- Richard Bellman. 1957. A Markovian Decision Process. *Journal of Mathematics and Mechanics* 6, 5 (1957), 679–684. Publisher: Indiana University Mathematics Department.
- Richard Cheng, Gábor Orosz, Richard M Murray, and Joel W Burdick. 2019. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 3387–3395.
- Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerik, Todd Hester, Cosmin Paduraru, and Yuval Tassa. 2018. Safe exploration in continuous action spaces. *arXiv preprint arXiv:1801.08757* (2018).
- Bradley Efron. 1992. Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics: Methodology and distribution*. Springer, 569–593.
- Javier Garcia and Fernando Fernández. 2015. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research* 16, 1 (2015), 1437–1480.
- Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*. Pmlr, 1861–1870.
- Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Josef Dai, and Yaodong Yang. 2023. Safety gymnasium: A unified safe reinforcement learning benchmark. *Advances in Neural Information Processing Systems* 36 (2023), 18964–18993.
- Jiaming Ji, Jiayi Zhou, Borong Zhang, Juntao Dai, Xuehai Pan, Ruiyang Sun, Weidong Huang, Yiran Geng, Mickel Liu, and Yaodong Yang. 2024. Omnisafe: An infrastructure for accelerating safe reinforcement learning research. *Journal of Machine Learning Research* 25, 285 (2024), 1–6.
- Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *nature* 518, 7540 (2015), 529–533.
- Asha Ramanujam, Adam Elyoumi, Hao Chen, Sai Madhukiran Kompalli, Akshdeep Singh Ahluwalia, Shraman Pal, Dimitri J Papageorgiou, and Can Li. 2025. Safeor-gym: A benchmark suite for safe reinforcement learning algorithms on practical operations research problems. *arXiv preprint arXiv:2506.02255* (2025).
- Alex Ray, Joshua Achiam, and Dario Amodei. 2019. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708* 7, 1 (2019), 2.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

- Lindsay Spoor, Álvaro Serra-Gómez, Aske Plaat, and Thomas Moerland. 2025. Towards a Practical Understanding of Lagrangian Methods in Safe Reinforcement Learning. *arXiv preprint arXiv:2510.17564* (2025).
- Adam Stooke, Joshua Achiam, and Pieter Abbeel. 2020. Responsive Safety in Reinforcement Learning by PID Lagrangian Methods. In *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 9133–9143. ISSN: 2640-3498.
- Richard S Sutton, Andrew G Barto, et al. 2018. *Reinforcement learning: An introduction*. Vol. 2.
- Chen Tessler, Daniel J. Mankowitz, and Shie Mannor. 2019. Reward Constrained Policy Optimization. In *International Conference on Learning Representations*.
- Tristan Tomilin, Mourad Boustani, Mickey Beurskens, and Thiago D Simão. 2026. CRAX: Fast Safe Reinforcement Learning Benchmarking. *arXiv preprint arXiv:2606.20376* (2026).
- Tristan Tomilin, Meng Fang, and Mykola Pechenizkiy. 2025. HASARD: A Benchmark for Vision-Based Safe Reinforcement Learning in Embodied Agents. *arXiv:2503.08241 [cs.AI]*
- Linrui Zhang, Li Shen, Long Yang, Shixiang Chen, Bo Yuan, Xueqian Wang, and Dacheng Tao. 2022. Penalized proximal policy optimization for safe reinforcement learning. *arXiv preprint arXiv:2205.11814* (2022).
- Yiming Zhang, Quan Vuong, and Keith Ross. 2020. First Order Constrained Optimization in Policy Space. In *Advances in Neural Information Processing Systems*, Vol. 33. Curran Associates, Inc., 15338–15349. <https://proceedings.neurips.cc/paper/2020/hash/af5d5ef24881f3c3049a7b9bfe74d58b-Abstract.html>

A.1 Experimental setup

We train algorithms PPO-Lag (Ray et al., 2019), P3O (Zhang et al., 2022), FOCOPS (Zhang et al., 2020) and, as an unconstrained baseline, PPO (Schulman et al., 2017) using the CRAX benchmark suite (Tomilin et al., 2026), for 30 seeds each. The set of tasks is $\mathcal{M} = \{\text{Safe Goal Point, Safe Circle Point, Safe Push Point, Safe Button Point}\}$, all Level 1, and the set of safety bounds is $\mathcal{D} = \{15, 25, 50\}$. This set of tasks focuses on safe navigation and consists of different types of constraints and rewards, selected carefully to span a set of tasks with sufficient variety in their cost functions.

We train each run for $5 \cdot 10^8$ total environment steps, with an episode length of $T = 2000$ steps. During training, exploration noise is turned on, and training metrics are recorded once per complete batch: after every cycle in which all $N = 2048$ parallel environments finish their episode (episode length $T = 2000$ steps, or equivalently every $\sim 4.2 \cdot 10^6$ environment steps), yielding $J = 122$ data points per seed over $5 \cdot 10^8$ total environment steps. Each data point is averaged over the 2048 episodes in that batch, for a total of approximately $2.5 \cdot 10^5$ episodes per seed. At the final policy evaluation we turn off exploration noise and use a deterministic trained (greedy) policy, and we report the average over $N = 100$ evaluation episodes per seed.

For metric $X \in \{\bar{R}, \bar{C}, D_{\text{norm}}, V, D_{\text{norm}}^+\}$, we report the task- and bound-specific results as $\frac{1}{K} \sum_{k=1}^K X_{(m,d,a,k)} \pm \sigma_k$, where σ_k is the sample standard deviation for seed k , for $K = 30$ seeds. This standard deviation reflects the variability across individual training runs under identical conditions, and does not shrink with additional seeds. For aggregation over all tasks and safety bounds, we use the IQM as explained in Section 3.3. Since the IQM is a nonlinear statistic of the distribution across conditions, we report uncertainty as a 95% confidence interval (CI) computed via a stratified bootstrap with $B = 2000$ replicates (Efron, 1992; Agarwal et al., 2021). Unlike the standard deviation in a single task or bound, this confidence interval reflects the standard error of the IQM estimate itself, i.e., how much the aggregated IQM value would be expected to change under a different sample of seeds, and would narrow with additional seeds or conditions. We therefore report both: the standard deviation characterizes the inherent variability of individual training runs, while the IQM confidence interval characterizes the reliability of our aggregate statistics. Likewise, the CDF curve following Eq. 7 is estimated from finite data (K seeds $\times N$ episodes per condition), so we used the same stratified bootstrap procedure to report the CI band showing how much the curve would shift under different seeds.

Compute resources All experiments were run on NVIDIA L4 GPUs (24 GB memory), using a single GPU and 4 CPU cores per run. A single training run of $5 \cdot 10^8$ environment steps took approximately 2 hours of wall-clock time. Our full experimental suite consists of 4 tasks $\times$ 3 safety bounds $\times$ 4 algorithms (PPO, PPO-Lag, P3O, FOCOPS) $\times$ 30 seeds, totaling 1,440 runs, for an estimated total of approximately 2,880 GPU-hours, or equivalently, approximately 11,520 CPU-core-hours, given 4 cores allocated per run.

Table A.1: Hyperparameters used across all experiments.

