Constrained Variational Policy Optimization for Safe Reinforcement Learning

Zuxin Liu [1] **Zhepeng Cen** [1] **Vladislav Isenbaev** [2] **Wei Liu** [2] **Zhiwei Steven Wu** [1] **Bo Li** [3] **Ding Zhao** [1]

Abstract

Safe reinforcement learning (RL) aims to learn policies that satisfy certain constraints before deploying them to safety-critical applications. Previous primal-dual style approaches suffer from instability issues and lack optimality guarantees. This paper overcomes the issues from the perspective of probabilistic inference. We introduce a novel Expectation-Maximization approach to naturally incorporate constraints during the policy learning: 1) a provable optimal non-parametric variational distribution could be computed in closed form after a convex optimization (E-step); 2) the policy parameter is improved within the trust region based on the optimal variational distribution (M-step). The proposed algorithm decomposes the safe RL problem into a convex optimization phase and a supervised learning phase, which yields a more stable training performance. A wide range of experiments on continuous robotic tasks shows that the proposed method achieves significantly better constraint satisfaction performance and better sample efficiency than baselines. The code is available at https://github.com/liuzuxin/cvpo-safe-rl.

1. Introduction

The past few years have witnessed great success of reinforcement learning (RL) (Mnih et al., 2013; Silver et al., 2017). However, deploying a trained RL policy to the real world is challenging. One of the major obstacles is to ensure the learned policy satisfies safety constraints. Safe RL studies the RL problem subject to certain constraints, where the agent aims to not only maximize the task reward return, but also limit the constraint violation rate to a certain level. However, learning a parametrized policy that satisfies constraints is not a trivial task, especially when the policy is represented by black-box neural networks (Liu et al., 2020; As et al., 2022; Chen et al., 2021).

Many researchers have pointed out that prior knowledge of the environment (Dalal et al., 2018; Cheng et al., 2019; Thananjeyan et al., 2021; Chen et al., 2021) and expert interventions (Saunders et al., 2017; Alshiekh et al., 2018; Wagener et al., 2021) are helpful to improve safety, but the required domain knowledge on the dynamics, constraints, or the existence of an expert oracle may not always be available. This paper studies the safe RL problem in a more general setting: learning a safe policy purely from interacting with the environment and receiving constraint violation signals. We aim to unveil and resolve the fundamental issues for safe RL from the constrained optimization perspective.

Most constrained optimization approaches for safe RL are under the primal-dual framework, which transforms the original constrained problem into an unconstrained one by introducing the dual variables (i.e., Lagrange multipliers) to penalize constraint violations (Chow et al., 2017; Liang et al., 2018; Tessler et al., 2018; Bohez et al., 2019; Ray et al., 2019). However, the primal-dual iterative optimization may run into numerical *instability* issues and lacks *optimality* guarantee for each policy iteration (Chow et al., 2018; Stooke et al., 2020). The instability usually comes from imbalanced learning rates of the primal and dual problem, and the optimality term means both *feasibility* (constraint satisfaction) and reward maximization. Another line of work approximate the constrained optimization problem with low-order Taylor expansions such that the dual variables could be solved efficiently (Achiam et al., 2017; Yang et al., 2020; Zhang et al., 2020) , but the induced approximations errors may yield poor constraint satisfaction performance in practice (Ray et al., 2019). In addition, these policy-gradient algorithms are on-policy by design, and extending them to a more sample efficient off-policy setting is non-trivial. Note that in safe RL context, sample efficiency means using both minimum constraint violation costs and minimum interaction samples to achieve the same level of rewards. As a result, a constrained RL optimization method that is 1) sample efficient, 2) stable and has 3) performance guarantees is absent in the literature.

To bridge the gap, this paper proposes the Constrained Variational Policy Optimization (CVPO) algorithm from the

[1]Carnegie Mellon University [2]Nuro Inc. [3]University of Illinois Urbana-Champaign. Correspondence to: Zuxin Liu <zuxinl@cmu.edu>, Ding Zhao <dingzhao@cmu.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

probabilistic inference view. CVPO transforms the safe RL problem to a convex optimization phase with optimality guarantees and a supervised learning phase with policy improvement bound and the robustness guarantee to recover to the feasible region, which ensure stable training performance. The off-policy implementation also gives rise to high sample-efficiency in practice. The main contributions of this work are summarized as follows:

1. To our best knowledge, this is the first work that formulates safe RL as a *probabilistic inference* problem, which enables us to leverage the rich toolbox of inference techniques to solve the safe RL problem. These techniques are used to overcome many drawbacks of previous primal-dual policy optimization fashion, such as unstable training and lack of optimality guarantee.

2. We propose a novel two-step algorithm in an Expectation Maximization (EM) style to naturally incorporate safety constraints during the policy training. An optimal and feasible non-parametric variational distribution is solved *analytically* during the E-step, and then the parametrized policy is trained via a *supervised learning* fashion during the M-step, which allows us to improve the policy with off-policy data and increase the sample efficiency.

