CONFIDENCE-AWARE CHANCE-CONSTRAINED POLICY OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

In many real-world applications, reinforcement learning (RL) agents are expected to balance reward maximization and safety considerations. Constrained RL can address this issue by maximizing expected cumulative return while satisfying safety-related constraints. While RL algorithms with expected cumulative cost constraints have been extensively investigated, these expectation-based formulations provide a limited characterization of risk and do not directly capture the probability of safety violations. In this work, we propose a confidence-aware chance-constrained policy optimization (Chance-CPO) method that keeps the probability of constraint violation below a prescribed threshold at a user-specified confidence level. The proposed approach establishes a connection between expectation-based constraints and chance constraints, and uses expectation-related gradients to facilitate the satisfaction of chance constraints. By reducing an upper bound on the violation probability, Chance-CPO accounts for both the expected cumulative cost and the probability of unsafe events. We demonstrate the effectiveness of the proposed method on simulated robot tasks for which safety constraints are important.

1 INTRODUCTION

Deep reinforcement learning (RL) is promising for solving high-dimensional control problems, such as robotic locomotion Haarnoja et al. (2018); Radosavovic et al. (2024), autonomous driving Sallab et al. (2017), and game playing Mnih et al. (2015); Sorokin et al. (2015). In addition to high cumulative reward, safety is another important consideration from a practical point of view, since unsafe deployment may lead to catastrophic failures, physical damage, or unacceptable operational risks Jayant & Bhatnagar (2022); Chow et al. (2018a).

A standard framework for incorporating safety considerations into RL is the constrained Markov decision process (CMDP) Altman (2021), where the agent maximizes the expected cumulative reward subject to constraints that quantify safety-related costs. This framework has motivated a broad class of constrained RL algorithms, including primal-dual methods Ray et al. (2019); Jayant & Bhatnagar (2022); Ding et al. (2021), trust-region-based approaches Achiam et al. (2017), Lyapunov-based methods Chow et al. (2018b), and projection-based approaches Yang et al. (2020). Most existing constrained RL methods rely on expectation-based constraints, which require the expected cumulative cost to remain below a threshold Ray et al. (2019); Jayant & Bhatnagar (2022); Achiam et al. (2017); Chow et al. (2018b); Yang et al. (2020). While optimization problems with such constraints may be handled using gradient-based methods, the expectation-based constraints provide only a limited characterization of risk. In particular, they do not directly control the probability of violating the safety requirement; this violation probabability is a key risk criterion in safety-critical RL applications Jung et al. (2022).

Chance constraints provide a direct way to regulate safety by requiring the violation probability to remain below a prescribed threshold Nemirovski & Shapiro (2007); Ono & Williams (2008). However, chance-constrained optimization is generally difficult to solve, especially in high-dimensional decision-making, because chance constraints usually introduce non-convexity, and gradients of violation probabilities with respect to policy parameters can be difficult to obtain Paternain et al. (2022); Chen et al. (2024). A practical approach is to replace the chance constraint with a tractable surrogate, such as a conditional value-at-risk (CVaR) constraint Nemirovski & Shapiro (2007). The resulting

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

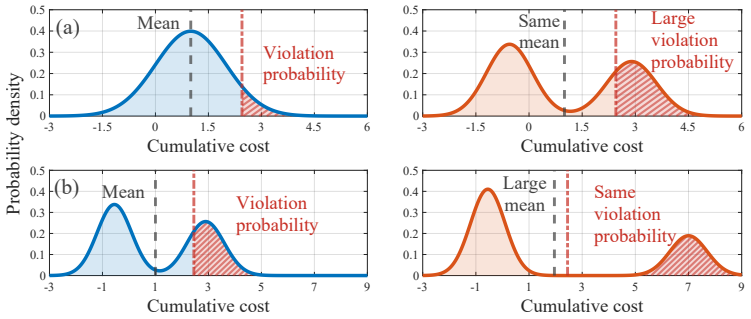


Figure 1: Cost distributions: (a) distributions with the same mean can have different violation probabilities; (b) distributions with the same violation probability can have different means.

RL problems with CVaR constraints can then be addressed using trust-region-based methods Kim & Oh (2022) or policy-gradient-based methods Chow et al. (2018a). Nevertheless, CVaR-based surrogates can be overly conservative as compared to the original chance constraint. Recently, methods such as quantile-constrained policy optimization (QCPO) Jung et al. (2022) have used distributional RL Bellemare et al. (2017); Dabney et al. (2018) to characterize the cost distribution through quantile estimation. While this facilitates chance-constrained policy optimization, the accuracy of quantile estimation can depend on the choice of quantile fractions Yang et al. (2019). QCPO may not adapt well across different environments, as it uses fixed quantile fractions. Moreover, approximation of the right tail of the cost distribution and cost-distribution shifts induced by policy updates may introduce additional errors that affect the reliability of chance-constraint satisfaction.

These approaches highlight two challenges in chance-constrained optimization: conservative approximations and uncertainty in constraint satisfaction when violation probabilities are estimated from data Nemirovski & Shapiro (2007); Jung et al. (2022). Moreover, although chance constraints provide a more informative risk measure by explicitly characterizing the safety violation probability, they do not account for the severity of safety violations. Rare violations with very high costs can lead to a substantial increase in expected cumulative cost, which suggests that both the expected cumulative cost and the probability of unsafe events are important for risk management. To address these challenges, we propose confidence-aware chance-constrained policy optimization (Chance-CPO), which explicitly controls the violation probability at a user-specified confidence level by leveraging gradient information associated with the expected cumulative cost. We demonstrate the effectiveness of the proposed Chance-CPO on simulated robotic control tasks, and we compare this method with two representative constrained RL baselines. The key contributions of this paper are summarized as follows:

- We derive an upper bound on the violation probability based on the expected cumulative cost by scaling the expectation with a designed scaling parameter;
- We propose a confidence-aware chance-constrained policy optimization (Chance-CPO) approach that uses expectation-related gradients to promote satisfaction of chance constraints;
- We evaluate the proposed Chance-CPO on simulated robotic control tasks with safety requirements across multiple random seeds. The results show lower mean cumulative costs and empirical violation percentages than the baselines in the reported evaluations.

2 Related Work

Constrained reinforcement learning Constrained reinforcement learning (RL) has typically been studied under the framework of constrained Markov decision processes (CMDPs) Altman (2021), where the Markov decision process is augmented with a set of cost functions that evaluate safety. Many constrained RL methods aim to find an optimal policy that maximizes the expected return subject to constraints on the expectations of discounted cumulative costs Achiam et al. (2017); Liu et al. (2020); Ding et al. (2021); Jayant & Bhatnagar (2022); Yang et al. (2020). In Achiam et al. (2017), a policy search algorithm called constrained policy optimization (CPO) was proposed by

introducing a trust region. To scale to high-dimensional systems, CPO Achiam et al. (2017) approximates the formulated optimization problem by a tractable convex subproblem at each policy update. Constrained proximal policy optimization adopts a primal–dual framework Jayant & Bhatnagar (2022) to extend proximal policy optimization (PPO) Schulman et al. (2017) to constrained settings, incorporating cost constraints into the objective through Lagrange multipliers.

Risk measures in reinforcement learning Although many constrained RL methods consider expected cumulative costs Ray et al. (2019); Jayant & Bhatnagar (2022); Achiam et al. (2017); Chow et al. (2018b); Yang et al. (2020); Liu et al. (2020), such constraints cannot explicitly account for the distribution of cumulative costs induced by a specific policy. Figure 1(a) provides an illustration of this limitation: two cost distributions with the same expectation can exhibit substantially different probabilities of violating a safety threshold. This limitation motivated risk-aware formulations that account for risk beyond the expected cumulative cost. Value-at-risk (VaR) and conditional value-at-risk (CVaR) are two promising alternatives for characterizing the tail of a cost distribution Rockafellar & Uryasev (2002); Chow et al. (2018a). The probabilistic constraints provide stricter restrictions by requiring the entire trajectory to remain in the safe set with high probability Geibel (2006); Paternain et al. (2022); Chen et al. (2024). Such constraints can be overly conservative for applications where occasional positive safety costs are acceptable. Another widely adopted measure is the chance constraint, which provides a natural formulation for settings where safety requirements are specified by an allowable frequency of unacceptable outcomes Nemirovski & Shapiro (2007). Chance constraints accommodate occasional positive safety costs requiring the probability of the cumulative cost exceeding a prescribed threshold to remain below a specified limit Nemirovski & Shapiro (2007); Ono & Williams (2008); Jung et al. (2022).

Chance-constrained reinforcement learning Optimization problems with chance constraints can be computationally intractable because chance constraints may induce non-convex feasible sets Nemirovski & Shapiro (2007). A viable strategy in stochastic programming is to replace these constraints with tractable conservative approximations, such as adopting a CVaR constraint Nemirovski & Shapiro (2007). In reinforcement learning, quantile-constrained policy optimization (QCPO) Jung et al. (2022) exploits the connection between chance constraints and VaR constraints and learns a quantile function of the cumulative cost using distributional RL Bellemare et al. (2017).

3 Preliminaries and problem formulation

3.1 Preliminaries

Let $\mathbb{N}$ denote the set of nonnegative integers, $\mathbb{R}$ denote the set of real numbers, and $\mathbb{R}_{\geq 0}$ denote the set of nonnegative real numbers. Consider a stochastic system with unknown underlying dynamics, modeled as a constrained Markov decision process (CMDP). Let $t \in \mathbb{N}$ denote the time index. At each time instant t, given the system state $s_t \in \mathcal{S} \subseteq \mathbb{R}^{n_s}$, the control input $a_t \in \mathcal{A} \subseteq \mathbb{R}^{n_a}$ is generated based on the policy that is defined by a conditional distribution $\pi(a_t|s_t)$, where $\mathcal{S}$ and $\mathcal{A}$ are the state space and action space, respectively.

Let d^π denote the trajectory distribution induced by policy π. Specifically, a trajectory $\kappa = (s_0, a_0, s_1, a_1, \ldots) \sim d^\pi$ is generated according to $s_0 \sim \mathbb{P}_0(s_0 \in \mathcal{S})$, $a_t \sim \pi(\cdot \mid s_t)$, and $s_{t+1} \sim \mathbb{P}(\cdot \mid s_t, a_t)$, where $\mathbb{P}_0(s_0 \in \mathcal{S})$ is the initial state distribution and $\mathbb{P}(\cdot \mid s_t, a_t)$ denotes the state transition probability. For a given trajectory κ, its performance is characterized by a discounted cumulative reward $J_r(\kappa) = \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, s_{t+1})$ and a discounted cumulative cost $J_c(\kappa) = \sum_{t=0}^{\infty} \gamma^t c(s_t, a_t, s_{t+1})$, where $\gamma \in (0, 1)$ denotes the discount factor, $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the reward function, and $c : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}_{\geq 0}$ is the safety-related cost function. The expected discounted cumulative reward and cost of policy π are then given by $\eta_r(\pi) = \mathbb{E}_{\kappa \sim d^\pi}[J_r(\kappa)]$ and $\eta_c(\pi) = \mathbb{E}_{\kappa \sim d^\pi}[J_c(\kappa)]$. The reward value function V_r^π, the reward state-action value function Q_r^π, and the reward advantage function A_r^π are defined as $V_r^\pi(s) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, s_{t+1}) \mid s_0 = s]$, $Q_r^\pi(s, a) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, s_{t+1}) \mid s_0 = s, a_0 = a]$, and $A_r^\pi(s, a) = Q_r^\pi(s, a) - V_r^\pi(s)$, respectively. Similarly, the cost value function V_c^π, the cost state-action value function Q_c^π, and the cost advantage function A_c^π can be defined with respect to the cost function c.

Let the discounted state visitation distribution under a policy π be defined as $\rho^\pi(s) = \sum_{t=0}^{\infty} \gamma^t \mathcal{P}(s_t = s|\pi)$. The difference in expected discounted cumulative reward between policies $\tilde{\pi}$

and π can be expressed as follows Kakade & Langford (2002):

$$\eta_r(\tilde{\pi}) - \eta_r(\pi) = \mathbb{E}_{s\sim\rho^{\tilde{\pi}}, a\sim\tilde{\pi}(a|s)}[A_r^{\pi}(s,a)] \tag{1}$$

(1) involves the discounted state visitation distribution $\rho^{\tilde{\pi}}$ induced by $\tilde{\pi}$, which is generally unavailable without executing $\tilde{\pi}$. A common practice is to approximate the expectation in (1) with the available state visitation distribution ρ^{π} associated with π. A well-known result is that the difference in expected performance between two policies can be bounded in terms of the advantage and the total variation divergence between them Schulman et al. (2015a); Achiam et al. (2017). This relationship is formally stated in the following theorem.