<i>Shared (all algorithms)</i>			
Hyperparameter	Value	Hyperparameter	Value
Total environment steps	5×10^8	Discount factor γ	0.99
Episode length	2000	GAE λ	0.95
Parallel environments	2048	PPO clipping ε	0.3
Unroll length	8	Entropy coefficient	5×10^{-3}
Batch size	1024	Reward scaling	0.1
Number of minibatches	32	Observation normalisation	✓
SGD epochs per batch	6	Deterministic eval. episodes	100
Learning rate	5×10^{-4}	Seeds	30
<i>Algorithm-specific</i>			
<i>PPO-Lag</i>		<i>FOCOPS</i>	
Lagrange multiplier LR	3.0	Initial ν	0.1
Initial λ	0.0	ν learning rate	1.0
		$\nu_{\max}$	100.0
		KL penalty λ	1.5
		Advantage temperature η	0.02
<i>P3O</i>			
Initial κ	0.01		
κ increase factor	1.1		
$\kappa_{\max}$	50.0		

A.2 Additional results

A.2.1 Aggregate metrics

Figure A.1 shows the empirical CDFs computed by Eq. 7, aggregated over all tasks, safety bounds, and seeds. Table A.2 presents algorithm performance on the aggregated metrics evaluated on final policies with exploration noise on (expl) and off (greedy), together with their stratified bootstrapped CIs. The results show that, across all four algorithms, every metric is significantly lower under exploration noise than under the greedy policy, consistent with the distributional shift illustrated in Figure A.1 (which is better visible in Figure 2 in the main paper). This holds even for the unconstrained PPO baseline, indicating that a final greedy policy systematically increases constraint violation across all evaluated algorithms, rather than being specific to safe RL algorithms only.

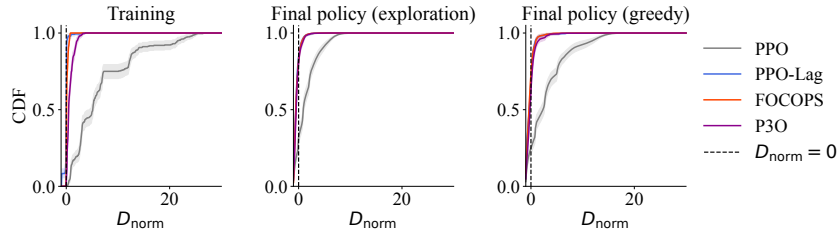


Figure A.1: CDFs following Eq. 7, over the entire range of training (left), for a final policy with exploration noise on (middle), and for a final greedy policy (right). Threshold $\kappa = 0$ is indicated by the dashed vertical line at $D_{\text{norm}} = 0$. Shaded regions indicate the 95% CIs computed via stratified bootstrap.

Table A.2: Safety and performance evaluation across algorithms, aggregated over all (m, d) pairs. IQM reported with 95% stratified bootstrap CI [lower, upper] ($B = 2000$), comparing the final policy evaluated with exploration noise enabled against the final, greedy policy. **Bold** indicates best result per column, excluding PPO.

Algorithm	IQM($\bar{R}$)		IQM($\bar{C}$)		IQM(D_{norm})		IQM(V)		IQM($D_{\text{norm}+}$)		Tier	
	Final (expl.)	Final (greedy)	Final (expl.)	Final (greedy)	Final (expl.)	Final (greedy)	Final (expl.)	Final (greedy)	Final (expl.)	Final (greedy)	Final (expl.)	Final (greedy)
PPO	50.37 _[40.32, 61.43]	68.17 _[67.07, 69.22]	60.63 _[59.07, 62.13]	83.60 _[81.24, 85.73]	2.196 _[2.111, 2.280]	0.821 _[0.805, 0.835]	0.941 _[0.925, 0.958]	1.285 _[1.238, 1.321]	2.288 _[2.213, 2.366]	0.497 _[0.429, 0.514]	0	0
PPO-Lag	38.80 _[36.91, 40.60]	54.18 _[51.99, 56.99]	14.26 _[13.17, 15.47]	18.14 _[16.50, 20.69]	-0.120 _[-0.204, -0.089]	0.110 _[0.090, 0.130]	0.341 _[0.312, 0.378]	0.425 _[0.244, 0.348]	0.497 _[0.429, 0.514]	0.497 _[0.429, 0.514]	2	2
P3O	42.68 _[41.01, 44.23]	57.80 _[55.95, 59.65]	15.71 _[14.70, 16.78]	23.76 _[21.80, 25.81]	-0.023 _[-0.046, -0.000]	0.175 _[0.142, 0.207]	0.421 _[0.400, 0.432]	0.435 _[0.304, 0.392]	0.531 _[0.478, 0.557]	0.531 _[0.478, 0.557]	2	2
FOCOPS	52.36 _[50.79, 53.83]	74.26 _[72.60, 75.70]	14.21 _[12.99, 15.29]	17.09 _[15.53, 19.31]	-0.193 _[-0.265, -0.131]	0.109 _[0.089, 0.137]	0.291 _[0.258, 0.319]	0.305 _[0.202, 0.244]	0.479 _[0.306, 0.414]	0.479 _[0.306, 0.414]	2	2

Table A.3: Safety and performance evaluation across algorithms, aggregated over all (m, d) pairs. IQM reported with 95% stratified bootstrap CI [lower, upper] ($B = 2000$), for both over the course of all policy iterates (Train) and final *greedy* policy (Final) settings. **Bold** indicates best result per column, excluding PPO.

Algorithm	IQM($\bar{R}$)		IQM($\bar{C}$)		IQM(D_{norm})		IQM(V)		IQM($D_{\text{norm}+}$)		Tier	
	Train	Final	Train	Final	Train	Final	Train	Final	Train	Final	Train	Final
PPO	115.63 _[114.87, 116.38]	68.17 _[67.08, 69.21]	145.98 _[145.18, 146.83]	83.60 _[81.24, 85.79]	4.77 _[4.70, 4.85]	2.20 _[2.11, 2.28]	1.00 _[1.00, 1.00]	0.94 _[0.92, 0.96]	4.79 _[4.72, 4.87]	2.29 _[2.21, 2.36]	0	0
PPO-Lag	77.72 _[73.31, 80.50]	54.18 _[51.92, 56.40]	26.62 _[25.69, 27.77]	18.14 _[16.48, 20.79]	0.00 _[-0.01, 0.00]	-0.13 _[-0.21, -0.09]	0.43 _[0.42, 0.44]	0.34 _[0.31, 0.37]	0.25 _[0.23, 0.28]	0.50 _[0.46, 0.56]	2	2
P3O	90.41 _[89.32, 91.53]	57.80 _[56.04, 59.67]	45.69 _[44.70, 46.75]	23.76 _[21.79, 25.82]	0.55 _[0.52, 0.57]	-0.03 _[-0.05, -0.01]	0.70 _[0.69, 0.72]	0.42 _[0.40, 0.43]	0.81 _[0.78, 0.83]	0.53 _[0.49, 0.57]	0	0
FOCOPS	108.76 _[107.77, 109.78]	74.26 _[72.63, 75.83]	30.39 _[30.26, 30.52]	17.09 _[15.58, 19.51]	0.06 _[0.06, 0.06]	-0.19 _[-0.26, -0.13]	0.41 _[0.40, 0.41]	0.29 _[0.26, 0.32]	0.34 _[0.33, 0.34]	0.48 _[0.43, 0.52]	0	2

A.2.2 Task- and bound-specific results

All training curves plotted in Figures A.3-A.6 are smoothed using a moving average filter with a window of 5. The episodic distributions for final policies with exploration noise on or off (greedy) for each individual task and safety bound are shown in Figures A.7-A.10, over $N = 100$ rollouts. All metrics $X \in \{\bar{R}, \bar{C}, D_{\text{norm}}, V, D_{\text{norm}}^+\}$ are reported in Tables A.4-A.7 as $\frac{1}{K} \sum_{k=1}^K X_{(m,d,a,k)} \pm \sigma_k$, where σ_k is the sample standard deviation for seed k , for $K = 30$ seeds, for both over the entire course of training, and for the final greedy policy.