3. We show the closed-form variational distribution in the E-step could be **computed efficiently** with **provable optimality guarantee**. The efficiency arises from the strict convexity of the dual problem in most cases, which ensures the uniqueness and optimality of the solution, and we did not find similar claims and proofs in the literature. Furthermore, the trust-region regularized policy improvement during the M-step gives us a worst-case constraint violation bound and robustness guarantee against worst-case training iterations.

4. We evaluate CVPO on a series of continuous control tasks. The empirical experiments demonstrate the effectiveness of the proposed method – more stable training, better constraint satisfaction, and up to 1000 times better sample efficiency than on-policy baselines.

2. Preliminary and Related Work

2.1. Constrained Markov Decision Processes

Constrained Markov Decision Processes (CMDPs) provide a mathematical framework to describe the safe RL problem (Altman, 1998), where the agents are enforced with restrictions on auxiliary safety constraint violation costs. A CMDP is defined by a tuple $(\mathcal{S}, \mathcal{A}, P, r, \gamma, \rho_0, C)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition kernel that specifies the transition probability $p(s_{t+1}|s_t, a_t)$ from state s_t to s_{t+1} under the action a_t, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, $\gamma \to [0, 1)$ is the discount factor, and $\rho_0 : \mathcal{S} \to [0, 1]$ is the distribution over the initial states. The last element C is a set of costs $\{c_i : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_{\geq 0}, i = 1, 2, ..., m\}$ for violating m constraints, which is the major difference between CMDP and traditional Markov Decision Process (MDP). Depending on the application, the cost $c_i \in C$ has different representations and physical meanings. For instance, it could be an indicator cost signal for being in an unsafe set of states and actions, or a continuous function of the distance w.r.t the constraint boundary. For simplicity of notation, we consider one universal constraint function $c \in C : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_{\geq 0}$ exists in the CMDP to characterize the corresponding constrained RL problem, which resembles the notation of reward function. All the definitions and theorems in this paper could be extended to multiple constraints as well.

Let $\pi(a|s)$ denote the policy, and $\tau = \{s_0, a_0, ..., \}$ denote the trajectory. We use shorthand $r_t = r(s_t, a_t)$ and $c_t = c_t(s_t, a_t)$ for simplicity. The discounted expect return of the reward under the policy π is $J_r(\pi) = \mathbb{E}_{\tau\sim\pi}[\sum_{t=0}^{\infty} \gamma^t r_t]$, and similarly the discounted expected return of the cost is $J_c(\pi) = \mathbb{E}_{\tau\sim\pi}[\sum_{t=0}^{\infty} \gamma^t c_t]$, where the initial state $s_0 \sim \rho_0$. The objective of a safe RL problem is to find the policy π^* that maximizes the expected cumulative rewards while limiting the costs incurred from constraint violations to a threshold $\epsilon_1 \in [0, +\infty)$:

$$\pi^* = \arg\max_{\pi} J_r(\pi), \quad s.t. \quad J_c(\pi) \leq \epsilon_1. \tag{1}$$

2.2. Related Work

Constrained RL optimization. Primal-dual approach is most commonly used in solving safe RL problems (Ding et al., 2020; Bohez et al., 2019). Generally, the primal-dual style algorithms alternate between optimizing the policy parameters and updating the dual variables, which are usually performed via gradient descent (Tessler et al., 2018; Liang et al., 2018; Zhang et al., 2020). Stooke et al. (2020) view the dual problem as a control system and propose a PID control method to update the Lagrange multipliers in a more stable way. Though the primal-dual framework is intuitive, the trained policy makes little safety guarantee with respect to both the converged policy and the behavior policy during each training iteration (Chow et al., 2018; Xu et al., 2021). Several works introduce a KL-regularized policy improvement mechanism and provide the worse-case performance bound (Achiam et al., 2017; Yang et al., 2020), however, the quadratic approximation of the original problem usually lead to high cost (Ray et al., 2019). In addition, the primal-dual approaches heavily rely on an accurate on-policy value estimation of constraints, so applying them to off-policy settings is not easy: one needs to backpropagate gradients from multiple Q-value functions to the policy network, which may cause instability issues.

RL as inference. Formulating RL as inference has been extensively studied recently (Levine, 2018). Haarnoja et al. (2017; 2018) perform exact inference over the probabilistic graphical model of RL via message passing. Abdolmaleki et al. (2018b;a) propose the Maximum a posteriori Policy Optimization (MPO) method to improve the policy with off-policy data in an Expectation-Maximization fashion, which can achieve comparable performance and better sample efficiency than on-policy methods. Song et al. (2019) extend MPO to on-policy setting, and Fellows et al. (2019) propose a variational inference (VI) framework for RL. While we believe the flourishing development of powerful RL as inference methods could provide a fresh view over the safe RL domain, however, a theoretical exploration of the connection between them has so far been lacking.

3. Constrained Variational Policy Optimization (CVPO)

We observe that under mild assumptions, safe RL could be viewed as a probabilistic inference problem, which yields CVPO — a generic approach to incorporate safety constraints in the inference step. We will detail our method in this section and show how CVPO inherits many theoretical benefits from both the RL as inference and the constrained RL domain.