Theorem 1 (Schulman et al. (2015a); Achiam et al. (2017)) *For two policies π and $\tilde{\pi}$, let $A_r^{\pi}(s,a)$ denote the advantage function associated with π. Define $\epsilon_r = \max_s |\mathbb{E}_{a\sim\tilde{\pi}(a|s)}[A_r^{\pi}(s,a)]|$ and $L_r^{\pi}(\tilde{\pi}) = \mathbb{E}_{s\sim\rho^{\pi}, a\sim\pi(a|s)}[(\frac{\tilde{\pi}(a|s)}{\pi(a|s)} - 1)A_r^{\pi}(s,a)]$. The following bounds on $\eta_r(\tilde{\pi}) - \eta_r(\pi)$ hold:*

$$L_r^{\pi}(\tilde{\pi}) + C\mathbb{E}_{s\sim\rho^{\pi}}[D_{TV}(\pi\|\tilde{\pi})[s]] \geq \eta_r(\tilde{\pi}) - \eta_r(\pi) \geq L_r^{\pi}(\tilde{\pi}) - C\mathbb{E}_{s\sim\rho^{\pi}}[D_{TV}(\pi\|\tilde{\pi})[s]] \tag{2}$$

where $C = \frac{2\epsilon_r\gamma}{(1-\gamma)^2}$, and $D_{TV}(\pi\|\tilde{\pi})[s] = \frac{1}{2}\sum_a |\tilde{\pi}(a|s) - \pi(a|s)|$ is the total variation divergence between two policies at state s.

3.2 Problem formulation

In this work, we consider a control optimization problem for a CMDP with a chance constraint. Specifically, we aim to find a policy π_θ parameterized by θ, which maximizes the expectation of discounted cumulative reward $\eta_r(\pi_\theta)$, while satisfying a chance constraint on the discounted cumulative cost. This constraint is defined by a prescribed cost threshold $d \in \mathbb{R}_{\geq 0}$ and violation probability threshold $\alpha \in (0, 1]$, and is required to hold with a confidence level $1 - \beta \in [0, 1]$.

To simplify the notation, we use $J_c(\pi_\theta)$ to represent the discounted cumulative cost of a trajectory κ induced by π_θ, and the violation probability with respect to d is defined as $\mathcal{P}\left(J_c(\pi_\theta) \geq d\right) := \mathcal{P}_{\kappa\sim d^{\pi_\theta}}\left(J_c(\kappa) \geq d\right)$. The problem of interest is then summarized as follows:

$$\max_{\theta} \quad \eta_r(\pi_\theta) \tag{3a}$$

$$\text{s.t.} \quad \mathcal{P}\Big(\mathcal{P}\big(J_c(\pi_\theta) \geq d\big) \leq \alpha\Big) \geq 1 - \beta \tag{3b}$$

where the discounted cumulative cost threshold d, the violation probability threshold α, and the confidence level $1 - \beta$ are all user-specified parameters.

A promising approach to solving a constrained RL problem is to formulate an unconstrained problem within the primal–dual framework and apply gradient descent methods to obtain a locally optimal solution Jayant & Bhatnagar (2022); Chen et al. (2024); Paternain et al. (2022). However, the gradient $\nabla_\theta \mathcal{P}\big(J_c(\pi_\theta) \geq d\big)$ of the violation probability in (3b) is in general intractable to compute, which poses a challenge for applying gradient-based optimization methods.

Additionally, while the chance constraint in (3b) is a widely used measure of handling safety, it only constrains the probability of constraint violation yet it does not characterize the severity of such violations Chow et al. (2018a). In particular, satisfying a chance constraint does not impose restrictions on the magnitude of cumulative costs. Figure 1(b) provides an illustration of this type of case scenario: a policy may satisfy the chance constraint while still allowing unsafe trajectories to incur arbitrarily high costs, which results in increased expectations. This observation highlights the importance of considering both the expected cost and the violation probability when assessing the overall safety level of a policy.

In this work, we propose a confidence-aware chance-constrained policy optimization approach, referred to as Chance-CPO, which enforces the chance constraint at a confidence level $1 - \beta$ while simultaneously reducing the expected discounted cumulative cost, to prompt an overall safer policy. Importantly, the proposed Chance-CPO provides a gradient-based optimization scheme for solving the optimization problem with chance constraints.

4 CONFIDENCE-AWARE CHANCE-CONSTRAINED POLICY OPTIMIZATION

4.1 SURROGATE OPTIMIZATION PROBLEM

To satisfy the chance constraint in (3b) while reducing the expected cost simultaneously, we seek to establish a connection between the expected cumulative cost $\eta_c(\pi_\theta)$ and the violation probability $\mathcal{P}\big(J_c(\pi_\theta) \geq d\big)$ in a way such that both criteria can be incorporated into a unified optimization framework. A classical tool for relating probabilities to expectations is Markov's inequality Patrick (1995):

$$\mathcal{P}\big(J_c(\pi_\theta) \geq d\big) \leq \frac{\eta_c(\pi_\theta)}{d} \tag{4}$$

Based on (4), replacing the violation probability $\mathcal{P}\big(J_c(\pi_\theta) \geq d\big)$ in (3b) with the scaled expected cost on the right-hand side of (4) yields a sufficient condition for enforcing the chance constraint. However, the upper bound provided by Markov's inequality can be overly conservative. In particular, policy π_θ that satisfies $\frac{\eta_c(\pi_\theta)}{d} \leq \alpha$ ensures the satisfaction of the chance constraint regardless of the shape of the cost distribution.

Although the probability bound in (4) can be loose, Markov's inequality suggests that the scaled expectation of the discounted cumulative cost $\eta_c(\pi)$ can provide an upper bound on the violation probability $\mathcal{P}\big(J_c(\pi_\theta) \geq d\big)$ by choosing an appropriate scaling parameter. Building on this observation, in this work, we formulate a surrogate optimization problem of (3) as follows:

$$\max_{\theta} \quad \eta_r(\pi_\theta) \tag{5a}$$

$$\text{s.t.} \quad \eta_c(\pi_\theta) \leq \delta\alpha \tag{5b}$$

where the scaling parameter δ is chosen such that $\frac{\eta_c(\pi_\theta)}{\delta}$ provides a valid yet less conservative upper bound on the violation probability in (3b) at a confidence level $1-\beta$. By introducing the surrogate problem in (5), the chance-constrained optimization problem is converted into a form commonly considered in constrained RL, enabling the use of existing approaches, such as CPO Achiam et al. (2017) and constrained PPO Jayant & Bhatnagar (2022); Ray et al. (2019).

Inspired by RL frameworks that perform iterative policy updates and rely on system interactions Achiam et al. (2017); Jayant & Bhatnagar (2022); Ray et al. (2019), in this work, we collect data using the old policy obtained in the previous iteration and perform policy updates for several gradient steps using the same dataset. Specifically, at the kth iteration, a dataset $\mathcal{D}_k$ is collected by applying the policy $\pi_{\theta_{k-1}}$ obtained at the $(k-1)$th iteration. The dataset $\mathcal{D}_k$ is then used to estimate a candidate scaling parameter at the kth iteration and perform multiple gradient steps, which update the policy from $\pi_{\theta_{k-1}}$ to π_{θ_k}. Let $J_c^i(\pi_{\theta_{k-1}})$ denote the discounted cumulative cost of the ith trajectory in $\mathcal{D}_k$. Let N denote the number of trajectories included in each dataset, and let n_k denote the number of unsafe trajectories in $\mathcal{D}_k$, defined as those whose discounted cumulative cost is at least d. An empirical violation probability $\hat{\mathcal{P}}\big(J_c(\pi_{\theta_{k-1}}) \geq d\big)$ obtained based on $\mathcal{D}_k$ can be computed as $\hat{\mathcal{P}}\big(J_c(\pi_{\theta_{k-1}}) \geq d\big) = \frac{n_k+1}{N+1}$. An estimated scaling parameter $\hat{\delta}_k$ can then be defined as follows:

$$\hat{\delta}_k = \frac{\eta_c(\pi_{\theta_{k-1}})(N+1)}{n_k+1} \tag{6}$$

Based on (6), $\pi_{\theta_{k-1}}$ that satisfies $\eta_c(\pi_{\theta_{k-1}}) \leq \hat{\delta}_k\alpha$ also ensures that $\hat{\mathcal{P}}\big(J_c(\pi_{\theta_{k-1}}) \geq d\big) \leq \alpha$, which suggests that an appropriate surrogate optimization problem at the kth iteration can be formulated by replacing δ in (5b) with $\hat{\delta}_k$.

Following Schulman et al. (2015a; 2017); Achiam et al. (2017); Jayant & Bhatnagar (2022); Jung et al. (2022), we may reuse both the dataset $\mathcal{D}_k$ and the estimated scaling parameter $\hat{\delta}_k$ corresponding to policy $\pi_{\theta_{k-1}}$ in (6) through all gradient steps within this iteration. However, two sources of error may prevent the scaled expected cost based on $\hat{\delta}_k$ from upper-bounding the true violation probability:

- Only finite samples are included in $\mathcal{D}_k$. Therefore, mismatch between the empirical probability and the exact probability may exist. Consequently, $\pi_{\theta_{k-1}}$ that satisfies $\eta_c(\pi_{\theta_{k-1}}) \leq \hat{\delta}_k\alpha$ may not reliably ensure the satisfaction of chance constraints.

- Multiple gradient steps within one iteration induce a cumulative cost distribution shift, which may sacrifice the validity of reusing $\hat{\delta}_k$ throughout the whole iteration. Since $\hat{\delta}_k$ is estimated from $\mathcal{D}_k$ collected by using $\pi_{\theta_{k-1}}$, it may fail to provide a valid upper bound for the violation probability of π_θ, and thus potentially compromise the constraint guarantee.

We next introduce an approach for selecting the scaling parameter δ in (5) that accounts for these errors. With this choice of δ, any feasible solution to the surrogate constrained optimization problem (5) also satisfies the chance constraint with probability at least $1-\beta$, as specified in (3b).

4.2 Scaling parameter selection

In this section, we propose a method for selecting δ in (5b) such that $\frac{\eta_c(\pi_\theta)}{\delta}$ provides a valid upper bound on $\mathcal{P}\big(J_c(\pi_\theta) \geq d\big)$ in (3b) with a probability no less than $1-\beta$. Specifically, we account for finite-sample error by analyzing the statistical properties of trajectories in $\mathcal{D}_k$ and constructing δ_k that results in a upper bound more conservative than the empirically estimated probability associated with $\hat{\delta}_k$. We then introduce a possible choice of δ that adapts with the policy update by refining δ_k with adaptive correction terms. The selected δ ensures that the violation probability remains under control despite multiple gradient steps and dataset reuse within each iteration.

Firstly, we construct δ_k based on the upper bound of the Clopper–Pearson interval Clopper & Pearson (1934); Thulin (2014) to account for the error arising from the finite number of samples. Let $\mathrm{BetaInv}(q; v, w)$ represent the q-quantile of the Beta distribution $\mathrm{Beta}(v, w)$ Johnson et al. (1997). Define

$$\delta_k := \max\Big\{\frac{\eta_c(\pi_{\theta_{k-1}})}{\mathrm{BetaInv}(1-\beta; n_k+1, N-n_k)}, d\Big\} \tag{7}$$

where $\max\{a,b\} := a$ if $a \geq b$, and $\max\{a,b\} := b$ otherwise. When $n_k = N$, $\mathrm{BetaInv}(1-\beta; n_k+1, N-n_k)$ is not well defined. In this case, we set $\delta_k = d$. In the following proposition, we show that δ_k in (7) induces an upper bound on $\mathcal{P}\big(J_c(\pi_{\theta_{k-1}}) \geq d\big)$ with a probability at least $1-\beta$.

Proposition 1 *If δ_k is in the form of (7), any $\pi_{\theta_{k-1}}$ satisfying $\eta_c(\pi_{\theta_{k-1}}) \leq \delta_k\alpha$ also satisfies $\mathcal{P}\Big(\mathcal{P}\big(J_c(\pi_{\theta_{k-1}}) \geq d\big) \leq \alpha\Big) \geq 1-\beta$.*

The proof of this proposition is presented in Appendix A.1.

The definition of δ_k in (7) selects the tighter one between the Markov-inequality bound and the Clopper–Pearson upper confidence bound, providing a valid upper bound at the given confidence level $1-\beta$ while formulating a less conservative surrogate problem.

Next, we address the cost distribution shift induced by the multiple-step policy updates. In the following, we introduce an adaptive correction approach for δ_k, which is applied when δ_k in (7) is selected to be $\frac{\eta_c(\pi_{\theta_{k-1}})}{\mathrm{BetaInv}(1-\beta; n_k+1, N-n_k)}$. This correction is designed to preserve the effectiveness of the scaling parameter under policy updates.