On the results for Safe Button Point The results for PPO-Lag on Safe Button Point in Table A.7 show that, for $d = 25$, the average reward $\bar{R}$ is negative at both training and for the final policy, and drops substantially further at the final policy performance (-8.84 ± 41.09) compared to training (-0.17 ± 9.63). We suspect this artifact is caused by a collapse in the cost, in which the Lagrange multiplier λ grew too large during training, effectively freezing the policy into doing nothing as shown in Figure A.6b. We confirm this by inspecting the corresponding Lagrange multiplier trajectory, shown in Figure A.2.

Furthermore, comparing the training and final policy columns for $\bar{R}$ and D_{norm} across all algorithms on Safe Button Point in Table A.7, both metrics show a large drop from training to final policy performance. Rewards in this task come from actively pressing goal buttons (Tomilin et al., 2026), which requires exploration or directed behavior. The final (greedy) policies of all evaluated algorithms appear to converge to a passive strategy resulting in low reward and low cost, possibly navigating the environment without pressing any buttons at all. This is consistent with the drop in V observed at the final policy performance across all algorithms: pressing fewer buttons means fewer opportunities to trigger a hazard and violate the safety constraint.

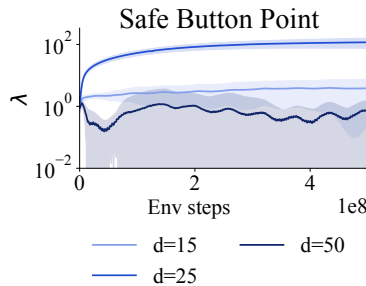


Figure A.2: Lagrange multiplier λ for PPO-Lag on Safe Button Point (Level 1), for safety bounds $d \in \{15, 25, 50\}$, averaged across 30 seeds. The shaded regions denote the 1σ over all seeds. The curve for $d = 25$ shows that λ grows very large compared to $d = 15$ and $d = 50$, suggesting that the policy for $d = 25$ is frozen throughout the course of training.

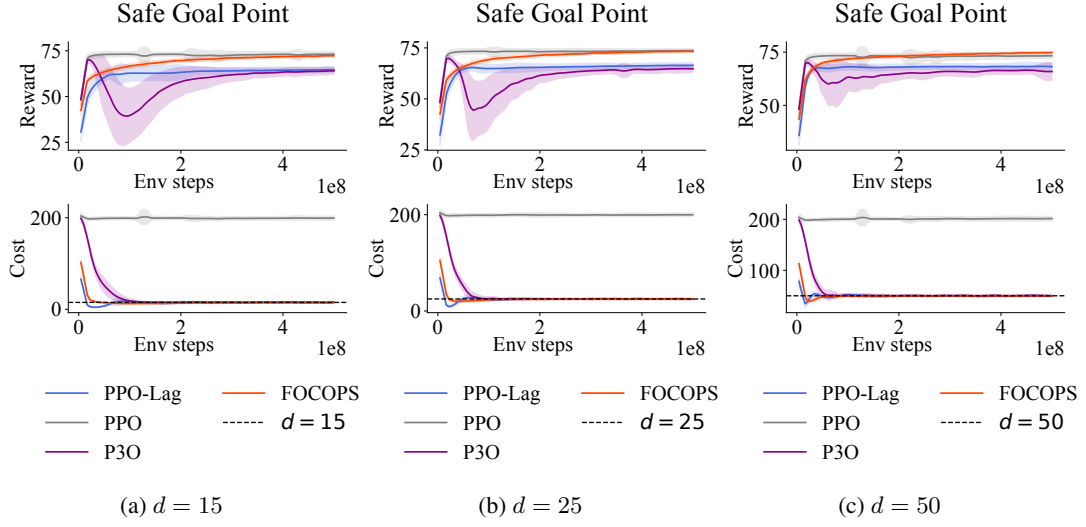


Figure A.3: Smoothed training curves for Safe Goal Point (Level 1), for safety bounds $d \in \{15, 25, 50\}$, averaged over 30 seeds. The shaded regions denote the 1σ over 30 seeds.

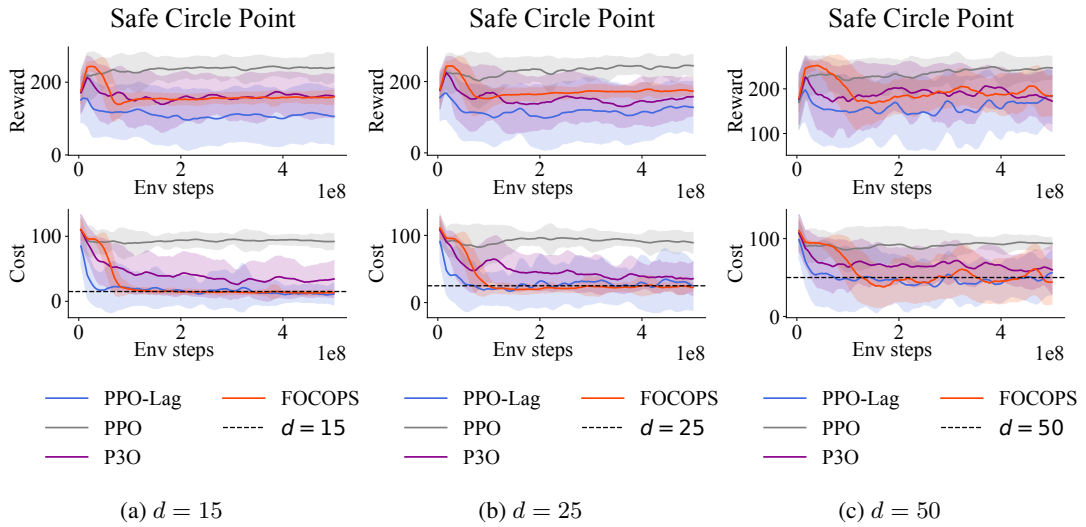


Figure A.4: Smoothed training curves for Safe Circle Point (Level 1), for safety bounds $d \in \{15, 25, 50\}$, averaged over 30 seeds. The shaded regions denote the 1σ over 30 seeds.

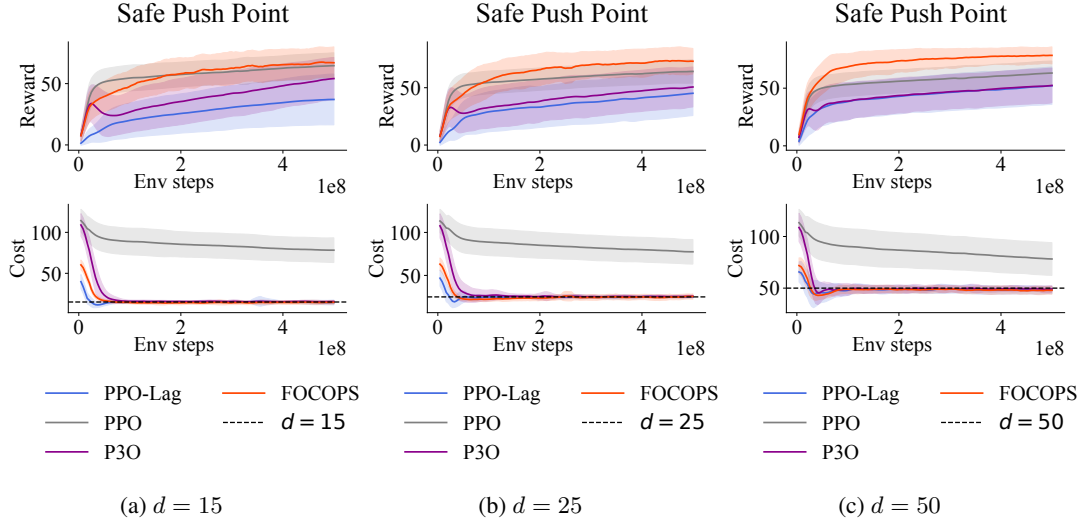


Figure A.5: Smoothed training curves for Safe Push Point (Level 1), for safety bounds $d \in \{15, 25, 50\}$, averaged over 30 seeds. The shaded regions denote the 1σ over 30 seeds.