3.1. Constrained RL as Inference

Before presenting the inference view, we first introduce the standard primal-dual perspective to solve CMDP, which transforms the objective (1) into a min-max optimization by introducing the Lagrange multiplier λ:

$$(\pi^*, \lambda^*) = \arg\min_{\lambda\geq 0}\max_{\pi} J_r(\pi) - \lambda(J_c(\pi) - \epsilon_1). \quad (2)$$

The core principle of primal-dual approaches is to solve the min-max problem iteratively. However, the optimal dual variable $\lambda = +\infty$ when $J_c(\pi) > \epsilon_1$ and $\lambda = 0$ when $J_c(\pi) < \epsilon_1$, so selecting a proper learning rate for λ is critical. Approximately solving the minimization also leads to suboptimal dual variables for each iteration. In addition, the non-stationary cost penalty term involving λ will make the policy gradient step in the primal problem hard to optimize, just as shown in the upper diagram of Fig. 1.

To tackle the above problems, we view the safe RL problem from the probabilistic inference perspective – inferring safe actions that result in "observed" high reward in states. This is done by introducing an optimality variable O to represent the event of maximizing the reward. Following similar probabilistic graphical models and notations in (Levine, 2018), we consider an infinite discounted reward formulation, where the likelihood of being optimal given a trajectory is proportional

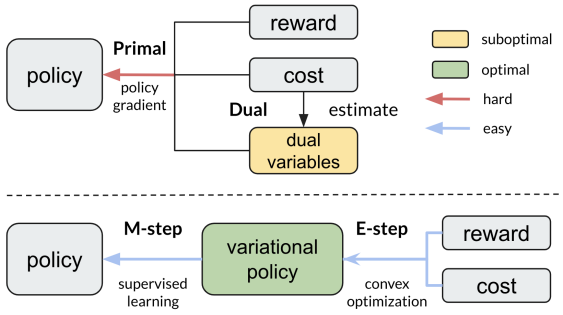


Figure 1. Primal-dual view (upper) and inference view (lower).

to the exponential of the discounted cumulative reward: $p(O = 1 \mid \tau) \propto \exp(\sum_t \gamma^t r_t/\alpha)$, where α is a temperature parameter. Denote the probability of a trajectory τ under the policy π as $p_\pi(\tau)$, then the lower bound of the log-likelihood of optimality under the policy π is (see Appendix A.1 for proof):

$$\log p_\pi(O = 1) = \log \int p(O = 1 \mid \tau) p_\pi(\tau) d\tau \quad (3)$$

$$\geq \mathbb{E}_{\tau\sim q}[\sum_{t=0}^{\infty} \gamma^t r_t] - \alpha D_{\mathrm{KL}}(q(\tau)\|p_\pi(\tau)) = \mathcal{J}(q, \pi) \quad (4)$$

where $q(\tau)$ is an auxiliary trajectory distribution and $\mathcal{J}(q, \pi)$ is the evidence lower bound (ELBO). To ensure safety, we limit the choices of $q(\tau)$ within a feasible distribution family that is subject to constraints. Recall that $c_t = c(s_t, a_t)$ is the cost for constraint violations. We define the **feasible distribution family** in terms of the threshold ϵ_1 as

$$\Pi_{\mathcal{Q}}^{\epsilon_1} := \{q(a|s) : \mathbb{E}_{\tau\sim q}[\sum_{t=0}^{\infty} \gamma^t c_t] < \epsilon_1, a \in \mathcal{A}, s \in \mathcal{S}\},$$

which is a set of all the state-conditioned action distributions that satisfy the safety constraint. Afterwards, by factorizing the trajectory distributions

$$q(\tau) = p(s_0) \prod_{t\geq 0} p(s_{t+1}|s_t, a_t) q(a_t|s_t), \forall q \in \Pi_{\mathcal{Q}}^{\epsilon_1},$$

$$p_{\pi_\theta}(\tau) = p(s_0) \prod_{t\geq 0} p(s_{t+1}|s_t, a_t) \pi_\theta(a_t|s_t) p(\theta),$$

where $\theta \in \Theta$ is the policy parameters, and $p(\theta)$ is a prior distribution, we obtain the following ELBO over the feasible state-conditioned action distribution $q(a|s)$ by cancelling the transitions:

$$\mathcal{J}(q, \theta) = \mathbb{E}_{\tau\sim q}\left[\sum_{t=0}^{\infty} \left(\gamma^t r_t - \alpha D_{\mathrm{KL}}(q(\cdot|s_t)\|\pi_\theta(\cdot|s_t))\right)\right] + \log p(\theta), \quad \forall q(a|s_t) \in \Pi_{\mathcal{Q}}^{\epsilon_1}. \quad (5)$$

Optimizing the new lower bound $\mathcal{J}(q,\theta)$ w.r.t q within the feasible distribution space $\Pi_{\mathcal{Q}}^{\epsilon_1}$ (E-step) and π within the parameter space Θ (M-step) iteratively via an Expectation-Maximization (EM) fashion yields the Constrained Variational Policy Optimization (CVPO) algorithm. We could interpret the inference formulation as answering "given future success in maximizing task rewards, what are the **feasible** actions most likely to have been taken?", instead of "what are the actions that could maximize task rewards while satisfying the constraints?" in the primal-dual formulation. The decoupling choice by separating the E-step and M-step provides more flexibility to optimize the control policies and select their representations. Similar idea is also adopted in previous works for standard RL setting (Abdolmaleki et al., 2018b;a).