We design two auxiliary system states, $y_t \in \mathbb{R}$ and $w_t \in \mathbb{R}$, initialized as $y_0 = d$ and $w_0 = 1$, respectively. Their dynamics are defined as follows:

$$y_{t+1} = \frac{y_t - c(s_t, a_t, s_{t+1})}{\gamma}; \quad w_{t+1} = \frac{w_t}{\gamma} \tag{8}$$

where $t \in \mathbb{N}$, y_t represents the cost margin, and w_t is an inverse-discount scaling factor. Appendix B explains the cost-margin construction. Additionally, we define an augmented MDP with the system state $x_t = [s_t^\top \; y_t \; w_t]^\top$, policy $\pi(a_t|x_t) = \pi(a_t|s_t)$, and an auxiliary cost function $\bar{c}(x_t, a_t, x_{t+1})$:

$$\bar{c}(x_t, a_t, x_{t+1}) = w_t \mathbb{1}(y_t > 0 \;\wedge\; y_{t+1} \leq 0) := \begin{cases} w_t, & y_t > 0 \text{ and } y_{t+1} \leq 0 \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where $\wedge$ denotes logical conjunction.

Let $d_x^{\pi_\theta}$ denote the trajectory distribution of the augmented MDP with state $x_t = [s_t^\top \; y_t \; w_t]^\top$ induced by $a_t \sim \pi_\theta(a_t|s_t)$, $s_0 \sim \mathbb{P}_0(s_0 \in \mathcal{S})$, $s_{t+1} \sim \mathbb{P}(s_{t+1}|s_t, a_t)$ and the auxiliary dynamics

in (8). For a specific trajectory of the augmented MDP $\kappa_x = (x_0, a_0, x_1, a_1, \ldots) \sim d_x^{\pi_\theta}$, define the corresponding discounted cumulative auxiliary cost $J_{\bar{c}}(\kappa_x)$ as follows:

$$J_{\bar{c}}(\kappa_x) = \sum_{t=0}^{\infty} \gamma^t \bar{c}(x_t, a_t, x_{t+1}) = \sum_{t=0}^{\infty} \gamma^t w_t \mathbb{1}(y_t > 0 \ \wedge \ y_{t+1} \leq 0) \tag{10}$$

Based on this augmented MDP construction, we introduce the following proposition.

Proposition 2 *$\mathcal{P}(J_c(\pi_\theta) \geq d) = \eta_{\bar{c}}(\pi_\theta)$, where $\eta_{\bar{c}}(\pi_\theta) = \mathbb{E}_{\kappa_x \sim d_x^{\pi_\theta}}[J_{\bar{c}}(\kappa_x)]$.*

The proof of this proposition is presented in Appendix A.2.

For notational simplicity, let $L_{\bar{c}}^{\pi_{\theta_{k-1}}}(\pi_\theta) = \mathbb{E}_{x \sim \rho_x^{\pi_{\theta_{k-1}}}, a \sim \pi_{\theta_{k-1}}(a|s)}[(\frac{\pi_\theta(a|s)}{\pi_{\theta_{k-1}}(a|s)} - 1) A_{\bar{c}}^{\pi_{\theta_{k-1}}}(x, a)]$ and $C_{\bar{c}} = \frac{2\gamma \max_x |\mathbb{E}_{a \sim \pi_\theta(a|s)}[A_{\bar{c}}^{\pi_{\theta_{k-1}}}(x,a)]|}{(1-\gamma)^2}$, where $\rho_x^{\pi_{\theta_{k-1}}}$ is the discounted state visitation distribution of the augmented MDP under $\pi_{\theta_{k-1}}$, and $A_{\bar{c}}^{\pi_{\theta_{k-1}}}(x, a)$ is the auxiliary cost advantage function.

Based on Theorem 1 and Propositions 1-2, we correct the scaling parameter from δ_k in (7) to $\delta(\theta)$ following:

$$\delta(\theta) = \max\{\frac{\eta_c(\pi_{\theta_{k-1}}) + L_c^{\pi_{\theta_{k-1}}}(\pi_\theta) - C_c \mathbb{E}_{s \sim \rho^{\pi_{\theta_{k-1}}}}[D_{\text{TV}}(\pi_{\theta_{k-1}} \| \pi_\theta)[s]]}{\text{BetaInv}(1-\beta; n_k + 1, N - n_k) + L_{\bar{c}}^{\pi_{\theta_{k-1}}}(\pi_\theta) + C_{\bar{c}} \mathbb{E}_{s \sim \rho^{\pi_{\theta_{k-1}}}}[D_{\text{TV}}(\pi_{\theta_{k-1}} \| \pi_\theta)[s]]}, d\} \tag{11}$$

Theorem 2 *Based on Theorem 1 and Propositions 1-2, if δ in (5) takes the form of (11), then any feasible solution to (5) is also a feasible solution to (3).*

The proof of this theorem is shown in Appendix A.3.

5 Practical implementation

The problem (5) with the scaling parameter in (11) provides a rigorous guarantee but is difficult to optimize directly. In this section, we propose a practical algorithm for efficient policy updates. Specifically, we consider a tractable approximation with PPO-based clipped surrogates for the objective and constraint, which enables efficient optimization via a primal-dual approach.

Following Schulman et al. (2017), we modify the optimization problem (5) by rewriting the objective and constraint based on advantages and introducing a clipped policy probability ratio as follows:

$$\max_{\pi_\theta} \mathbb{E}_{\mathcal{D}_k}[\min\big(p_t(\theta)\hat{A}_t^r, \text{clip}(p_t(\theta), 1-\xi, 1+\xi)\hat{A}_t^r\big)] \tag{12a}$$

$$\text{s.t. } \mathbb{E}_{\mathcal{D}_k}[J_c(\pi_{\theta_{k-1}})] - \mathbb{E}_{\mathcal{D}_k}[\hat{A}_t^c] + \mathbb{E}_{\mathcal{D}_k}[\min\big(p_t(\theta)\hat{A}_t^c, \text{clip}(p_t(\theta), 1-\xi, 1+\xi)\hat{A}_t^c\big)] \leq \hat{\delta}(\theta)\alpha \tag{12b}$$

with

$$\hat{\delta}(\theta) = \max\{\frac{\mathbb{E}_{\mathcal{D}_k}[J_c(\pi_{\theta_{k-1}})] + \mathbb{E}_{\mathcal{D}_k}[(p_t(\theta) - 1)\hat{A}_t^c]}{\text{BetaInv}(1-\beta; n_k + 1, N - n_k) + \mathbb{E}_{\mathcal{D}_k}[(p_t(\theta) - 1)\hat{A}_t^{\bar{c}}]}, d\} \tag{13}$$

where $p_t(\theta) = \frac{\pi_\theta(a|s)}{\pi_{\theta_{k-1}}(a|s)}$ denotes the policy probability ratio; $\text{clip}(\cdot, 1-\xi, 1+\xi)$ clips $p_t(\theta)$ to the interval $[1-\xi, 1+\xi]$; $\mathbb{E}_{\mathcal{D}_k}[J_c(\pi_{\theta_{k-1}})]$ is the empirical estimate of $\eta_c(\pi_{\theta_{k-1}})$; and $\hat{A}_t^r$, $\hat{A}_t^c$, and $\hat{A}_t^{\bar{c}}$ denote the advantages estimated based on $\mathcal{D}_k$ using learned value functions and generalized advantage estimation (GAE) Mnih et al. (2016); Schulman et al. (2015b; 2017). Details of GAE are provided in Appendix C.1. While the clipping operation in equation 12b may weaken the previous upper-bound guarantee, it provides a practical way for controlling the size of policy updates. A similar design has also been adopted in related safe RL work Jayant & Bhatnagar (2022).

For notational simplification, let $\Phi_r(\theta) := \mathbb{E}_{\mathcal{D}_k}[\min\big(p_t(\theta)\hat{A}_t^r, \text{clip}(p_t(\theta), 1-\xi, 1+\xi)\hat{A}_t^r\big)]$ and $\Phi_c(\theta) := \mathbb{E}_{\mathcal{D}_k}[\min\big(p_t(\theta)\hat{A}_t^c, \text{clip}(p_t(\theta), 1-\xi, 1+\xi)\hat{A}_t^c\big)]$. The Lagrangian of the constrained optimization problem (12) is as follows:

$$L(\theta, \lambda) = \Phi_r(\theta) - \lambda\big(\Phi_c(\theta) + \mathbb{E}_{\mathcal{D}_k}[J_c(\pi_{\theta_{k-1}})] - \mathbb{E}_{\mathcal{D}_k}[\hat{A}_t^c] - \hat{\delta}(\theta)\alpha\big) \tag{14}$$

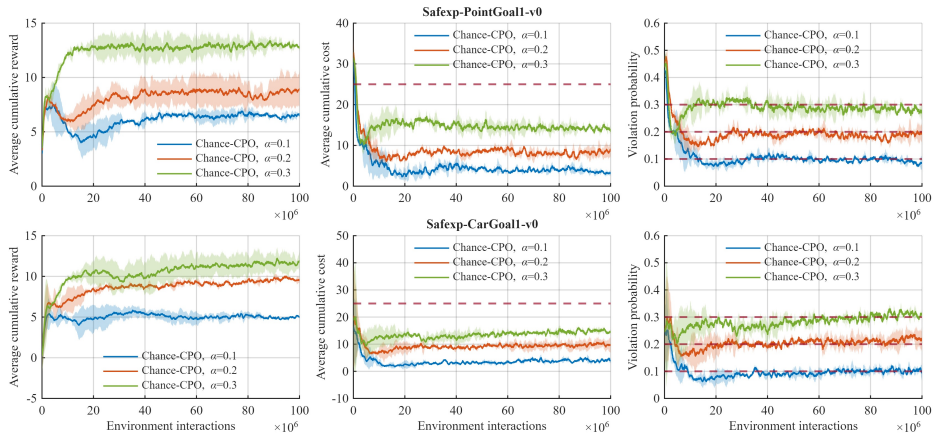


Figure 2: Performance of the proposed Chance-CPO in two environments with different violation probability thresholds $\alpha = 0.1, 0.2, 0.3$.

where $\lambda \in \mathbb{R}_+$ is the Lagrange multiplier associated with the constraint. (14) can be solved efficiently via a gradient search procedure. Specifically, let E denote the number of gradient steps per iteration. For the ith gradient step in the kth iteration, parameters θ and λ are updated as follows:

$$\theta_{i,k} = \theta_{i-1,k} + l_\theta \nabla_\theta L(\theta, \lambda_{i-1,k}) = \theta_{i-1,k} + l_\theta \big(\nabla_\theta \Phi_r(\theta) - \lambda_{i-1,k} \nabla_\theta \Phi_c(\theta)\big) \tag{15a}$$

$$\begin{aligned}\lambda_{i,k} &= [\lambda_{i-1,k} - l_\lambda \nabla_\lambda L(\theta_{i,k}, \lambda)]_+ \\ &= \big[\lambda_{i-1,k} + l_\lambda \big(\Phi_c(\theta_{i,k}) + \mathbb{E}_{\mathcal{D}_k}[J_c(\pi_{\theta_{k-1}})] - \mathbb{E}_{\mathcal{D}_k}[\hat{A}_t^c] - \hat{\delta}(\theta_{i,k})\alpha\big)\big]_+\end{aligned} \tag{15b}$$

where $\theta_{0,k} = \theta_{k-1}$, and $\lambda_{0,k} = \lambda_{k-1}$; l_θ and l_λ denote the learning rates of θ and λ, respectively. While the scaling parameter $\delta(\theta)$ also depends on θ, we do not backpropagate through δ when updating θ, due to the causal relationship between the two variables.

Additionally, $\mathbb{E}_{\mathcal{D}_k}[J_c(\pi_{\theta_{k-1}})]$ in (12b) and (13) is used to estimate $\eta_c(\pi_{\theta_{k-1}})$, which denotes the expected cumulative cost over an infinite horizon, but $\mathcal{D}_k$ contains only a finite number of steps. To address this issue, following Proposition 2 in Paternain et al. (2020) and Remark 2 in Paternain et al. (2022), we sample the length T of each trajectory in $\mathcal{D}_k$ from a geometric distribution with success probability $1-\gamma$, i.e., $\mathcal{P}(T = t) = (1-\gamma)\gamma^t$, so that the cumulative cost without discount in a finite horizon T provides an unbiased estimate of $\eta_c(\pi_{\theta_{k-1}})$ as shown in Appendix C.2. The training procedure of Chance-CPO is summarized in Appendix D.

6 Experiments

We assess the performance of the proposed Chance-CPO on environments with different robot morphologies from Safety Gym, including PointGoal1-v0 and CarGoal1-v0. Detailed descriptions are given in Appendix E.1. We conduct simulations with three different violation probability thresholds $\alpha \in \{0.1, 0.2, 0.3\}$ and set the discounted cumulative cost threshold to $d = 25$. Hyperparameter values of Chance-CPO are presented in Appendix E.2. Additionally, we compare the proposed method with two baseline methods: constrained PPO solved via Lagrangian relaxation (referred to as CPPO-Lag) Ray et al. (2019) and quantile-constrained policy optimization (referred to as QCPO) Jung et al. (2022). For both baseline methods, we set $d = 25$. For QCPO Jung et al. (2022), we consider the violation probability threshold $\alpha = 0.3$. For all approaches and configurations, we run 5 independent trials in each environment using 5 different seeds. The results are shown in Figures 2 and 3, where the solid curves show the mean over 5 runs and the shaded regions indicate the corresponding standard deviation. Additional simulation results on trained-policy evaluation and ablation studies are presented in Appendices F and G. Simulation results on the DoggoGoal1-v0 and button tasks (PointButton1-v0, CarButton1-v0, and DoggoButton1-v0) are presented in Appendix H.