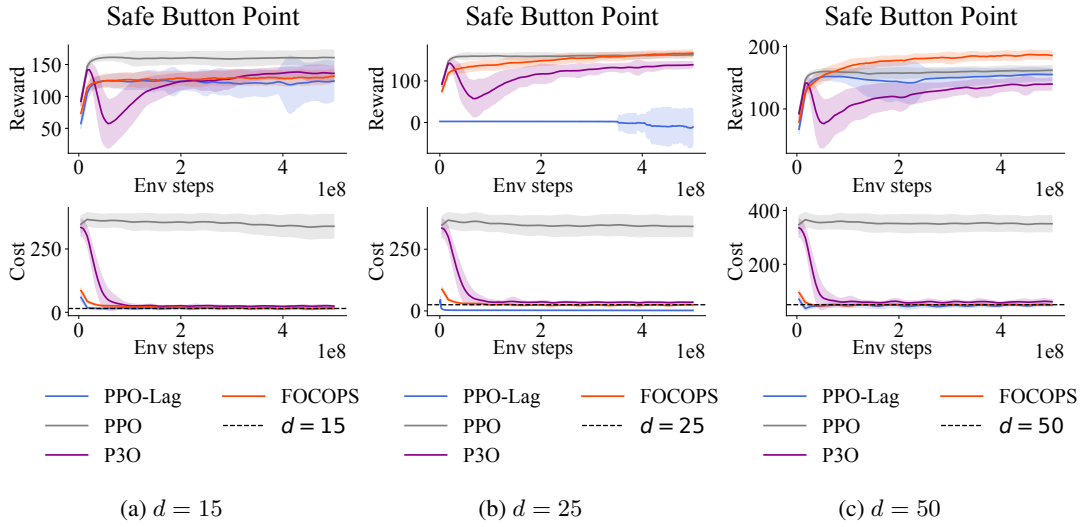


Figure A.6: Smoothed training curves for Safe Button Point (Level 1), for safety bounds $d \in \{15, 25, 50\}$, averaged over 30 seeds. The shaded regions denote the 1σ over 30 seeds.

Safe Goal Point

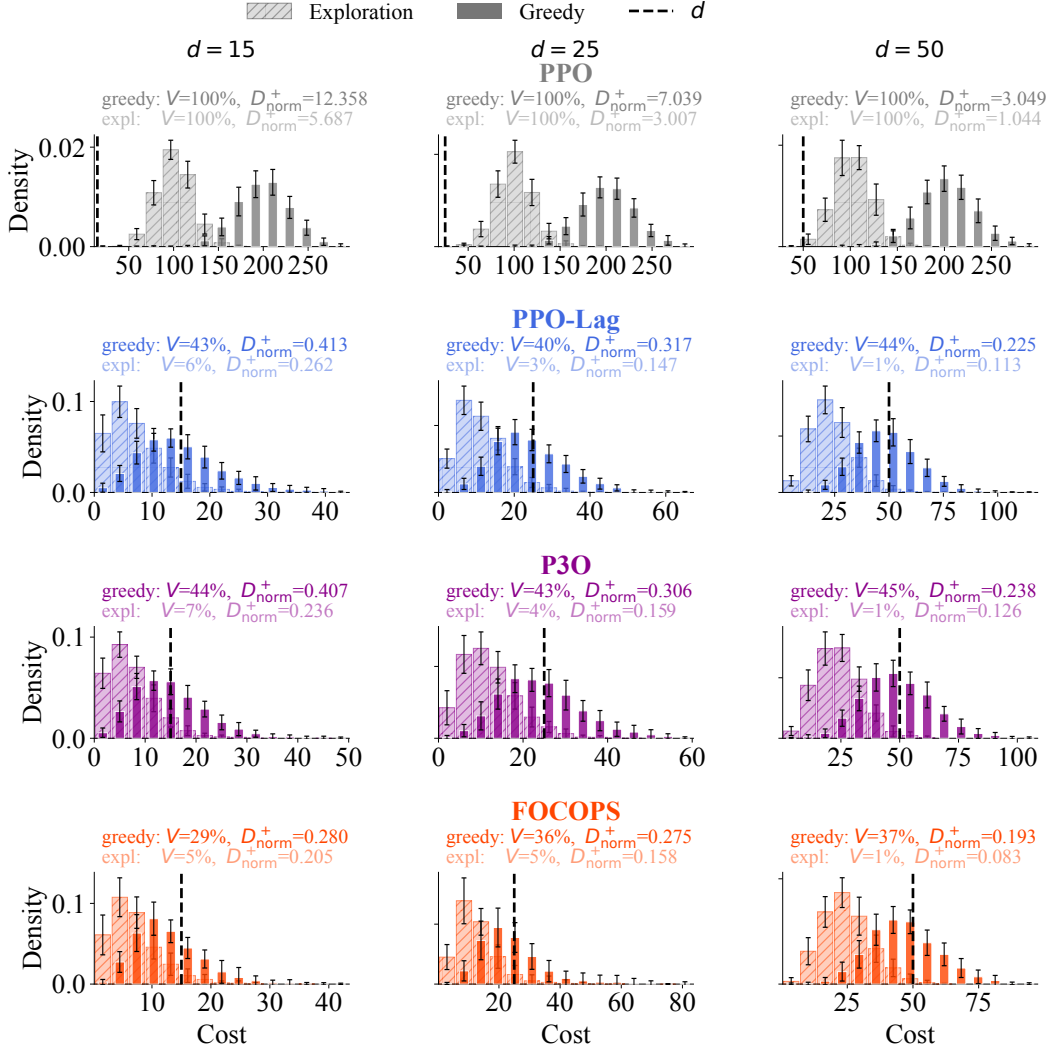


Figure A.7: Episodic cost distributions for final policies with exploration noise on and off (greedy). Safe Goal Point (Level 1). Averaged over 30 seeds. Error bars denote 1σ over 30 seeds.

Table A.4: Safety and performance evaluation per metric for Safe Goal Point (Level 1), for each individual $d \in \{15, 25, 50\}$. The numbers indicate mean $\pm 1\sigma$ over 30 seeds. **Bold text** indicates best result in that specific column, excluding PPO, as this is not a safe RL algorithm.