Viewing safe RL as a variational inference problem has many benefits. As shown in the lower diagram of Fig. 1, we break the direct link between the inaccurate dual variable optimization and the difficult policy improvement and introduce a variational distribution in the middle to bridges the two steps, such that the policy improvement could be done via a much easier supervised learning fashion. The **key challenges** arise from the E-step because of the limitation of the variational distribution within a constrained set. However, we observe that the constrained q could be solved analytically, efficiently, with optimality and feasibility guarantee through convex optimization, as we show next.

3.2. Constrained E-step

The objective of this step is to find the optimal variational distribution $q \in \Pi_{\mathcal{Q}}^{\epsilon_1}$ to improve the return of task reward, while satisfying the safety constraint. At the $i\text{-}th$ iteration, we resort to perform a partial constrained E-step to maximize $\mathcal{J}(q,\theta)$ with respect to q by fixing the policy parameters $\theta = \theta_i$. We set the initial value of $q = \pi_{\theta_i}$ such that the return of task reward $Q_r^q(s,a)$ and cost $Q_c^q(s,a)$ could be estimated by: $Q_r^q(s,a) = Q_r^{\pi_{\theta_i}}(s,a) = \mathbb{E}_{\tau\sim\pi_{\theta_i}, s_0=s, a_0=a}\Big[\sum_{t=0}^{\infty}\gamma^t r_t\Big]$, $Q_c^q(s,a) = Q_c^{\pi_{\theta_i}}(s,a) = \mathbb{E}_{\tau\sim\pi_{\theta_i}, s_0=s, a_0=a}\Big[\sum_{t=0}^{\infty}\gamma^t c_t\Big]$, where the trajectory τ could be sampled from the replay buffer and thus the critics could be updated in an off-policy fashion. Given π_{θ_i}, $Q_r^{\pi_{\theta_i}}(s,a)$ and $Q_c^{\pi_{\theta_i}}(s,a)$, we further optimize $q(\cdot|s)$ by the following KL regularized objective:

$$\begin{aligned}\bar{\mathcal{J}}(q) &= \mathbb{E}_{\rho_q}\Big[\mathbb{E}_{q(\cdot|s)}\big[Q_r^{\pi_{\theta_i}}(s,a)\big] - \alpha D_{\mathrm{KL}}(q\|\pi_{\theta_i})\Big], \\ s.t.\quad & \mathbb{E}_{\rho_q}\Big[\mathbb{E}_{q(\cdot|s)}\big[Q_c^{\pi_{\theta_i}}(s,a)\big]\Big] \le \epsilon_1\end{aligned} \tag{6}$$

where $\rho_q(s)$ is the stationary state distribution induced by $q(a|s)$ and ρ_0. The constraint ensures the optimized distribution is within the feasible set $\Pi_{\mathcal{Q}}^{\epsilon_1}$. Solving the E-step (6) could be regarded as a KL-regularized constrained optimization problem. However, since the expected reward return term $\mathbb{E}_{q(\cdot|s)}\big[Q_r^{\pi_{\theta_i}}(s,a)\big]$ could be on an arbitrary scale, it is hard to choose a proper penalty coefficient α of the KL regularizer for different CMDP settings. Therefore, we impose a hard constraint ϵ_2 on the KL divergence between the non-parametric distribution $q(a|s)$ that to be optimized and the parametrized policy $\pi_{\theta_i}(a|s)$. Then the E-step yields the following constrained optimization:

$$\begin{aligned}\max_q \quad & \mathbb{E}_{\rho_q}\Big[\int q(a|s)Q_r^{\pi_{\theta_i}}(s,a)da\Big] \\ s.t. \quad & \mathbb{E}_{\rho_q}\Big[\int q(a|s)Q_c^{\pi_{\theta_i}}(s,a)da\Big] \le \epsilon_1 \\ & \mathbb{E}_{\rho_q}\Big[D_{\mathrm{KL}}(q(a|s)\|\pi_{\theta_i})\Big] \le \epsilon_2 \\ & \int q(a|s)da = 1, \quad \forall s\sim\rho_q\end{aligned} \tag{7}$$

where we have three constraints for this optimization problem in total. The **first safety constraint** in terms of ϵ_1 is to ensure the optimal non-parametric distribution belongs to the feasible set such that the safety constraints of CMDP could be satisfied. The **second regularization constraint** in terms of the KL threshold ϵ_2 aims to restrict the optimized variational distribution q within a trust region of the old policy distribution. The **last equality constraint** is to make sure the solved q is a valid action distribution across all the states. Intuitively, in E-step, we aim to find the optimal variational distribution that 1) maximizes the task rewards, 2) belongs to the feasible distribution family $\Pi_{\mathcal{Q}}^{\epsilon_1}$, and 3) stays within the trust region of the old policy.