Figure 2 shows the performance of the proposed Chance-CPO under three different violation probability thresholds α in two environments. The right column in Figure 2 reports the Clopper–Pearson

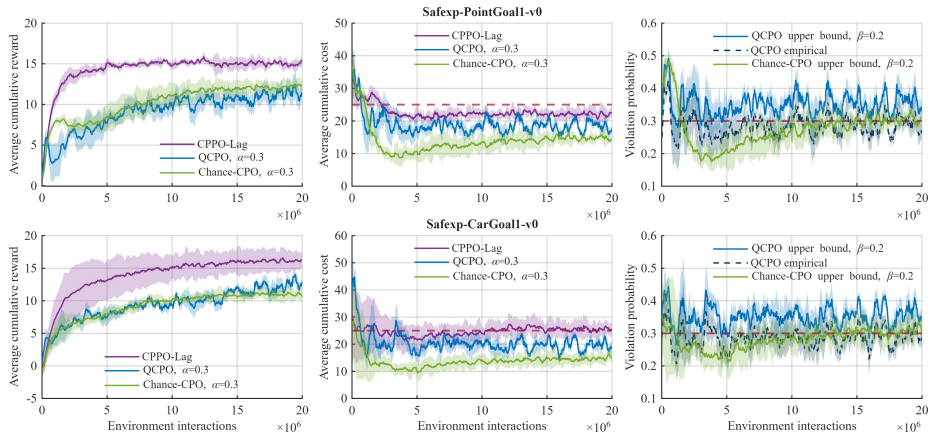


Figure 3: Performance of CPPO-Lag Ray et al. (2019), QCPO Jung et al. (2022), and the proposed Chance-CPO in two environments.

upper confidence bound on the violation probability at confidence level $1 - \beta = 0.8$. As shown in the right column, the proposed method ensures that the Clopper–Pearson upper bound of the violation probability to converge near α, with only small fluctuations across different α. The middle column in Figure 2 presents the corresponding average cumulative cost. As the prescribed violation probability threshold becomes smaller, the trained policy achieves a lower average cumulative cost, since a smaller α imposes a stricter safety requirement. Simultaneously, as shown in the left column, the average cumulative reward decreases under smaller violation probability thresholds, because the policy becomes more conservative in order to improve safety.

Figure 3 presents the performance of the three considered approaches. For the proposed Chance-CPO and QCPO Jung et al. (2022), we report the average cumulative reward, average cumulative cost, and Clopper–Pearson upper confidence bound of the violation probability with $1 - \beta = 0.8$ under $\alpha = 0.3$. For QCPO Jung et al. (2022), we also report the empirical violation probability. For CPPO-Lag Ray et al. (2019), only the average cumulative reward and cost are shown since it considers constraints only in expectation. As shown in Figure 3, the proposed Chance-CPO drives the violation probability converge to and remain within a narrow region of the threshold, as measured by the the Clopper–Pearson upper confidence bound. The empirical violation probabilities of QCPO are generally close to the threshold, while its upper bound under the same confidence level used by Chance-CPO exceeds the given α. Additionally, the proposed method achieves lower average cumulative costs than the two baseline methods, demonstrating its superior safety performance. Moreover, the average cumulative reward of Chance-CPO remains comparable to that of QCPO Jung et al. (2022). Simulations in Appendix H further demonstrate that Chance-CPO achieves lower average cumulative cost and violation probability than QCPO. Moreover, in high-dimensional environments such as DoggoGoal1-v0 and DoggoButton1-v0, Chance-CPO attains higher average rewards.

7 Conclusion

We proposed a confidence-aware chance-constrained policy optimization (Chance-CPO) method that maintains the violation probability around a reasonably selected, user-specified confidence level by connecting chance constraints with expectation-based constraints. An upper bound on the violation probability was provided in terms of the scaled expected cumulative cost through an appropriately selected scaling parameter. The proposed method also accounts for statistical errors due to finite trajectory samples and changes in the violation probability caused by multiple policy gradient updates. We evaluated the performance of the proposed Chance-CPO on simulated robotic tasks and compared it with constrained PPO based on Lagrangian relaxation and QCPO. Simulation results showed that Chance-CPO can achieve lower expected cumulative cost than baseline methods on the simulated robotic tasks. In the considered experiment settings, Chance-CPO also yields a lower probability of unsafe events than QCPO, suggesting its potential for improved control of safety risks.

AI USE STATEMENT

In this work, we used generative AI tools for code debugging and language polishing to improve readability. We did not use generative AI tools for literature search or summarization, the generation of conceptual frameworks or research hypotheses, the formulation of mathematical claims, methodology development, experimental design, or dataset generation. Other tasks with required disclosure are not applicable to this work. The authors reviewed and revised all AI-assisted code and text to ensure accuracy and clarity of meaning and take full responsibility for the final content of this work, including all claims, analyses, results, and conclusions.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide the pseudocode of the proposed algorithm in Appendix D, and release the full implementation in an anonymous GitHub repository at `https://github.com/Iris45016/Chance-CPO.git`. The proofs of Propositions 1-2 and Theorem 2, together with explanations of other algorithm design details, are provided in Appendices A-C.

REFERENCES

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International Conference on Machine Learning*, pp. 22–31. Pmlr, 2017.

Eitan Altman. *Constrained Markov decision processes*. Routledge, 2021.

Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *International Conference on Machine Learning*, pp. 449–458. Pmlr, 2017.

Weiqin Chen, Dharmashankar Subramanian, and Santiago Paternain. Probabilistic constraint for safety-critical reinforcement learning. *IEEE Transactions on Automatic Control*, 69(10):6789–6804, 2024.

Yinlam Chow, Mohammad Ghavamzadeh, Lucas Janson, and Marco Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *Journal of Machine Learning Research*, 18(167): 1–51, 2018a.

Yinlam Chow, Ofir Nachum, Edgar Duenez-Guzman, and Mohammad Ghavamzadeh. A Lyapunov-based approach to safe reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018b.

Charles J Clopper and Egon S Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934.

Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International Conference on Machine Learning*, pp. 1096–1105. PMLR, 2018.

Dongsheng Ding, Xiaohan Wei, Zhuoran Yang, Zhaoran Wang, and Mihailo Jovanovic. Provably efficient safe exploration via primal-dual policy optimization. In *International Conference on Artificial Intelligence and Statistics*, pp. 3304–3312. PMLR, 2021.

Peter Geibel. Reinforcement learning for mdps with constraints. In *European Conference on Machine Learning*, pp. 646–653. Springer, 2006.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*, pp. 1861–1870. Pmlr, 2018.

Ashish K Jayant and Shalabh Bhatnagar. Model-based safe deep reinforcement learning via a constrained proximal policy optimization algorithm. *Advances in Neural Information Processing Systems*, 35:24432–24445, 2022.

Norman L Johnson, Samuel Kotz, and Narayanaswamy Balakrishnan. *Discrete multivariate distributions*. John Wiley & Sons, 1997.

Whiyoung Jung, Myungsik Cho, Jongeui Park, and Youngchul Sung. Quantile constrained reinforcement learning: A reinforcement learning framework constraining outage probability. *Advances in Neural Information Processing Systems*, 35:6437–6449, 2022.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *International Conference on Machine Learning*, pp. 267–274, 2002.

Dohyeong Kim and Songhwai Oh. Trc: Trust region conditional value at risk for safe reinforcement learning. *IEEE Robotics and Automation Letters*, 7(2):2621–2628, 2022.

Yongshuai Liu, Jiaxin Ding, and Xin Liu. IPO: Interior-point policy optimization under constraints. *Conference on Artificial Intelligence*, 34(04):4940–4947, 2020.

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. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning*, pp. 1928–1937. PmLR, 2016.

Arkadi Nemirovski and Alexander Shapiro. Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, 17(4):969–996, 2007.

Masahiro Ono and Brian C Williams. An efficient motion planning algorithm for stochastic dynamic systems with constraints on probability of failure. In *AAAI*, pp. 1376–1382, 2008.

Santiago Paternain, Juan Andrés Bazerque, Austin Small, and Alejandro Ribeiro. Stochastic policy gradient ascent in reproducing kernel hilbert spaces. *IEEE Transactions on Automatic Control*, 66(8):3429–3444, 2020.

Santiago Paternain, Miguel Calvo-Fullana, Luiz FO Chamon, and Alejandro Ribeiro. Safe policies for reinforcement learning via primal-dual methods. *IEEE Transactions on Automatic Control*, 68(3):1321–1336, 2022.

Billingsley Patrick. *Probability and measure*. John Wiley, 1995.

Ilija Radosavovic, Tete Xiao, Bike Zhang, Trevor Darrell, Jitendra Malik, and Koushil Sreenath. Real-world humanoid locomotion with reinforcement learning. *Science Robotics*, 9(89):eadi9579, 2024.

Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 7(1):2, 2019.

R Tyrrell Rockafellar and Stanislav Uryasev. Conditional value-at-risk for general loss distributions. *Journal of Banking & Finance*, 26(7):1443–1471, 2002.

Ahmad EL Sallab, Mohammed Abdou, Etienne Perot, and Senthil Yogamani. Deep reinforcement learning framework for autonomous driving. *arXiv preprint arXiv:1704.02532*, 2017.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International Conference on Machine Learning*, pp. 1889–1897. PMLR, 2015a.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015b.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Shahil Shaik, Jonathon M Smereka, and Yue Wang. Generalized advantage estimation for distributional policy gradients. In *American Control Conference*, pp. 3073–3078. IEEE, 2025.

Ivan Sorokin, Alexey Seleznev, Mikhail Pavlov, Aleksandr Fedorov, and Anastasiia Ignateva. Deep attention recurrent q-network. *arXiv preprint arXiv:1512.01693*, 2015.

Måns Thulin. The cost of using exact confidence intervals for a binomial proportion. *Electronic Journal of Statistics*, 8(1):817–840, 2014.

Derek Yang, Li Zhao, Zichuan Lin, Tao Qin, Jiang Bian, and Tie-Yan Liu. Fully parameterized quantile function for distributional reinforcement learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Tsung-Yen Yang, Justinian Rosca, Karthik Narasimhan, and Peter J. Ramadge. Projection-based constrained policy optimization. In *International Conference on Learning Representations*, 2020.

A Proofs of Propositions and Theorem

A.1 Proof of Proposition 1

Let the random variable X_i indicate whether the ith trajectory in $\mathcal{D}_k$ is unsafe, that is,

$$X_i = \mathbb{1}(J_c^i(\pi_{\theta_{k-1}}) \geq d) \tag{16}$$

where $\mathbb{1}(\cdot)$ is the indicator function, i.e., $\mathbb{1}(J_c^i(\pi_{\theta_{k-1}}) \geq d) = 1$ if $J_c^i(\pi_{\theta_{k-1}}) \geq d$ and 0 otherwise. Since all trajectories in $\mathcal{D}_k$ are independently generated by applying $\pi_{\theta_{k-1}}$ to the same system, the random variables $X_1, \ldots, X_N$ are independent and identically distributed (i.i.d.) Bernoulli random variables with violation probability $p_{\pi_{\theta_{k-1}}} := \mathcal{P}(J_c(\pi_{\theta_{k-1}}) \geq d)$ as follows:

$$X_i \sim \text{Bernoulli}(p_{\pi_{\theta_{k-1}}}), \quad i = 1, \ldots, N \tag{17}$$

Therefore, according to the definition of the binomial distribution, it holds that:

$$S_k = \sum_{i=1}^{N} X_i \sim \text{Binomial}(N, p_{\pi_{\theta_{k-1}}}) \tag{18}$$

For the binomial distribution in (18), the Clopper-Pearson exact method Clopper & Pearson (1934); Thulin (2014) gives an upper confidence bound $\bar{p}_{\pi_{\theta_{k-1}}} := \text{BetaInv}(1-\beta; n_k+1, N-n_k)$ that guarantees

$$\mathcal{P}(p_{\pi_{\theta_{k-1}}} \leq \bar{p}_{\pi_{\theta_{k-1}}}) \geq 1-\beta \tag{19}$$

When $\bar{p}_{\pi_{\theta_{k-1}}} \leq \frac{\eta_c(\pi_{\theta_{k-1}})}{d}$, based on (7), $\delta_k = \frac{\eta_c(\pi_{\theta_{k-1}})}{\text{BetaInv}(1-\beta; n_k+1, N-n_k)}$. In this case, $\eta_c(\pi_{\theta_{k-1}}) \leq \delta_k \alpha$ gives that

$$\eta_c(\pi_{\theta_{k-1}}) \leq \delta_k \alpha = \frac{\eta_c(\pi_{\theta_{k-1}})}{\bar{p}_{\pi_{\theta_{k-1}}}} \alpha \tag{20}$$

Since $c(s_t, a_t) \geq 0$, it holds that $\eta_c(\pi_{\theta_{k-1}}) \geq 0$. When $\eta_c(\pi_{\theta_{k-1}}) > 0$, (20) yields

$$\bar{p}_{\pi_{\theta_{k-1}}} \leq \alpha \tag{21}$$

Based on (19) and (21), we have

$$\mathcal{P}(p_{\pi_{\theta_{k-1}}} \leq \alpha) \geq \mathcal{P}(p_{\pi_{\theta_{k-1}}} \leq \bar{p}_{\pi_{\theta_{k-1}}}) \geq 1-\beta \tag{22}$$

Recalling that $p_{\pi_{\theta_{k-1}}} := \mathcal{P}(J_c(\pi_{\theta_{k-1}}) \geq d)$, we obtain from (22) that

$$\mathcal{P}(\mathcal{P}(J_c(\pi_{\theta_{k-1}}) \geq d) \leq \alpha) \geq 1-\beta \tag{23}$$

When $\eta_c(\pi_{\theta_{k-1}}) = 0$, we have $c(s_t, a_t) = 0$, $t = 0, \ldots, \infty$, since $c(s_t, a_t) \geq 0$. Therefore, we have $\mathcal{P}\big(J_c(\pi_{\theta_{k-1}}) \geq d\big) = 0$, and $\mathcal{P}\big(\mathcal{P}\big(J_c(\pi_{\theta_{k-1}}) \geq d\big) \leq \alpha\big) \geq 1-\beta$ holds trivially.