Algorithm	R		C		D_{norm}		V		D_{norm}^+		Tier	
	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)
$d = 15$												
PPO	72.45 $\pm$ 1.50	72.05 $\pm$ 1.59	199.05 $\pm$ 3.47	200.39 $\pm$ 5.23	12.27 $\pm$ 0.23	12.36 $\pm$ 0.35	1.00 $\pm$ 0.00	1.00 $\pm$ 0.00	12.27 $\pm$ 0.23	12.36 $\pm$ 0.35	T0	F0
PPO-Lag	62.46 $\pm$ 2.34	63.82 $\pm$ 1.79	14.84 $\pm$ 0.25	14.78 $\pm$ 1.32	-0.01 $\pm$ 0.02	-0.01 $\pm$ 0.09	0.44 $\pm$ 0.03	0.43 $\pm$ 0.08	0.22 $\pm$ 0.07	0.41 $\pm$ 0.07	T2	F2
P3O	58.12 $\pm$ 4.66	63.15 $\pm$ 1.96	26.68 $\pm$ 1.76	14.80 $\pm$ 1.21	0.78 $\pm$ 0.12	-0.01 $\pm$ 0.08	0.62 $\pm$ 0.03	0.44 $\pm$ 0.07	1.29 $\pm$ 0.16	0.41 $\pm$ 0.07	T0	F2
FOCOPS	68.86 $\pm$ 0.41	73.73 $\pm$ 0.67	15.79 $\pm$ 0.05	12.65 $\pm$ 2.19	0.05 $\pm$ 0.00	-0.16 $\pm$ 0.15	0.32 $\pm$ 0.03	0.29 $\pm$ 0.14	0.44 $\pm$ 0.07	0.28 $\pm$ 0.08	T0	F2
$d = 25$												
PPO	72.92 $\pm$ 1.21	72.35 $\pm$ 1.33	199.35 $\pm$ 3.56	201.53 $\pm$ 5.87	6.97 $\pm$ 0.14	7.06 $\pm$ 0.23	1.00 $\pm$ 0.00	1.00 $\pm$ 0.00	6.97 $\pm$ 0.14	7.06 $\pm$ 0.24	T0	F0
PPO-Lag	64.70 $\pm$ 1.65	65.40 $\pm$ 1.40	24.79 $\pm$ 0.36	23.76 $\pm$ 1.27	-0.01 $\pm$ 0.01	-0.05 $\pm$ 0.05	0.46 $\pm$ 0.03	0.40 $\pm$ 0.06	0.12 $\pm$ 0.02	0.32 $\pm$ 0.04	T2	F2
P3O	60.79 $\pm$ 3.14	64.00 $\pm$ 1.91	34.75 $\pm$ 1.08	24.01 $\pm$ 2.40	0.39 $\pm$ 0.04	-0.04 $\pm$ 0.10	0.59 $\pm$ 0.02	0.43 $\pm$ 0.11	0.70 $\pm$ 0.08	0.31 $\pm$ 0.05	T0	F2
FOCOPS	70.44 $\pm$ 0.35	74.17 $\pm$ 2.12	25.49 $\pm$ 0.04	23.53 $\pm$ 6.57	0.02 $\pm$ 0.00	-0.06 $\pm$ 0.26	0.37 $\pm$ 0.03	0.36 $\pm$ 0.21	0.22 $\pm$ 0.02	0.28 $\pm$ 0.18	T0	F2
$d = 50$												
PPO	72.66 $\pm$ 1.48	72.34 $\pm$ 1.34	200.85 $\pm$ 3.50	202.53 $\pm$ 5.50	3.02 $\pm$ 0.07	3.05 $\pm$ 0.11	1.00 $\pm$ 0.00	1.00 $\pm$ 0.00	3.02 $\pm$ 0.07	3.05 $\pm$ 0.11	T0	F0
PPO-Lag	67.32 $\pm$ 1.46	67.29 $\pm$ 1.61	49.87 $\pm$ 0.22	48.60 $\pm$ 1.80	0.00 $\pm$ 0.00	-0.03 $\pm$ 0.04	0.49 $\pm$ 0.04	0.44 $\pm$ 0.06	0.06 $\pm$ 0.01	0.23 $\pm$ 0.03	T2	F2
P3O	64.86 $\pm$ 2.71	64.89 $\pm$ 4.10	57.01 $\pm$ 0.92	48.99 $\pm$ 2.30	0.14 $\pm$ 0.02	-0.02 $\pm$ 0.05	0.56 $\pm$ 0.03	0.45 $\pm$ 0.07	0.28 $\pm$ 0.03	0.24 $\pm$ 0.04	T0	F2
FOCOPS	72.39 $\pm$ 0.27	75.65 $\pm$ 0.60	50.27 $\pm$ 0.02	46.39 $\pm$ 3.35	0.01 $\pm$ 0.00	-0.07 $\pm$ 0.07	0.43 $\pm$ 0.03	0.37 $\pm$ 0.10	0.08 $\pm$ 0.01	0.19 $\pm$ 0.04	T0	F2

Safe Circle Point

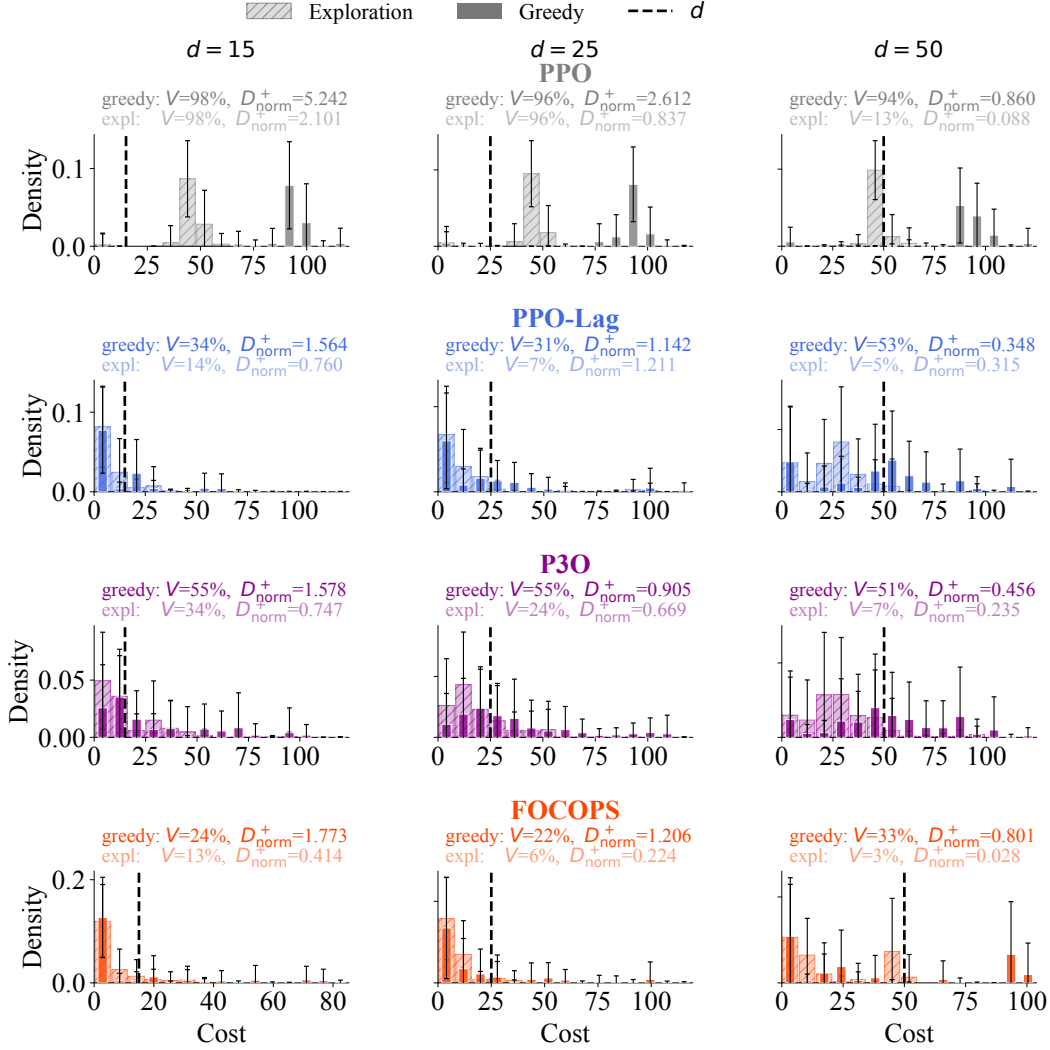


Figure A.8: Episodic cost distributions for final policies with exploration noise on and off (greedy). Safe Circle Point (Level 1). Averaged over 30 seeds. Error bars denote 1σ over 30 seeds.

Table A.5: Safety and performance evaluation per metric for Safe Circle Point (Level 1), for each individual $d \in \{15, 25, 50\}$. The numbers indicate mean $\pm 1\sigma$ over 30 seeds. **Bold text** indicates best result in that column, excluding PPO, as this is not a safe RL algorithm.