Before solving the constrained optimization problem (7), we need to specify the representation of the variational distribution q. The inference formulation of the safe RL problem gives us the flexibility to choose its representation freely. Note that if we use a parametric representation of $q(a|s)$, then the E-step is similar to the policy updating in CPO, where we could regard optimizing q as updating the policy parameters from θ_i to θ_{i+1}. As such, the M-step is no longer required. However, as shown in CPO, the constrained optimization problem with a parametrized q is generally intractable, so approximations over the parameter space are usually required, which may lead to poor constraint satisfaction (Ray et al., 2019).

To avoid the performance degradation induced by approximation errors, we choose a non-parametric form for $q(a|s)$. Namely, given a state s, we use $|\mathcal{A}|$ variables for $q(\cdot|s)$ if the action space is finite. Otherwise, we sample K particles within the action space to represent the variational distribution for the continuous action space. By constructing a non-parametric form of q, we could see that problem (7) is a convex problem since the objective is linear and all the constraints are convex. Moreover, we could obtain the optimal (and mostly unique) solution in an analytical form (8) after solving a convex dual problem. We show the strong duality

guarantee under mild assumptions (see Appendix A.2):

Assumption 1. *(Slater's condition). There exists a feasible distribution $\bar{q} \in \Pi_{\mathcal{Q}}^{\epsilon_1}$ within the trust region of the old policy π_{θ_i}: $D_{\mathrm{KL}}(\bar{q} \| \pi_{\theta_i}) < \epsilon_2$.*

Theorem 1. *If assumption 1 holds, then the optimal variational distribution within $\Pi_{\mathcal{Q}}^{\epsilon_1}$ for problem (7) has the form:*

$$q_i^*(a|s) = \frac{\pi_{\theta_i}(a|s)}{Z(s)} \exp\left(\frac{Q_r^{\theta_i}(s,a) - \lambda^* Q_c^{\theta_i}(s,a)}{\eta^*}\right) \quad (8)$$

where $Z(s)$ is a constant normalizer to make sure q^* is a valid distribution, and the dual variables η^* and λ^* are the solutions of the following convex optimization problem:

$$\min_{\lambda,\eta \geq 0} \quad g(\eta,\lambda) = \lambda\epsilon_1 + \eta\epsilon_2 + \\ \eta \mathbb{E}_{\rho_q}\left[\log \mathbb{E}_{\pi_{\theta_i}}\left[\exp\left(\frac{Q_r^{\theta_i}(s,a) - \lambda Q_c^{\theta_i}(s,a)}{\eta}\right)\right]\right]. \quad (9)$$

Theorem 1 suggests the non-parametric variational distribution could be easily solved in close-form with optimality guarantee, since Assumption 1 ensures zero duality gap. Eq. (8) indicates that the optimal q is re-weighted based on the old policy π_{θ_i}, where the weights are controlled by $Q_r^{\theta_i}(s,a), Q_c^{\theta_i}(s,a), \eta, \lambda$. Note that the $Q_r^{\theta_i}(s,a)$ and $Q_c^{\theta_i}(s,a)$ are viewed as constants here as discussed in section 3.1. We can see that higher weights are given to the actions that have higher future task rewards and lower safety costs, where the weight between them is balanced by λ. Intuitively, the dual variable η serves as a temperature to control the flatness of the weights, such that the updated variational distribution would not be far away from the old policy or collapse to one action quickly. Higher η enforces stronger restrictions on the flatness, which makes sense since we limit the solution within a trust region. One exciting property of Theorem 1 is that we prove the strong convexity conditions of the dual problem, which guarantees the optimality and uniqueness of the solution, as shown below:

Theorem 2. *The multivariate dual function $g(\eta,\lambda)$ in (9) is convex on $\mathbb{R}^2_{\geq 0}$. It is strictly convex and has a unique optimal solution when (1) $Q_r^{\theta_i}(s,\cdot), Q_c^{\theta_i}(s,\cdot)$ are not constant functions; (2) $\forall C \in \mathbb{R}, \exists a_0, s.t. Q_r^{\theta_i}(s,a_0) \neq C \cdot Q_c^{\theta_i}(s,a_0)$; and (3) $\lambda < +\infty$. When the Slater condition in Assumption 1 holds, at least one optimal solution exists.*

Proof and discussions are in Appendix A.3. Each condition has corresponding meaning, as we explain in the following remarks.

Remark 1. *For the first condition, if $Q_r(s,\cdot)$ is a constant function, then the objective in (7) will always be a constant — the optimization becomes meaningless since all the distributions that within the trust region should be the same; if $Q_c(s,\cdot)$ is a constant function, then the safety constraint will be inactive, since no policy could change the feasibility status. In addition, if $Q_c(s,\cdot)$ is a constant, the gradient of the safety constraint dual variable λ will either be a positive constant when the problem is feasible (tends to make $\lambda \to 0$) or a negative constant when the problem is infeasible (tends to make $\lambda \to +\infty$).*