When $\bar{p}_{\pi_{\theta_{k-1}}} > \frac{\eta_c(\pi_{\theta_{k-1}})}{d}$, the Clopper–Pearson bound computed from the sample statistics is more conservative than the upper bound given by Markov's inequality in (4). In this case, we adopt the Markov-inequality bound instead. Specifically, according to the definition of δ_k in (7), we have $\delta_k = d$. $\pi_{\theta_{k-1}}$ satisfying $\eta_c(\pi_{\theta_{k-1}}) \leq d\alpha$ also satisfies $\mathcal{P}\big(J_c(\pi_{\theta_{k-1}}) \geq d\big) \leq \frac{\eta_c(\pi_{\theta_{k-1}})}{d} \leq \alpha$ based on Markov's inequality (4). Thus, the chance constraint is satisfied deterministically, and hence with probability at least $1-\beta$.

A.2 Proof of Proposition 2

Since $w_0 = 1$ and $w_{t+1} = \frac{w_t}{\gamma}$ in (8), we have $w_t = \gamma^{-t}$. Hence, for all $t \in \mathbb{N}$, it holds that $\gamma^t w_t = \gamma^t \gamma^{-t} = 1$.

Now suppose that the auxiliary state y_t is less than zero for the first time at $\tau := \inf\{t \in \mathbb{N} \mid y_{t+1} \leq 0\}$. Then $y_\tau > 0$ and $y_{\tau+1} \leq 0$, and therefore $\mathbb{1}(y_\tau > 0 \wedge y_{\tau+1} \leq 0) = 1$. Moreover, since

$c(s_t, a_t, s_{t+1}) \geq 0$ and $\gamma > 0$, once $y_t \leq 0$, the auxiliary dynamics imply $y_{t+1} = \frac{y_t - c(s_t, a_t, s_{t+1})}{\gamma} \leq 0$. Therefore, the indicator $\mathbb{1}(y_t > 0 \ \wedge \ y_{t+1} \leq 0)$ in (10) can be equal to one at most once.

Consequently, if there exists such a $\tau \in \mathbb{N}$, then $J_{\bar{c}}(\kappa_x) = \gamma^\tau w_\tau = \gamma^\tau \gamma^{-\tau} = 1$. Otherwise, all indicators in (10) are zero and $J_{\bar{c}}(\kappa_x) = 0$. Moreover, from the recursion of y_t in (8), we obtain $y_{t+1} = \gamma^{-(t+1)} \left(d - \sum_{k=0}^{t} \gamma^k c(s_k, a_k, s_{k+1}) \right)$. Thus, $y_{t+1} \leq 0$ is equivalent to $\sum_{k=0}^{t} \gamma^k c(s_k, a_k, s_{k+1}) \geq d$.

Therefore, $J_{\bar{c}}(\kappa_x)$ indicates whether the discounted cost along the trajectory reaches the threshold d, that is,

$$J_{\bar{c}}(\kappa_x) = \mathbb{1}(J_c(\kappa) \geq d) \tag{24}$$

Taking expectation over $\kappa_x \sim d_x^{\pi_\theta}$, we obtain

$$\eta_{\bar{c}}(\pi_\theta) = \mathbb{E}_{\kappa_x \sim d_x^{\pi_\theta}} \left[J_{\bar{c}}(\kappa_x) \right] = \mathbb{E}_{\kappa_x \sim d_x^{\pi_\theta}} \left[\mathbb{1}(J_c(\kappa_x) \geq d) \right] \tag{25}$$

where $J_c(\kappa_x)$ denotes the original discounted cumulative cost evaluated along the state-action components of the augmented trajectory κ_x. Moreover, notice that the marginal distribution of $d_x^{\pi_\theta}$ over the original state-action trajectory coincides with d^{π_θ}, since the auxiliary states y_t and w_t are generated deterministically from the original trajectory through (8). Therefore, we have

$$\mathbb{E}_{\kappa_x \sim d_x^{\pi_\theta}} \left[\mathbb{1}(J_c(\kappa_x) \geq d) \right] = \mathbb{E}_{\kappa \sim d^{\pi_\theta}} \left[\mathbb{1}(J_c(\kappa) \geq d) \right] \tag{26}$$

Consequently, based on (25) and (26), it holds that

$$\eta_{\bar{c}}(\pi_\theta) = \mathbb{E}_{\kappa \sim d^{\pi_\theta}} \left[\mathbb{1}(J_c(\kappa) \geq d) \right] = \mathcal{P}\left(J_c(\pi_\theta) \geq d \right) \tag{27}$$

which completes the proof.

Remark 1 *Proposition 2 establishes the relationship between the expected discounted cumulative auxiliary cost under the infinite-horizon setting, i.e., $\eta_{\bar{c}}(\pi_\theta)$ in equation 27, and the corresponding violation probability $\mathcal{P}\left(J_c(\pi_\theta) \geq d \right)$. The analysis in Proposition 2 is built upon the infinite-horizon condition. Specifically, the auxiliary state variables y_t and w_t are defined to evolve over an infinite horizon, i.e., $t \in \mathbb{N}$. However, infinite-horizon sampling is not feasible in practical implementation. Therefore, in the practical algorithm developed in Section 5, following the policy optimization framework of trust region policy optimization (TRPO) Schulman et al. (2015a) and proximal policy optimization (PPO) Schulman et al. (2017), we use the advantage functions associated with the auxiliary cost instead. The advantages are estimated from finite-length sampled trajectories using generalized advantage estimation (GAE) Mnih et al. (2016), as introduced in Appendix C.1.*

A.3 Proof of Theorem 2

Based on (19) in Proposition 1, it holds that $\mathcal{P}(\mathcal{P}(J_c(\pi_{\theta_{k-1}}) \geq d) \leq \mathrm{BetaInv}(1-\beta; n_k + 1, N - n_k)) \geq 1 - \beta$. Combining this confidence bound with Theorem 1 and Proposition 2, the denominator of the first term in (11) constitutes an upper bound of $\mathcal{P}(J_c(\pi_\theta) \geq d)$ with probability at least $1 - \beta$. Meanwhile, based on Theorem 1, the numerator of the first term in (11) is a lower bound on $\eta_c(\pi_\theta)$. Therefore, with probability at least $1 - \beta$,

$$\frac{\eta_c(\pi_{\theta_{k-1}}) + L_c^{\pi_{\theta_{k-1}}}(\pi_\theta) - C_c \mathbb{E}_{s \sim \rho^{\pi_{\theta_{k-1}}}} \left[D_{\mathrm{TV}}(\pi_{\theta_{k-1}} \| \pi_\theta)[s] \right]}{\mathrm{BetaInv}(1-\beta; n_k + 1, N - n_k) + L_{\bar{c}}^{\pi_{\theta_{k-1}}}(\pi_\theta) + C_{\bar{c}} \mathbb{E}_{s \sim \rho^{\pi_{\theta_{k-1}}}} \left[D_{\mathrm{TV}}(\pi_{\theta_{k-1}} \| \pi_\theta)[s] \right]} \leq \frac{\eta_c(\pi_\theta)}{\mathcal{P}(J_c(\pi_\theta) \geq d)}$$

Moreover, Markov's inequality in (4) gives $\mathcal{P}\left(J_c(\pi_\theta) \geq d \right) \leq \frac{\eta_c(\pi_\theta)}{d}$. If the violation probability is positive, this is equivalent to $d \leq \frac{\eta_c(\pi_\theta)}{\mathcal{P}(J_c(\pi_\theta) \geq d)}$. If the violation probability is zero, the chance constraint holds directly.

Thus, both terms inside the maximum in (11) are no larger than $\frac{\eta_c(\pi_\theta)}{\mathcal{P}(J_c(\pi_\theta) \geq d)}$ with probability at least $1 - \beta$, that is,

$$\mathcal{P}\big(\delta(\theta) \leq \frac{\eta_c(\pi_\theta)}{\mathcal{P}(J_c(\pi_\theta) \geq d)} \big) \geq 1 - \beta \tag{28}$$

Consequently, if π_θ satisfies the constraint in (5b), then based on (28), we have

$$\mathcal{P}\big(\eta_c(\pi_\theta) \leq \frac{\eta_c(\pi_\theta)}{\mathcal{P}(J_c(\pi_\theta) \geq d)}\alpha\big) \geq 1-\beta \tag{29}$$

which is equivalent to $\mathcal{P}\big(\mathcal{P}(J_c(\pi_\theta) \geq d) \leq \alpha\big) \geq 1-\beta$. This implies that π_θ is also feasible for (3).

B PHYSICAL INTERPRETATION OF THE AUXILIARY STATE DESIGN

From a distributional perspective of MDPs, the discounted cumulative cost of trajectories starting from a state-action pair (s,a), denoted by $Z_c^{\pi_\theta}(s,a)$, is a random variable defined as follows:

$$Z_c^{\pi_\theta}(s,a) = [\sum_{t=0}^{\infty} \gamma^t c(s_t,a_t,s_{t+1}) \mid s_0 = s, a_0 = a] \tag{30}$$

where $s_{t+1} \sim \mathbb{P}(\cdot \mid s_t, a_t)$ for $t \geq 0$, and $a_t \sim \pi_\theta(\cdot \mid s_t)$ for $t \geq 1$. The corresponding value distribution satisfies the distributional Bellman equation Bellemare et al. (2017); Shaik et al. (2025); Dabney et al. (2018):

$$Z_c^{\pi_\theta}(s_t,a_t) \overset{D}{=} c(s_t,a_t,s_{t+1}) + \gamma Z_c^{\pi_\theta}(s_{t+1},a_{t+1}) \tag{31}$$

where the distribution equation $M \overset{D}{=} N$ indicates that the random variable M is distributed under the same law as N.

The distributional Bellman equation allows the violation probability to be propagated along a trajectory. In particular,

$$\mathcal{P}(Z_c^{\pi_\theta}(s_t,a_t) \geq d) = \mathcal{P}(c(s_t,a_t,s_{t+1}) + \gamma Z_c^{\pi_\theta}(s_{t+1},a_{t+1}) \geq d) \tag{32a}$$

$$= \mathcal{P}\left(Z_c^{\pi_\theta}(s_{t+1},a_{t+1}) \geq \frac{d - c(s_t,a_t,s_{t+1})}{\gamma}\right) \tag{32b}$$

This observation suggests introducing a recursively updated cost margin. Let $y_t := d$ and define $y_{t+1} := \frac{y_t - c(s_t,a_t,s_{t+1})}{\gamma}$. Applying the distributional Bellman equation recursively yields:

$$\mathcal{P}(Z_c^{\pi_\theta}(s_t,a_t) \geq d) = \mathcal{P}\left(c(s_t,a_t,s_{t+1}) + \gamma\big(c(s_{t+1},a_{t+1},s_{t+2}) + \gamma Z_c^{\pi_\theta}(s_{t+2},a_{t+2})\big) \geq d\right) \tag{33a}$$

$$= \mathcal{P}\left(Z_c^{\pi_\theta}(s_{t+2},a_{t+2}) \geq y_{t+2}\right) \tag{33b}$$

where $y_{t+2} := \frac{y_{t+1} - c(s_{t+1},a_{t+1},s_{t+2})}{\gamma}$

By induction, this recursion gives that, for any $\tau \geq 0$,

$$\mathcal{P}\left(Z_c^{\pi_\theta}(s_t,a_t) \geq d\right) = \mathcal{P}\left(Z_c^{\pi_\theta}(s_{t+\tau},a_{t+\tau}) \geq y_{t+\tau}\right) \tag{34}$$

Therefore, y_t can be viewed as a cost margin propagated along the trajectory.