Algorithm	R		C		D_{norm}		V		D_{norm}^+		Tier	
	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)
$d = 15$												
PPO	234.01 $\pm$ 17.41	240.14 $\pm$ 37.48	93.02 $\pm$ 5.40	91.52 $\pm$ 12.31	5.20 $\pm$ 0.36	5.10 $\pm$ 0.82	0.99 $\pm$ 0.01	0.98 $\pm$ 0.11	5.24 $\pm$ 0.35	5.24 $\pm$ 0.45	T0	F0
PPO-Lag	111.81 $\pm$ 52.03	100.74 $\pm$ 80.28	17.55 $\pm$ 8.36	12.37 $\pm$ 15.73	0.17 $\pm$ 0.56	-0.18 $\pm$ 1.05	0.35 $\pm$ 0.12	0.34 $\pm$ 0.43	2.02 $\pm$ 1.10	2.89 $\pm$ 3.66	T0	F2
P3O	161.95 $\pm$ 36.26	151.60 $\pm$ 69.61	41.46 $\pm$ 12.14	29.86 $\pm$ 25.48	1.76 $\pm$ 0.81	0.99 $\pm$ 1.70	0.76 $\pm$ 0.09	0.55 $\pm$ 0.38	2.38 $\pm$ 0.83	1.83 $\pm$ 1.83	T0	F0
FOCOPS	162.69 $\pm$ 9.35	179.63 $\pm$ 45.95	24.42 $\pm$ 1.25	11.55 $\pm$ 21.07	0.63 $\pm$ 0.08	-0.23 $\pm$ 1.40	0.48 $\pm$ 0.11	0.24 $\pm$ 0.42	1.65 $\pm$ 0.50	1.78 $\pm$ 1.54	T0	F2
$d = 25$												
PPO	230.53 $\pm$ 20.00	244.86 $\pm$ 22.52	92.04 $\pm$ 7.72	88.55 $\pm$ 16.18	2.68 $\pm$ 0.31	2.54 $\pm$ 0.65	0.97 $\pm$ 0.05	0.96 $\pm$ 0.17	2.78 $\pm$ 0.23	2.62 $\pm$ 0.49	T0	F0
PPO-Lag	117.27 $\pm$ 49.88	128.72 $\pm$ 75.68	28.77 $\pm$ 15.65	23.61 $\pm$ 34.47	0.15 $\pm$ 0.63	-0.06 $\pm$ 1.38	0.39 $\pm$ 0.11	0.31 $\pm$ 0.40	1.35 $\pm$ 0.69	1.89 $\pm$ 2.14	T0	F2
P3O	152.18 $\pm$ 40.39	157.25 $\pm$ 53.58	47.50 $\pm$ 9.04	35.41 $\pm$ 23.06	0.90 $\pm$ 0.36	0.42 $\pm$ 0.92	0.74 $\pm$ 0.09	0.55 $\pm$ 0.40	1.32 $\pm$ 0.34	1.13 $\pm$ 1.25	T0	F0
FOCOPS	174.84 $\pm$ 10.24	198.84 $\pm$ 35.64	31.87 $\pm$ 1.37	19.19 $\pm$ 30.20	0.27 $\pm$ 0.05	-0.23 $\pm$ 1.21	0.37 $\pm$ 0.07	0.22 $\pm$ 0.40	1.22 $\pm$ 0.38	1.44 $\pm$ 1.45	T0	F2
$d = 50$												
PPO	234.20 $\pm$ 16.73	228.97 $\pm$ 55.88	91.60 $\pm$ 8.90	88.75 $\pm$ 17.95	0.83 $\pm$ 0.18	0.78 $\pm$ 0.36	0.96 $\pm$ 0.08	0.94 $\pm$ 0.16	0.89 $\pm$ 0.10	0.86 $\pm$ 0.17	T0	F0
PPO-Lag	159.31 $\pm$ 35.70	172.76 $\pm$ 69.93	48.52 $\pm$ 5.20	48.39 $\pm$ 27.42	-0.03 $\pm$ 0.10	-0.03 $\pm$ 0.55	0.48 $\pm$ 0.05	0.53 $\pm$ 0.46	0.53 $\pm$ 0.33	0.47 $\pm$ 0.54	T2	F1
P3O	190.05 $\pm$ 29.27	166.43 $\pm$ 70.37	66.74 $\pm$ 10.65	56.41 $\pm$ 31.54	0.33 $\pm$ 0.21	0.13 $\pm$ 0.63	0.74 $\pm$ 0.12	0.52 $\pm$ 0.43	0.55 $\pm$ 0.15	0.64 $\pm$ 0.59	T0	F0
FOCOPS	196.17 $\pm$ 6.95	198.88 $\pm$ 64.22	57.18 $\pm$ 1.81	38.57 $\pm$ 39.78	0.14 $\pm$ 0.04	-0.23 $\pm$ 0.80	0.47 $\pm$ 0.04	0.33 $\pm$ 0.47	0.80 $\pm$ 0.08	0.87 $\pm$ 0.41	T0	F2

Safe Push Point

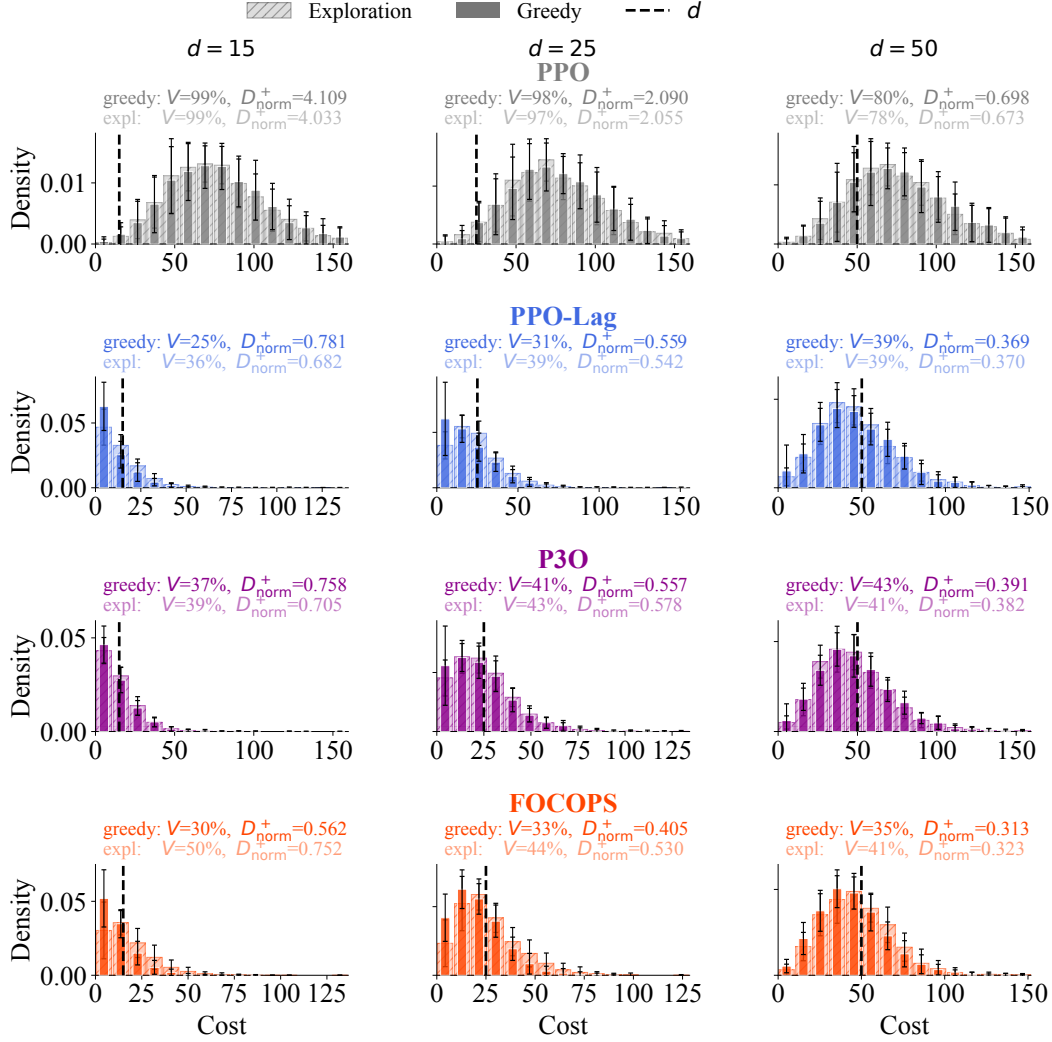


Figure A.9: Episodic cost distributions for final policies with exploration noise on and off (greedy). Safe Push Point (Level 1). Averaged over 30 seeds. Error bars denote 1σ over 30 seeds.