Remark 2. *The second condition indicates that if the task reward Q_r value is proportional to the cost Q_c value everywhere, then multiple optimal solutions may exist on the KL constraint boundary or the safety constraint boundary. Intuitively, the problem becomes a bounded convex optimization — maximizing the risks $\mathbb{E}_q[Q_c(s,a)]$ within the trust region defined by ϵ_2 and a ball defined by ϵ_1. Though the solutions may not be unique, the optimality could still be guaranteed as long as the Slater's condition 1 holds.*

Remark 3. *The third condition is equivalent to the violation of our Slater's condition assumption 1 — no distribution exists within the trust region that satisfies the safety constraints. In that sense, the gradient of the dual variable λ will always be negative, and thus $\lambda \to +\infty$, which tends to impose huge penalties on safety critics Q_c. In practice, this rarely happens since we could select a large KL threshold for E-step and choose a much smaller trust region size for M-step, such that n-step robustness could be guaranteed and the Slater's condition could be easily met, as we show in Appendix A.7.*

From Theorem 2 and above remarks, we can see the conditions for strict convexity are easy to be satisfied: the Q value functions should not be constants and the reward return will not be proportional to the cost return in most safe RL settings. The strict convexity analysis guarantees the uniqueness of the optimal solution in most situations and provides some other theoretical implications of the E-M procedure, such as the convergence to a stationary point (Sriperumbudur et al., 2009). Furthermore, the Slater condition indicates that $\lambda < +\infty$ and the existence of an optimal solution, which ensures the dual problem could be solved efficiently via any convex optimization tools. We believe this finding is original and non-trivial, which also lays the theoretical foundation of the outstanding empirical performance of our method, as we show in Sec. 4.

3.3. M-step

After E-step, we obtain an optimal feasible variational distribution $q_i^*(\cdot|s)$ for each state by solving (7). In M-step, we aim to improve the ELBO (5) w.r.t the policy parameter θ. By dropping the terms in Eq. (5) that are independent of θ, we obtain the following M-step objective:

$$\bar{\mathcal{J}}(\theta) = \mathbb{E}_{\rho_q}\left[\alpha \mathbb{E}_{q_i^*(\cdot|s)}\left[\log \pi_\theta(a|s)\right]\right] + \log p(\theta) \quad (10)$$

where α is a temperature parameter to balance the weight between likelihood and prior, and $q_i^*(\cdot|s)$ serve as the weights

for the samples π_θ. Note that the objective could be viewed as a weighted maximum a-posteriori problem, which is similar to MPO (Abdolmaleki et al., 2018b;a). Since optimizing $\bar{\mathcal{J}}(\theta)$ is a supervised learning problem, we could use any prior $p(\theta)$ to regularize the policy and use any optimization methods to solve (10), depending on the policy parametrization. Particularly, we adopt a Gaussian prior around the old policy parameter θ_i in this paper: $\theta \sim \mathcal{N}(\theta_i, \frac{F_{\theta_i}}{\alpha\beta})$, where F_{θ_i} is the Fisher information matrix and β is a positive constant. With this Gaussian prior, we obtain the following generalized M-step (see Appendix A.4 for detailed proof):

$$\max_\theta \mathbb{E}_{\rho_q}\Big[\mathbb{E}_{q_i^*(\cdot|s)}\big[\log \pi_\theta(a|s)\big] - \beta D_{\mathrm{KL}}(\pi_{\theta_i}\|\pi_\theta)\Big]. \quad (11)$$

Similar to the E-step, we convert the soft KL regularizer to a hard KL constraint to deal with different objective scales:

$$\begin{aligned} \max_\theta \quad & \mathbb{E}_{\rho_q}\Big[\mathbb{E}_{q_i^*(\cdot|s)}\big[\log \pi_\theta(a|s)\big]\Big] \\ s.t. \quad & \mathbb{E}_{\rho_q}\big[D_{\mathrm{KL}}(\pi_{\theta_i}(a|s)\|\pi_\theta(a|s))\big] \le \epsilon. \end{aligned} \quad (12)$$

The regularizer is important to improve the updating robustness and prevent the policy from overfitting, as we will show in the experiment section 4.4.

3.4. Theoretical Analysis

CVPO updates the policy by maximizing KL-regularized objective functions in an EM manner, which brings it two advantages over primal-dual methods – the ELBO improvement guarantee and the worst-case constraint satisfaction bound (see Appendix A.5 & A.6 for details).

Proposition 1. *Suppose $\pi_{\theta_{i-1}}, \pi_{\theta_i}$ satisfy the Slater's condition, then the ELBO in Eq. (5) is guaranteed to be non-decreasing: $J(q_i, \theta_{i+1}) \ge J(q_{i-1}, \theta_i)$.*

Proposition 1 provides the policy improvement guarantee for the reward. Regarding the constraint violation cost, we have the following bound holds for CVPO:

Proposition 2. *Suppose $\pi_{\theta_i} \in \Pi_{\mathcal{Q}}^{\epsilon_1}$. $\pi_{\theta_{i+1}}$ and π_{θ_i} are related by the M-step (12), then we have the upper bound:*

$$J_c(\pi_{\theta_{i+1}}) \le \epsilon_1 + \frac{[(1-\gamma) + \sqrt{2\epsilon}\gamma]\delta_c^{\pi_{\theta_{i+1}}}}{(1-\gamma)^2} \quad (13)$$

where $\delta_c^{\pi_{\theta_{i+1}}} = \max_s |\mathbb{E}_{a\sim\pi_{\theta_{i+1}}}[A_c^{\theta_i}(s,a)]|$, $A_c^{\theta_i}(s,a)$ is the cost advantage function of π_{θ_i}.