C PRACTICAL DESIGN

C.1 ADVANTAGE ESTIMATION

Let κ_i denote the ith trajectory in $\mathcal{D}_k$ with length T_i, and let H denote the total number of time steps in $\mathcal{D}_k$, i.e., $H = \sum_{\kappa_i \in \mathcal{D}_k} T_i$. We train three neural networks, $F_{\psi_r}(s)$, $F_{\psi_c}(s)$, and $F_{\psi_{\bar{c}}}(x)$, to approximate the reward value function, the cost value function, and the auxiliary cost value function, respectively. Their parameters ψ_r, ψ_c, and $\psi_{\bar{c}}$ are optimized by minimizing the mean squared errors to the corresponding empirical reward or cost-to-go as follows Jayant & Bhatnagar (2022):

$$\mathcal{L}_r(\psi_r) = \frac{1}{H} \sum_{\kappa_i \in \mathcal{D}_k} \sum_{t=0}^{T_i-1} \left(F_{\psi_r}(s_{i,t}) - \hat{R}_{i,t} \right)^2, \tag{35a}$$

$$\mathcal{L}_c(\psi_c) = \frac{1}{H} \sum_{\kappa_i \in \mathcal{D}_k} \sum_{t=0}^{T_i-1} \left(F_{\psi_c}(s_{i,t}) - \hat{C}_{i,t} \right)^2, \tag{35b}$$

$$\mathcal{L}_{\bar{c}}(\psi_{\bar{c}}) = \frac{1}{H} \sum_{\kappa_i \in \mathcal{D}_k} \sum_{t=0}^{T_i-1} \left(F_{\psi_{\bar{c}}}(x_{i,t}) - \hat{\bar{C}}_{i,t} \right)^2, \tag{35c}$$

where $\hat{R}_{i,t} := \sum_{t'=t}^{T_i-1} \gamma^{t'-t} r_{i,t'}$, $\hat{C}_{i,t} := \sum_{t'=t}^{T_i-1} \gamma^{t'-t} c_{i,t'}$, and $\hat{\bar{C}}_{i,t} := \sum_{t'=t}^{T_i-1} \gamma^{t'-t} \bar{c}_{i,t'}$ denote the empirical reward-to-go, cost-to-go, and auxiliary cost-to-go targets, respectively; the subscript (i,t) indicates that the corresponding variable is evaluated at time step t along trajectory κ_i.

For a trajectory κ_i of length T_i, define $\zeta_t^r = r_t + \gamma F_{\psi_r}(s_{t+1}) - F_{\psi_r}(s_t)$, $\zeta_t^c = c_t + \gamma F_{\psi_c}(s_{t+1}) - F_{\psi_c}(s_t)$, and $\zeta_t^{\bar{c}} = \bar{c}_t + \gamma F_{\psi_{\bar{c}}}(x_{t+1}) - F_{\psi_{\bar{c}}}(x_t)$. The GAE-based advantage estimates are then given by Mnih et al. (2016); Schulman et al. (2015b; 2017):

$$\hat{A}_t^r = \zeta_t^r + \gamma\mu\zeta_{t+1}^r + \cdots + (\gamma\mu)^{T_i-t-1}\zeta_{T_i-1}^r, \tag{36a}$$

$$\hat{A}_t^c = \zeta_t^c + \gamma\mu\zeta_{t+1}^c + \cdots + (\gamma\mu)^{T_i-t-1}\zeta_{T_i-1}^c, \tag{36b}$$

$$\hat{A}_t^{\bar{c}} = \zeta_t^{\bar{c}} + \gamma\mu\zeta_{t+1}^{\bar{c}} + \cdots + (\gamma\mu)^{T_i-t-1}\zeta_{T_i-1}^{\bar{c}}. \tag{36c}$$

where μ is the GAE parameter that controls the bias–variance trade-off in the advantage estimation Schulman et al. (2015b).

C.2 UNBIASED ESTIMATE OF INFINITE-HORIZON CUMULATIVE COST

Consider the expectation of the finite-horizon cumulative cost with the horizon T independently drawn from a geometric distribution. It holds that:

$$\mathbb{E}_{\kappa \sim d^{\pi_{\theta_{k-1}}},\, T \sim \mathrm{Geom}(1-\gamma)} \left[\sum_{t=0}^{T} c(s_t, a_t, s_{t+1}) \right] \tag{37a}$$

$$= \mathbb{E}_{\kappa \sim d^{\pi_{\theta_{k-1}}},\, T \sim \mathrm{Geom}(1-\gamma)} \left[\sum_{t=0}^{\infty} \mathbb{1}(t \le T) c(s_t, a_t, s_{t+1}) \right] \tag{37b}$$

Since the cost function satisfies $c(s_t, a_t, s_{t+1}) \ge 0$, based on the monotone convergence theorem, we have that:

$$\mathbb{E}_{\kappa \sim d^{\pi_{\theta_{k-1}}},\, T \sim \mathrm{Geom}(1-\gamma)} \left[\sum_{t=0}^{\infty} \mathbb{1}(t \le T) c(s_t, a_t, s_{t+1}) \right] \tag{38a}$$

$$= \sum_{t=0}^{\infty} \mathbb{E}_{T \sim \mathrm{Geom}(1-\gamma)}[\mathbb{1}(t \le T)]\, \mathbb{E}_{\kappa \sim d^{\pi_{\theta_{k-1}}}}[c(s_t, a_t, s_{t+1})] \tag{38b}$$

Since T is drawn independently from a geometric distribution, i.e., $T \sim \mathrm{Geom}(1-\gamma)$ and $\mathcal{P}(T = t) = (1-\gamma)\gamma^t$, it follows that:

$$\mathbb{E}_{T \sim \mathrm{Geom}(1-\gamma)}[\mathbb{1}(t \le T)] = \gamma^t \tag{39}$$

Based on (38) and (39), we have that:

$$\mathbb{E}_{\kappa\sim d^{\pi_{\theta_{k-1}}},\, T\sim\mathrm{Geom}(1-\gamma)}\left[\sum_{t=0}^{\infty}\mathbb{1}(t\leq T)c(s_t,a_t,s_{t+1})\right] \tag{40a}$$

$$=\sum_{t=0}^{\infty}\gamma^t\mathbb{E}_{\kappa\sim d^{\pi_{\theta_{k-1}}}}\left[c(s_t,a_t,s_{t+1})\right] \tag{40b}$$

This implies that the expectation of the finite-horizon cumulative cost without discounting provides an unbiased estimate of $\eta_c(\pi_{\theta_{k-1}})$, when the finite-horizon T satisfies $T\sim\mathrm{Geom}(1-\gamma)$. Specifically, by rewriting (40), we obtain the following conclusion:

$$\mathbb{E}_{\kappa\sim d^{\pi_{\theta_{k-1}}},\, T\sim\mathrm{Geom}(1-\gamma)}\left[\sum_{t=0}^{T}c(s_t,a_t,s_{t+1})\right] \tag{41a}$$

$$=\mathbb{E}_{\kappa\sim d^{\pi_{\theta_{k-1}}}}\left[\sum_{t=0}^{\infty}\gamma^t c(s_t,a_t,s_{t+1})\right]=\eta_c(\pi_{\theta_{k-1}}) \tag{41b}$$

D TRAINING ALGORITHM

The training and interaction process of the proposed Chance-CPO is summarized in Algorithm 1.

Algorithm 1: Environment interaction and neural network training of the proposed Chance-CPO.

Input: Initialized neural network parameters θ_0,ψ_{r_0}, ψ_{c_0}, $\psi_{\bar{c}_0}$, initialized Lagrange multiplier $\lambda_0\geq 0$, cost threshold d, confidence level $1-\beta$; number of iterations H; number of gradient steps per iteration E

Output: Trained policy π_{θ^*}

1. **for** $k=1,\ldots,H$ **do**
   - 1.1 **while** $j\leq N$ **do**
     - Sample $T\sim\mathrm{Geom}(1-\gamma)$.
     - Collect a length-T trajectory using $\pi_{\theta_{k-1}}$, $F_{\psi_{r_{k-1}}}(s)$, $F_{\psi_{c_{k-1}}}(s)$, and $F_{\psi_{\bar{c}_{k-1}}}(x)$.
     - $j\leftarrow j+1$.

     Combine the collected trajectories to form the dataset $\mathcal{D}_k$.
   - 1.2 Compute δ_k in (7) based on $\mathcal{D}_k$.
   - 1.3 Set $\theta_{0,k}\leftarrow\theta_{k-1}$ and $\lambda_{0,k}\leftarrow\lambda_{k-1}$.
   - 1.4 **for** $i=1$ *to* E **do**
     - Update the policy parameter θ based on (15a).
     - Compute $\hat{\delta}(\theta_{i,k})$ based on (13).
     - Update the Lagrange multiplier λ based on (15b).

     Set $\theta_k\leftarrow\theta_{E,k}$ and $\lambda_k\leftarrow\lambda_{E,k}$
   - 1.5 **for** $i=1$ *to* E **do**
     - Update ψ_r, ψ_c, and $\psi_{\bar{c}}$ using loss functions in (35) with learning rates l_r, l_c, $l_{\bar{c}}$ as follows:
     - $\psi_r\leftarrow\psi_r-l_r\nabla_{\psi_r}\mathcal{L}_r$; $\psi_c\leftarrow\psi_c-l_c\nabla_{\psi_c}\mathcal{L}_c$; $\psi_{\bar{c}}\leftarrow\psi_{\bar{c}}-l_{\bar{c}}\nabla_{\psi_{\bar{c}}}\mathcal{L}_{\bar{c}}$

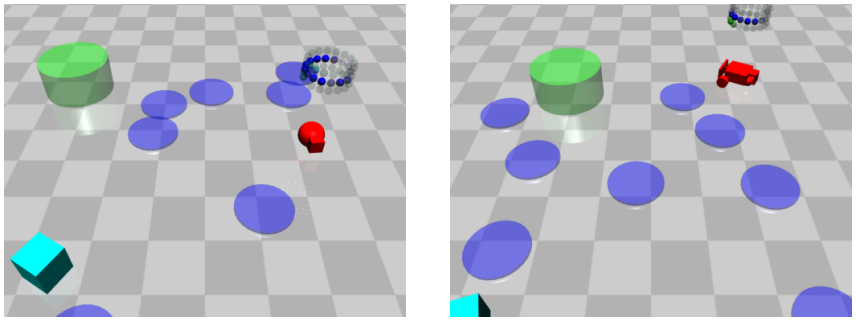
     Set $\psi_{r_k}\leftarrow\psi_r$, $\psi_{c_k}\leftarrow\psi_c$, and $\psi_{\bar{c}_k}\leftarrow\psi_{\bar{c}}$
2. Set $\theta^*\leftarrow\theta_H$.

return π_{θ^*}

E EXPERIMENT DETAILS

E.1 ENVIRONMENT

In Section 6, we consider two environments from Safety Gym [1], which are illustrated in Figure 4. These environments share the same navigation objective and safety constraint, but differ in robot dynamics, actuation mechanisms, and dimensions. The point robot is constrained to the two-dimensional plane and has two control inputs, one for turning and the other for moving forward or backward. The observation and action dimensions for the point robot are 60 and 2, respectively. The car robot has two independently driven parallel wheels and a free-rolling rear wheel, with a 72-dimensional observation space and a 2-dimensional action space. Turning and motion require coordinated control of both actuators.



(a) PointGoal1-v0 (b) CarGoal1-v0

Figure 4: Safety Gym goal environments used in Section 6. The green region denotes the goal, blue regions denote hazards, and the red object denotes the controlled agent.

In all environments, the robot aims to navigate to a sequence of goal positions while avoiding hazards. When the robot reaches the current goal region, a new goal is sampled, and the episode continues until the maximum episode length is reached. The reward function consists of both dense and sparse components. The dense reward encourages progress toward the current goal, while the sparse reward is given when the robot reaches the goal region. Safety is encoded through a cost signal returned by the environment. At each time step, the cost value is 1 if the robot enters a hazard region and 0 otherwise.

E.2 HYPERPARAMETERS

Hyperparameters for the training of the proposed Chance-CPO are presented in Table 1.

Table 1: Hyperparameters of Chance-CPO training.

Parameters	Values
Discount factor γ	0.999
GAE parameter μ	0.97
Confidence level $1-\beta$	0.8
Discounted cumulative cost threshold d	25
Clip ratio ξ	0.15
Learning rate of π_θ	1×10^{-4}
Learning rates of F_{ψ_r}, F_{ψ_c}, $F_{\psi_{\bar{c}}}$	3×10^{-4}
Learning rate of λ	8×10^{-5}

[1] https://github.com/openai/safety-gym

F EVALUATIONS

To assess the safety behavior of the learned policies, we repeatedly roll out each trained policy in its corresponding environment and record the cumulative cost of each evaluation episode. The Chance-CPO and QCPO policies are trained with the violation probability threshold $\alpha = 0.3$. Specifically, for each trained policy, we collect 200 evaluation trajectories. As described in Section 6, each method is trained with 5 seeds. Thus, for each method–environment pair, we collect 1,000 evaluation trajectories in total, with 200 trajectories for each of the five trained policies.