Table A.6: Safety and performance evaluation per metric for Safe Push Point (Level 1), for each individual $d \in \{15, 25, 50\}$. The numbers indicate mean $\pm 1\sigma$ over 30 seeds. **Bold text** indicates best result in that specific column, excluding PPO, as this is not a safe RL algorithm.

Algorithm	R		C		D_{norm}		V		D_{norm}^+		Tier	
	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)
$d = 15$												
PPO	57.07 $\pm$ 9.99	64.79 $\pm$ 10.41	85.43 $\pm$ 15.02	77.26 $\pm$ 15.05	4.70 $\pm$ 1.00	4.15 $\pm$ 1.00	1.00 $\pm$ 0.00	0.99 $\pm$ 0.01	4.70 $\pm$ 1.00	4.19 $\pm$ 1.00	T0	F0
PPO-Lag	27.13 $\pm$ 15.82	33.84 $\pm$ 22.98	15.23 $\pm$ 0.15	10.69 $\pm$ 3.45	0.02 $\pm$ 0.01	-0.29 $\pm$ 0.23	0.44 $\pm$ 0.04	0.25 $\pm$ 0.12	0.15 $\pm$ 0.07	1.00 $\pm$ 0.83	T0	F2
P3O	38.51 $\pm$ 17.15	52.49 $\pm$ 18.49	20.44 $\pm$ 1.16	14.93 $\pm$ 3.44	0.36 $\pm$ 0.08	-0.00 $\pm$ 0.23	0.71 $\pm$ 0.08	0.37 $\pm$ 0.09	0.54 $\pm$ 0.11	1.02 $\pm$ 0.64	T0	F2
FOCOPS	56.53 $\pm$ 11.59	63.78 $\pm$ 22.21	16.18 $\pm$ 0.27	12.52 $\pm$ 4.87	0.08 $\pm$ 0.02	-0.17 $\pm$ 0.32	0.42 $\pm$ 0.05	0.30 $\pm$ 0.15	0.37 $\pm$ 0.04	0.78 $\pm$ 0.88	T0	F2
$d = 25$												
PPO	57.13 $\pm$ 9.60	64.15 $\pm$ 10.40	85.11 $\pm$ 14.41	77.52 $\pm$ 14.28	2.40 $\pm$ 0.58	2.10 $\pm$ 0.57	1.00 $\pm$ 0.00	0.98 $\pm$ 0.03	2.40 $\pm$ 0.58	2.15 $\pm$ 0.56	T0	F0
PPO-Lag	33.45 $\pm$ 17.36	42.98 $\pm$ 22.80	25.18 $\pm$ 0.29	20.49 $\pm$ 4.85	0.01 $\pm$ 0.01	-0.18 $\pm$ 0.19	0.45 $\pm$ 0.04	0.31 $\pm$ 0.11	0.12 $\pm$ 0.10	0.69 $\pm$ 0.43	T0	F2
P3O	39.50 $\pm$ 17.30	50.25 $\pm$ 18.56	29.66 $\pm$ 2.10	23.55 $\pm$ 4.51	0.19 $\pm$ 0.08	-0.06 $\pm$ 0.18	0.64 $\pm$ 0.12	0.41 $\pm$ 0.10	0.34 $\pm$ 0.15	0.57 $\pm$ 0.12	T0	F2
FOCOPS	63.27 $\pm$ 12.00	76.30 $\pm$ 12.88	25.49 $\pm$ 0.34	21.43 $\pm$ 6.06	0.02 $\pm$ 0.01	-0.14 $\pm$ 0.24	0.39 $\pm$ 0.03	0.33 $\pm$ 0.13	0.23 $\pm$ 0.04	0.47 $\pm$ 0.42	T0	F2
$d = 50$												
PPO	55.57 $\pm$ 10.01	63.36 $\pm$ 10.15	86.52 $\pm$ 16.04	77.98 $\pm$ 16.49	0.73 $\pm$ 0.32	0.56 $\pm$ 0.33	0.99 $\pm$ 0.05	0.80 $\pm$ 0.15	0.73 $\pm$ 0.32	0.73 $\pm$ 0.26	T0	F0
PPO-Lag	42.73 $\pm$ 15.06	51.73 $\pm$ 17.17	48.78 $\pm$ 3.17	46.31 $\pm$ 5.81	-0.02 $\pm$ 0.06	-0.07 $\pm$ 0.12	0.39 $\pm$ 0.09	0.10 $\pm$ 0.08	0.10 $\pm$ 0.08	0.38 $\pm$ 0.09	T2	F2
P3O	43.69 $\pm$ 14.52	52.02 $\pm$ 15.01	51.35 $\pm$ 2.40	49.26 $\pm$ 3.03	0.03 $\pm$ 0.05	-0.01 $\pm$ 0.06	0.48 $\pm$ 0.16	0.43 $\pm$ 0.06	0.17 $\pm$ 0.14	0.43 $\pm$ 0.12	T0	F2
FOCOPS	71.08 $\pm$ 8.84	81.92 $\pm$ 7.79	48.83 $\pm$ 1.08	44.14 $\pm$ 4.86	-0.02 $\pm$ 0.02	-0.12 $\pm$ 0.10	0.33 $\pm$ 0.09	0.35 $\pm$ 0.10	0.09 $\pm$ 0.01	0.31 $\pm$ 0.06	T2	F2