We can see the worst-case episodic constraint violation upper bound for $\pi_{\theta_{i+1}}$ is related to the trust region size ϵ and the worst-case approximation error $\delta_c^{\theta_{i+1}}$. Though we inherit a similar bound as CPO, our method ensures the optimality of each update and could be done in a more sample-efficient off-policy fashion. In addition, we observe that by selecting proper KL constraints ϵ in the M-step, we have the following policy improvement robustness property:

Proposition 3. *Suppose $\pi_{\theta_i} \in \Pi_{\mathcal{Q}}^{\epsilon_1}$. $\pi_{\theta_{i+1}}$ and π_{θ_i} are related by the M-step. If $\epsilon < \epsilon_2$, where ϵ, ϵ_2 are the KL threshold in M-step and E-step respectively, then the variational distribution q_{i+1}^* in the next iteration is guaranteed to be feasible and optimal.*

Proposition 3 provides us with an interesting perspective and a theoretical guarantee of the policy improvement robustness – no matter how bad the M-step update in one iteration is, we are still able to recover to an optimal policy within the feasible region. Furthermore, We find that under the Gaussian policy assumption, multiple steps robustness could also be achieved (see Appendix A.7 for proof and figure illustrations). The monotonic reward improvement guarantee, the worst-case cost bound and the policy updating robustness together ensure the **training stability** of CVPO.

3.5. Practical Implementation

Another advantage of the proposed scheme is that the framework can be easily applied to a more sample efficient off-policy setting, which is important for safe RL because worse samples efficiency usually indicates more constraint violations. Practically, the stationary state distribution ρ_q could be approximated by the samples from the replay buffer, which yields the off-policy version of CVPO. Algo. 1 highlights the key steps of one training epoch for continuous action space. To solve tasks with discrete action space, we only need to replace summation with integration over action space. We also decompose the KL constraints into separate constraints on mean and covariance during the M-step (12) to achieve better exploration-exploitation trade-off (Abdolmaleki et al., 2018b). For more implementation details and the full algorithm, please refer to Appendix B.1.

Algorithm 1 CVPO Training for One Epoch

Input: batch size B, particle size K, policy parameter θ_i
Output: Updated policy parameter θ_{i+1}

Sample B transitions from replay buffer
▷ *E-step begins*
Update $Q_r^{\theta_i}, Q_c^{\theta_i}$ via Bellman backup
for $b = 1, ..., B$ **do**
  Sample K actions $\{a_1, ..., a_K\}$ for s_b
  Compute $\{Q_r^{\theta_i}(s_b, a_k), Q_c^{\theta_i}(s_b, a_k); k = 1, ..., K\}$.
end for
Compute optimal dual variables η^*, λ^* by solving the convex optimization problem (9)
Compute the optimal variational distribution for each state $\{q^*(\cdot|s_b); b = 1, ..., B\}$ by Eq. (8)
▷ *M-step begins*
Update policy from π_{θ_i} to $\pi_{\theta_{i+1}}$ via supervised learning objective (12)

4. Experiment

Motivated by previous works (Ray et al., 2019; Achiam et al., 2017; Zhang et al., 2020; Stooke et al., 2020; Gronauer, 2022), we designed 5 robotic control tasks with different difficulty levels to show the effectiveness of our method. Detail description of the task environments and method implementations could be found in Appendix B. The code is also available at `https://github.com/liuzuxin/cvpo-safe-rl`.

4.1. Tasks

Circle Task. A car robot is expected to move on a circle in clock-wise direction. The agent will receive higher rewards by increasing the velocity and approaching the boundary of the circle. The safety zone is defined by two parallel plane boundaries that are intersected with the circle. The agent receives a cost equals 1 upon leaving the safety zone. The observation space includes the car's ego states and the sensing of the boundary. We name this task as `Car-Circle`.

Goal Task. The agent aims to reach the goal buttons while avoiding static surrounding obstacles. After the agent press the correct button, the environment will randomly select a new goal button. The agent will receive positive rewards for moving towards the goal button, and a bonus will be given for successfully reaching the goal. A cost will be penalized for violating safety constraints — colliding with the static obstacles or pressing the wrong button. The observation space includes the agent's ego states and the sensing information about the obstacles and the goal, which is represented by pseudo LiDAR points. We use a Point robot and a Car robot in this environment. We name them as `Point-Goal` and `Car-Goal`.

Button Task. This task is a harder version of `Goal`, where dynamic obstacles are presented in this task. The dynamic obstacles are moving along a circle continuously, and the agent needs to reach the goal while avoiding both static and dynamic obstacles. We can see that the `Button` task is harder than `Circle` and `Goal`, since the agent is required to infer the surrounding obstacles' states from raw sensing data. We also use a Point robot and a Car robot, and we name the tasks as `Point-Button` and `Car-Button`.