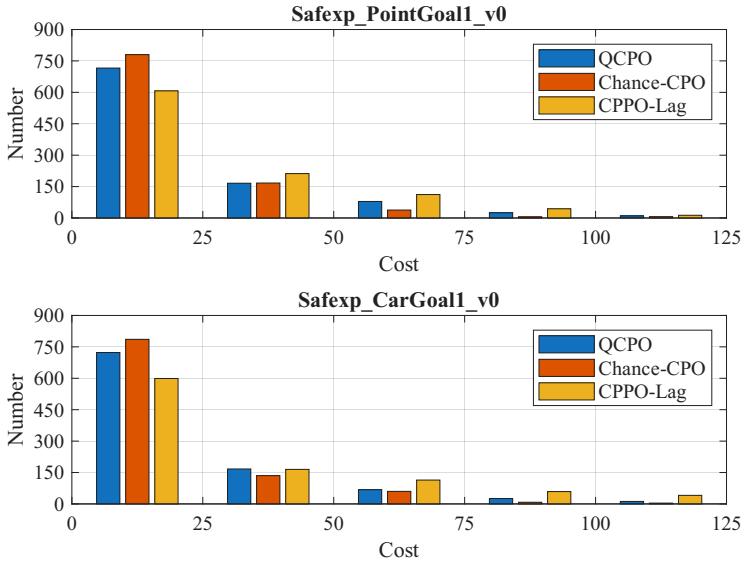


Figure 5: Cost distribution of trained policies for three methods under repeated evaluation rollouts.

Table 2: Violation probability of three methods during evaluation.

Environment	Method	Violation percentage
PointGoal1-v0	Chance-CPO	**21.6 %**
	QCPO Jung et al. (2022)	27.9 %
	CPPO-Lag Ray et al. (2019)	38.5 %
CarGoal1-v0	Chance-CPO	**20.6 %**
	QCPO Jung et al. (2022)	27.2 %
	CPPO-Lag Ray et al. (2019)	39.50 %

Figure 5 shows the distribution of cumulative costs over these evaluation trajectories. During training, the cumulative cost threshold is set to $d = 25$. As shown in Figure 5, the proposed Chance-CPO policy yields a larger number of evaluation trajectories below the threshold compared with the baseline methods. Moreover, the proposed Chance-CPO produces the fewest trajectories in the high-cost region, i.e., trajectories with cumulative cost greater than 75, indicating better safety performance and reduced tail risk.

Table 2 further summarizes the proportion of evaluation trajectories whose cumulative cost exceeds the prescribed threshold among the 1000 collected trajectories. Both Chance-CPO and QCPO keep the proportion of trajectories with cumulative cost above the threshold below the specified level $\alpha = 0.3$ (i.e., violation percentage below 30%), whereas CPPO-Lag fails to satisfy this requirement. Among all compared methods, the proposed Chance-CPO achieves the lowest violation percentage in both environments.

Additionally, the mean and standard deviation (std.) of the cumulative reward and cumulative cost are summarized in Tables 3 and 4, respectively. A higher mean reward indicates better task performance, whereas a lower mean cost indicates reduced average constraint violation. Meanwhile,

Table 3: Mean and standard deviation of cumulative reward for three methods.

Environment	Method	Reward mean	Reward std.
PointGoal1-v0	Chance-CPO	12.432	**8.0295**
	QCPO Jung et al. (2022)	11.012	8.0675
	CPPO-Lag Ray et al. (2019)	**15.236**	9.3845
CarGoal1-v0	Chance-CPO	11.557	**8.1804**
	QCPO Jung et al. (2022)	12.379	9.1112
	CPPO-Lag Ray et al. (2019)	**17.525**	10.818

Table 4: Mean and standard deviation of cumulative cost for three methods.

Environment	Method	Cost mean	Cost std.
PointGoal1-v0	Chance-CPO	**14.000**	**25.686**
	QCPO Jung et al. (2022)	18.028	24.640
	CPPO-Lag Ray et al. (2019)	25.539	34.395
CarGoal1-v0	Chance-CPO	**13.842**	23.919
	QCPO Jung et al. (2022)	17.382	**24.591**
	CPPO-Lag Ray et al. (2019)	28.797	35.613

a lower standard deviation indicates more stable performance across evaluation episodes and seeds. CPPO-Lag achieves the highest reward, but it also incurs the highest cumulative cost. Compared with QCPO, which also considers chance constraints, the proposed Chance-CPO obtains comparable reward: on PointGoal1-v0, Chance-CPO achieves a reward that is 12.9% higher than QCPO, while on CarGoal1-v0, its reward is 6.6% lower than QCPO. Chance-CPO also achieves the lowest mean cumulative cost among the three methods in both environments.

G Ablations

In this work, the proposed Chance-CPO includes two key designs to ensure the chance constraints can be satisfied reliably: (a) the Clopper–Pearson upper confidence bound for conservative violation-probability estimation, and (b) the scale-parameter correction terms computed based on an auxiliary MDP to account for policy-update-induced distributional drift. Here, we conduct ablation studies to assess the contribution of these two components.

Specifically, we compare the proposed Chance-CPO with two ablated variants:

- **Chance-CPO-A**: directly estimates the violation probability from empirical samples by replacing the Clopper–Pearson upper confidence bound in the denominator of (11) with $\hat{\mathcal{P}}\big(J_c(\pi_{\theta_{k-1}}) \geq d\big) = \frac{n_k+1}{N+1}$ introduced in Section 4.1.
- **Chance-CPO-B**: ignores the distributional drift induced by policy updates and directly uses δ_k in (7) throughout the kth iteration.

In the following, we evaluate these variants on PointGoal1-v0 and CarGoal1-v0, and compare their performance with the proposed Chance-CPO method.

G.1 Clopper–Pearson upper bound

We conduct ablation studies to evaluate the effect of the Clopper–Pearson upper confidence bound used in Chance-CPO. Specifically, we train Chance-CPO-A on PointGoal1-v0 and CarGoal1-v0 with $\alpha = 0.3$, where Chance-CPO-A directly uses the empirical estimate of the violation probability without accounting for the possible underestimation caused by a finite number of samples. For each environment, we train 5 independent policies with different seeds and compare Chance-CPO-A with Chance-CPO.

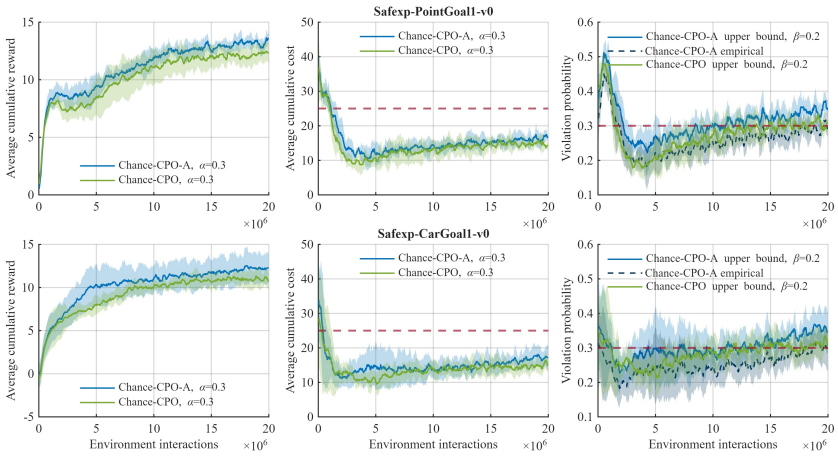


Figure 6: Performance comparison between Chance-CPO and Chance-CPO-A in two Safety Gym environments.

As shown in Figure 6, incorporating the Clopper–Pearson upper bound leads to a more conservative policy, since Chance-CPO utilizes an upper confidence bound of the violation probability rather than its direct empirical estimate. This design improves the reliability of chance-constraint satisfaction under finite-sample estimation uncertainty. In contrast, Chance-CPO-A directly uses the empirical violation probability estimate and therefore tends to achieve higher reward. Meanwhile, both the empirical violation probability of Chance-CPO-A and the Clopper–Pearson upper bound of Chance-CPO converge to the user-specified threshold α. This indicates that both methods are driven toward the same target violation level, while Chance-CPO enforces the chance constraint more conservatively.

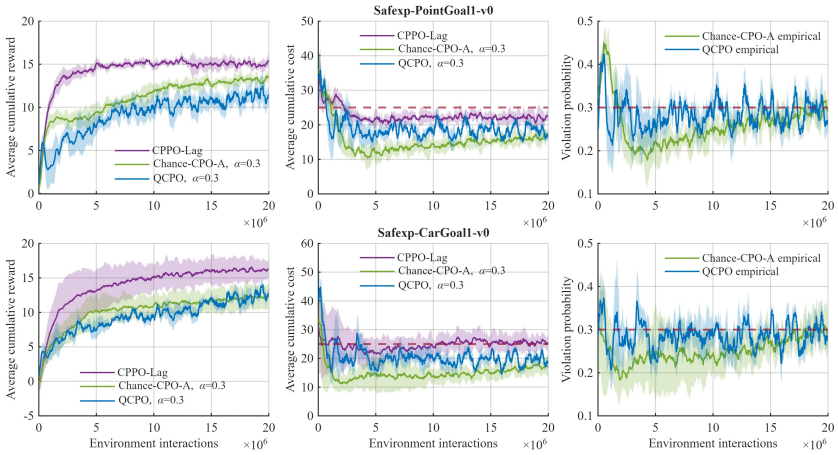


Figure 7: Performance comparison between CPPO-Lag Ray et al. (2019), QCPO Jung et al. (2022), and Chance-CPO-A in two environments.

Figure 7 compares Chance-CPO-A with two baseline methods. CPPO-Lag only constrains the expected cumulative cost and is the least conservative among the three methods. Both Chance-CPO-A and QCPO estimate the violation probability directly from a finite-sample dataset, and their violation probabilities converge to the user-specified threshold α. Under the same chance-constraint formulation, Chance-CPO-A delivers better overall performance than QCPO. Figure 7 shows that Chance-CPO-A achieves a higher average cumulative reward on PointGoal1-v0 and a comparable reward on CarGoal1-v0. Meanwhile, Chance-CPO-A achieves the lowest average cumulative cost

among the three compared methods, indicating a better trade-off between task performance and safety. Moreover, QCPO exhibits a less stable training process compared with the other two methods. One potential reason is that QCPO relies on learning a quantile function, whose estimation is generally more sensitive to finite-sample effects than expectation-based quantities, since quantiles depend on the local distributional information around the target probability level.

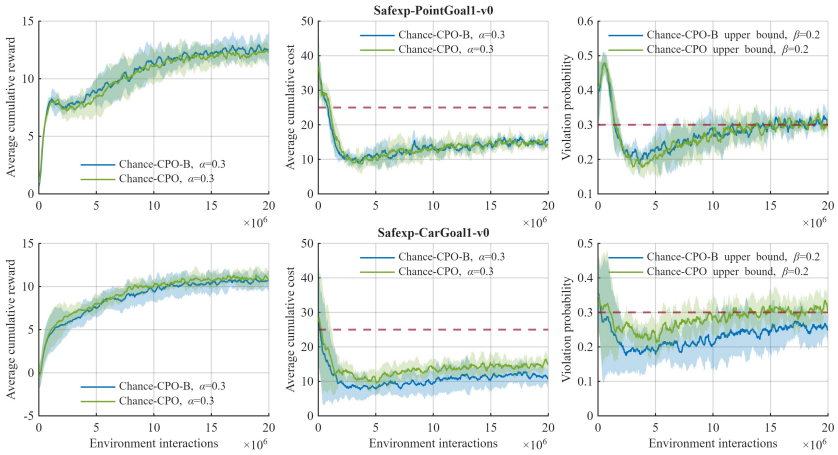


Figure 8: Performance of Chance-CPO and Chance-CPO-B in two environments.

G.2 Scaling parameter correction

In this work, as shown in (11), we correct both the numerator and denominator of the scaling parameter using advantage estimates to obtain a stable scaling parameter that defines a valid surrogate problem and enforces satisfaction of the chance constraints under the user-specified violation probability α. The surrogate problem induced by the corrected scaling parameter is expected to be neither too loose, which would fail to guarantee feasibility of the original problem, nor too conservative, which would pursue an unnecessarily low violation probability and sacrifice reward performance. In other words, an ideal scaling parameter should calibrate the resulting violation probability to the user-specified violation probability α.

To investigate the effect of scale-parameter correction, we evaluate Chance-CPO-B on PointGoal1-v0 and CarGoal1-v0 with $\alpha = 0.3$. Other parameters are the same as those used by Chance-CPO. For each environment, 5 Chance-CPO-B policies are trained independently with different seeds. Figure 8 reports the training performance of Chance-CPO-B and the full Chance-CPO method, where solid curves denote the mean over seeds and shaded regions indicate one standard deviation.