Safe Button Point

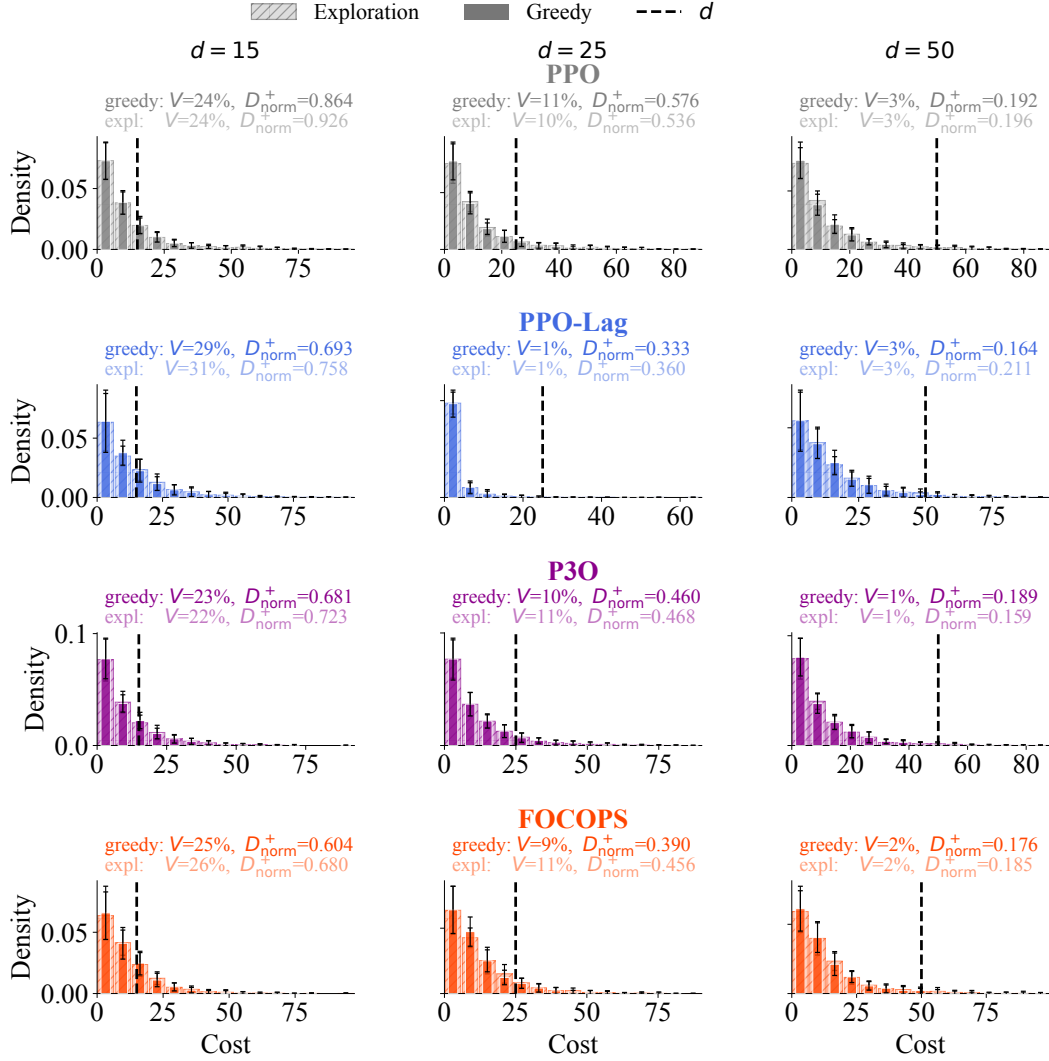


Figure A.10: Episodic cost distributions for final policies with exploration noise on and off (greedy). Safe Button Point (Level 1). Averaged over 30 seeds. Error bars denote 1σ over 30 seeds.

Table A.7: Safety and performance evaluation per metric for Safe Button Point (Level 1), for each individual $d \in \{15, 25, 50\}$. The numbers indicate mean $\pm 1\sigma$ over 30 seeds. **Bold text** indicates best result in that specific column, excluding PPO, as this is not a safe RL algorithm.

Algorithm	R		C		D_{norm}		V		D_{norm}^+		Tier	
	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)	Train	Final (greedy)
$d = 15$												
PPO	158.70 $\pm$ 8.70	2.47 $\pm$ 0.02	351.56 $\pm$ 30.21	10.77 $\pm$ 1.88	22.44 $\pm$ 2.01	-0.28 $\pm$ 0.13	1.00 $\pm$ 0.00	0.24 $\pm$ 0.06	22.44 $\pm$ 2.01	0.86 $\pm$ 0.26	T0	F2
PPO-Lag	121.04 $\pm$ 13.19	2.44 $\pm$ 0.11	16.02 $\pm$ 1.27	11.66 $\pm$ 4.82	0.07 $\pm$ 0.08	-0.22 $\pm$ 0.32	0.47 $\pm$ 0.11	0.29 $\pm$ 0.15	0.34 $\pm$ 0.11	0.69 $\pm$ 0.19	T0	F2
P3O	119.25 $\pm$ 9.73	2.47 $\pm$ 0.02	44.49 $\pm$ 5.53	9.63 $\pm$ 2.17	1.97 $\pm$ 0.37	-0.36 $\pm$ 0.14	1.00 $\pm$ 0.00	0.23 $\pm$ 0.08	1.97 $\pm$ 0.37	0.68 $\pm$ 0.20	T0	F2
FOCOPS	126.48 $\pm$ 7.90	2.46 $\pm$ 0.05	20.51 $\pm$ 0.56	10.58 $\pm$ 2.35	0.37 $\pm$ 0.04	-0.29 $\pm$ 0.16	0.80 $\pm$ 0.06	0.25 $\pm$ 0.07	0.48 $\pm$ 0.02	0.63 $\pm$ 0.27	T0	F2
$d = 25$												
PPO	159.05 $\pm$ 6.10	2.47 $\pm$ 0.02	349.05 $\pm$ 36.43	10.55 $\pm$ 2.08	12.96 $\pm$ 1.46	-0.58 $\pm$ 0.08	1.00 $\pm$ 0.00	0.11 $\pm$ 0.04	12.96 $\pm$ 1.46	0.58 $\pm$ 0.22	T0	F2
PPO-Lag	-0.17 $\pm$ 9.63	-8.84 $\pm$ 41.09	2.33 $\pm$ 0.46	2.11 $\pm$ 1.45	-0.91 $\pm$ 0.02	-0.92 $\pm$ 0.06	0.00 $\pm$ 0.00	0.01 $\pm$ 0.01	0.77 $\pm$ 0.32	0.33 $\pm$ 0.27	T2	F2
P3O	116.64 $\pm$ 10.90	2.47 $\pm$ 0.02	54.62 $\pm$ 5.23	10.42 $\pm$ 2.31	1.18 $\pm$ 0.21	-0.58 $\pm$ 0.09	0.99 $\pm$ 0.01	0.10 $\pm$ 0.04	1.20 $\pm$ 0.21	0.46 $\pm$ 0.18	T0	F2
FOCOPS	149.51 $\pm$ 7.69	2.46 $\pm$ 0.02	26.24 $\pm$ 0.35	10.81 $\pm$ 2.25	0.05 $\pm$ 0.01	-0.57 $\pm$ 0.09	0.41 $\pm$ 0.05	0.09 $\pm$ 0.05	0.28 $\pm$ 0.05	0.40 $\pm$ 0.18	T0	F2
$d = 50$												
PPO	158.01 $\pm$ 5.82	2.47 $\pm$ 0.02	353.01 $\pm$ 26.43	11.06 $\pm$ 2.15	6.06 $\pm$ 0.53	-0.78 $\pm$ 0.04	1.00 $\pm$ 0.00	0.03 $\pm$ 0.02	6.06 $\pm$ 0.53	0.20 $\pm$ 0.13	T0	F0
PPO-Lag	148.02 $\pm$ 12.79	2.46 $\pm$ 0.02	47.39 $\pm$ 1.97	13.82 $\pm$ 5.11	-0.05 $\pm$ 0.04	-0.72 $\pm$ 0.10	0.30 $\pm$ 0.11	0.03 $\pm$ 0.04	0.14 $\pm$ 0.08	0.20 $\pm$ 0.16	T2	F2
P3O	122.83 $\pm$ 10.21	2.47 $\pm$ 0.02	75.26 $\pm$ 4.41	9.91 $\pm$ 2.21	0.51 $\pm$ 0.09	-0.80 $\pm$ 0.04	0.93 $\pm$ 0.03	0.01 $\pm$ 0.01	0.56 $\pm$ 0.09	0.19 $\pm$ 0.19	T0	F2
FOCOPS	174.33 $\pm$ 6.08	2.45 $\pm$ 0.07	50.02 $\pm$ 0.12	13.23 $\pm$ 9.55	0.00 $\pm$ 0.00	-0.74 $\pm$ 0.19	0.41 $\pm$ 0.03	0.03 $\pm$ 0.07	0.08 $\pm$ 0.01	0.24 $\pm$ 0.35	T2	F2