As shown in Figure 8, in both environments, the upper confidence bound of the violation probability under the proposed Chance-CPO converges to the user-specified level $\alpha = 0.3$. This suggests that the surrogate problem induced by Chance-CPO with the scale-parameter correction is well calibrated: it is neither overly conservative nor too loose. In contrast, without the correction, the fixed δ_k in (7) may either remain a reasonable estimate or become too loose or too conservative when the policy is updated. As shown at the bottom of Figure 8, Chance-CPO-B behaves conservatively in CarGoal1-v0. The upper confidence bound of the violation probability of Chance-CPO-B stays significantly below the user-specified level. Both the expected cumulative cost and reward of Chance-CPO-B are lower than those of Chance-CPO.

For each trained Chance-CPO-B policy, we collect 200 evaluation trajectories to evaluate its control performance. Figure 9 shows the distribution of cumulative costs for Chance-CPO and Chance-CPO-B. As shown in Figure 9, removing the scale-parameter correction may lead to either overly conservative policies or degraded safety performance, depending on the environment and training dynamics. The proposed scaling parameter correction is introduced to ensure that the formulated surrogate problem remains valid and appropriate across different scenarios.

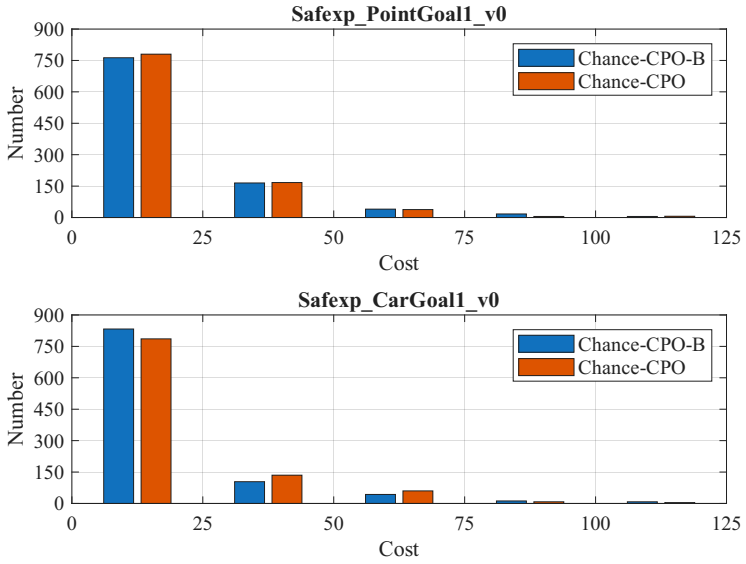


Figure 9: Cost distribution of trained policies for Chance-CPO and Chance-CPO-B.

H ADDITIONAL EXPERIMENTS

H.1 EXPERIMENTS ON DOGGOGOAL1-V0

We present simulation results for the goal-directed task using the Doggo robot in Safety Gym [2] (DoggoGoal1-v0). This environment has higher observation and action dimensionality compared to the two environments discussed in Section 6. The control objective is the same as that in the two environments introduced in Appendix E.1. This environment contains a simulated doggo robot, which is a quadruped with bilateral symmetry. Each of its four legs has three actuated joints, resulting in a higher-dimensional control problem. The observation and action dimensions of the doggo robot are 104 and 12, respectively.

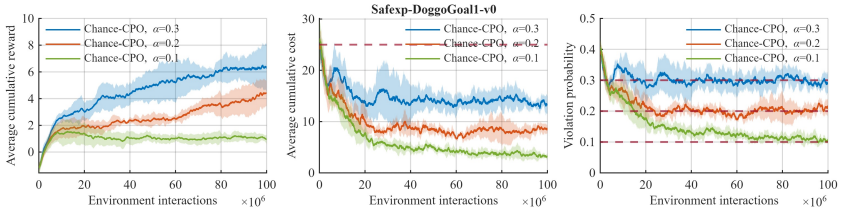


Figure 10: Performance of Chance-CPO on the DoggoGoal1-v0 environment with three violation probability thresholds $\alpha = 0.1, 0.2, 0.3$.

We evaluate the proposed Chance-CPO with three violation probability thresholds $\alpha \in \{0.1, 0.2, 0.3\}$. The discounted cumulative cost threshold is set to $d = 25$. We conduct 5 independent training experiments using 5 different seeds for each configuration. Figure 10 shows the training performance of the proposed Chance-CPO. As shown in this figure, the upper confidence bound of the violation probability converges to the given α under all configurations. Compared to the two low-dimensional environments presented in Section 6, simulations on the DoggoGoal1-v0 environment require more interaction steps to reach the desired safety level, particularly when the violation probability threshold α is small.

Additionally, we evaluate two baseline methods, CPPO-Lag Ray et al. (2019) and QCPO Jung et al. (2022), on the DoggoGoal1-v0 environment and compare them with the proposed approach. For

[2] https://github.com/openai/safety-gym

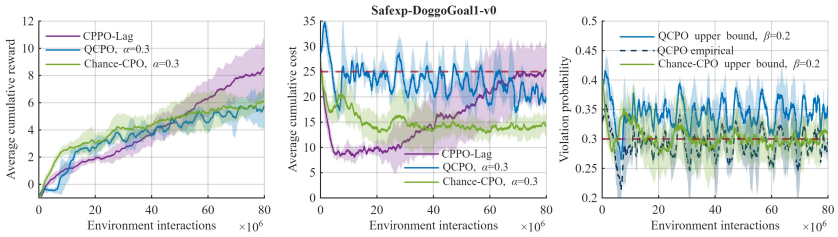


Figure 11: Performance of CPPO-Lag Ray et al. (2019), QCPO Jung et al. (2022), and Chance-CPO on the DoggoGoal1-v0 environment.

Table 5: Violation probability of three methods on DoggoGoal1-v0.

Method	Violation percentage	Constraint satisfaction
Chance-CPO	**18.8 %**	✓
QCPO Jung et al. (2022)	30.4 %	✗
CPPO-Lag Ray et al. (2019)	39.9%	✗

Table 6: Mean and standard deviation of cumulative reward and cumulative cost for three methods on DoggoGoal1-v0.

Metric	Method	Mean	Std.
Reward	Chance-CPO	6.5179	5.8845
	QCPO Jung et al. (2022)	5.4247	**3.9584**
	CPPO-Lag Ray et al. (2019)	**10.425**	7.5365
Cost	Chance-CPO	**11.883**	**20.243**
	QCPO Jung et al. (2022)	19.079	30.891
	CPPO-Lag Ray et al. (2019)	29.051	35.304

both baselines, the cost threshold is 25, and for QCPO the violation probability threshold is set to 0.3. Figure 11 presents the training performance comparison between the two baselines and the proposed Chance-CPO with $\alpha = 0.3$. In this high-dimensional control task, Chance-CPO achieves a higher average reward and a significantly lower average cost than QCPO. Tables 5 and 6 report the violation probability as well as the average cumulative reward and cost, respectively. In the DoggoGoal1-v0 environment, QCPO slightly violates the chance constraints as its violation percentage is higher than 30%, whereas Chance-CPO consistently satisfies the chance constraint. The average cumulative cost of Chance-CPO is 37.7% and 59.1% lower than that of QCPO and CPPO-Lag, respectively, while its average cumulative reward is 20.2% higher than QCPO.

H.2 Experiments on button tasks

To further assess the effectiveness of the proposed method, we conduct additional experiments on button-task environments in Safety Gym with three robot morphologies, including PointButton1-v0, CarButton1-v0, and DoggoButton1-v0. In the three environments, the robot aims to press a sequence of goal buttons. Multiple immobile buttons are distributed across the environment, and the robot presses the currently highlighted button to achieve the goal. After the agent presses the correct button, a new goal button is designated.

We evaluate the proposed Chance-CPO under three violation probability thresholds $\alpha \in \{0.1, 0.2, 0.3\}$. The discounted cumulative cost threshold is set to $d = 25$ on PointButton1-v0, $d = 50$ on CarButton1-v0, and $d = 100$ on DoggoButton1-v0. Figure 12 illustrates the training performance of Chance-CPO in the three button tasks. Consistent with the results on goal tasks shown in Figure 2 and 10, the upper confidence bound of the violation probability converges to the specified α across all configurations and environments. Moreover, we compare the proposed

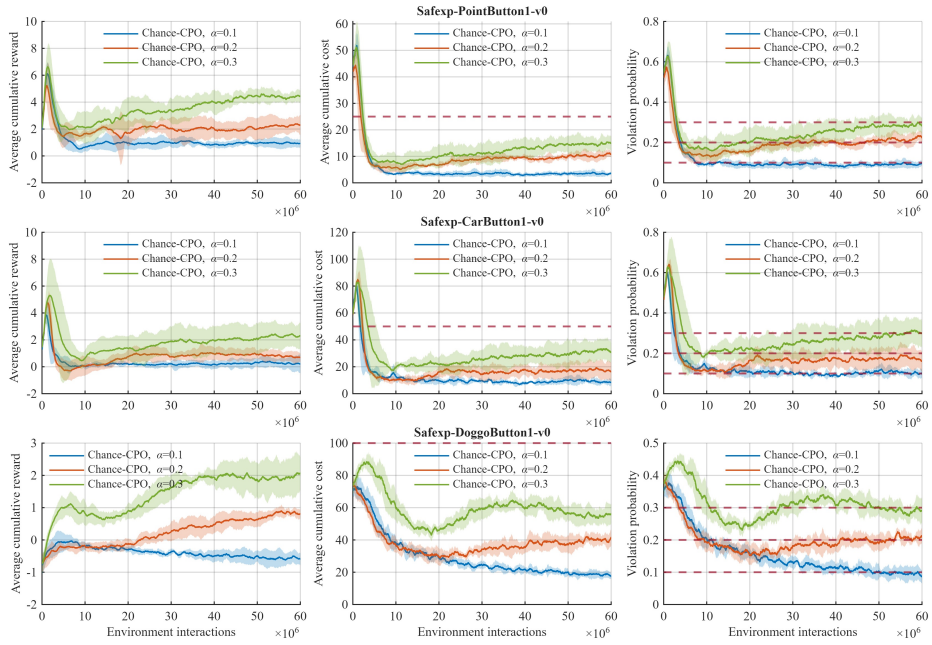


Figure 12: Performance of the proposed Chance-CPO on three button tasks with different violation probability thresholds $\alpha = 0.1, 0.2, 0.3$.

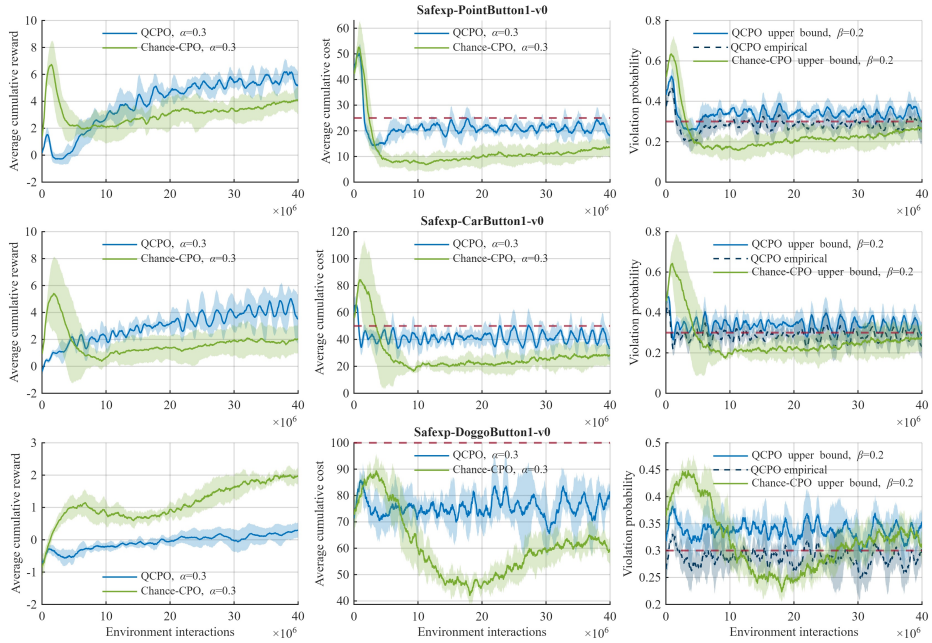


Figure 13: Performance of QCPO Jung et al. (2022) and the proposed Chance-CPO on button tasks.

Chance-CPO with QCPO Jung et al. (2022) on the three environments. As shown in Figure 13, the proposed Chance-CPO consistently achieves lower average cost and violation probability. In the two lower-dimensional environments (PointButton1-v0 and CarButton1-v0), QCPO attains higher rewards and costs, whereas in the higher-dimensional environment (DoggoButton1-v0), Chance-CPO yields lower costs while achieving higher rewards.