4.2. Baselines

Off-policy baselines. To better understand the role of variational inference, we use three off-policy primal-dual-based baselines. In contrast to the vanilla Lagrangian-based approaches (Ray et al., 2019), we use a stronger baseline — PID-Lagrangian approach (Stooke et al., 2020) — to update the dual variables, which can achieve more stable training. Since the Lagrangian approach could be easily included in most existing RL methods in principle, we adopt three well-known base algorithms: Soft Actor Critic (SAC) (Haarnoja et al., 2018), Deep Deterministic Policy Gradient (DDPG) (Lillicrap et al., 2015), and Twin Delayed DDPG (TD3) (Fujimoto et al., 2018). We name the PID-Lagrangian-augmented safe RL versions as `SAC-Lag`, `DDPG-Lag` and `TD3-Lag`.

On-policy baselines. Constrained Policy Optimization (CPO) (Achiam et al., 2017) is used as the on-policy safe RL baseline, since their KL-regularized policy updating algorithm is closely related to us. Additionally, we use two Lagrangian-based baselines that are commonly used in previous works — `TRPO-Lag` and `PPO-Lag`, which are modified from Trust Region Policy Optimization (Schulman et al., 2015) and Proximal Policy Gradient (Schulman et al., 2017). We also use `TRPO` as the unconstrained RL baseline to show what is the best task reward performance when ignoring the constraints.

For fair comparison, we use the same network sizes of the policy and critics for all the methods, including CVPO (our method). The safety critics updating rule and the discounting factor are also the same for all off-policy methods. For detailed hyperparameters, please refer to Appendix B.3.

4.3. Results

We separate the comparison with off-policy and on-policy baselines because off-policy methods are more sample efficient, so the scales for them are different and thus making it hard to distinguish the plots. Due to the page limit, we only present some results on a subset of tasks in this section. The complete experimental results for each comparison on all tasks could be found in the Appendix C.

Comparison to off-policy baselines. Fig. 2 shows the training curves for off-policy safe RL approaches. The first row is the undiscounted episodic task reward (the higher, the better). The second row is the undiscounted episodic cost (# of constraint violations), where the blue dashed line is the target cost threshold. We can clearly see that CVPO outperforms the off-policy baselines in terms of the constraint satisfaction while maintaining high episodic reward, which validates the optimality and feasibility guarantee of our method. Note that all the off-policy baselines use the same network architecture and sizes for Q_r and Q_c critics, and the major difference between them is the policy optimization. The training curves also demonstrate better stability of our approach, as we theoretically analyzed in section 3.4.

Comparison to on-policy baselines. Fig. 3 shows the training curves for on-policy baselines. Note that the curves for our method (CVPO) are the same as the ones in Fig. 2, and they look to be squeezed because other on-policy baselines require much more samples to converge. Among the

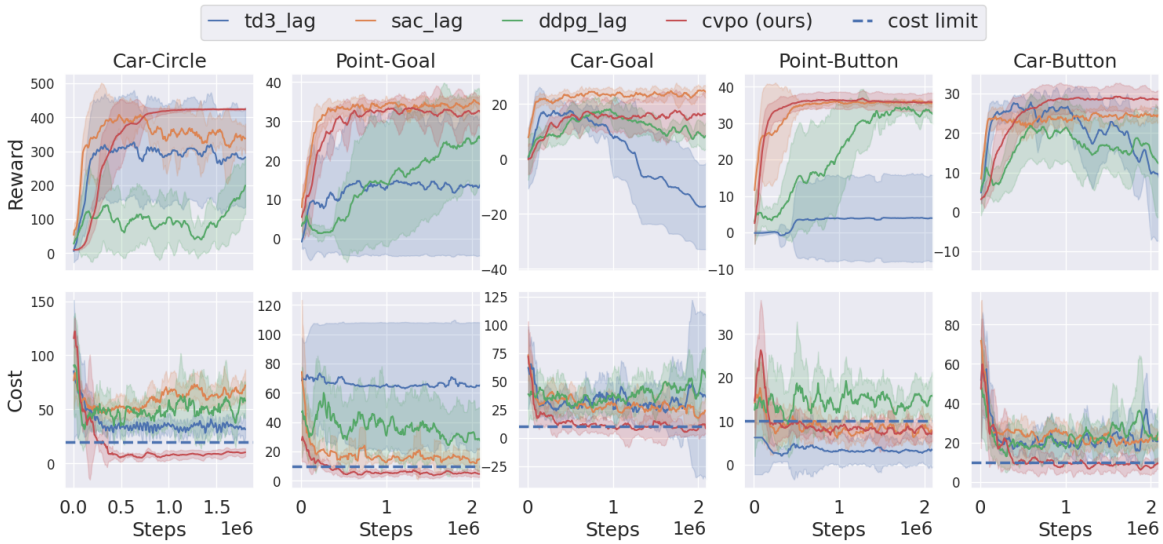


Figure 2. Training curves for off-policy baselines comparison. Each column corresponds to an environment. The curves are averaged over 10 random seeds, where the solid lines are the mean and the shadowed areas are the standard deviation.

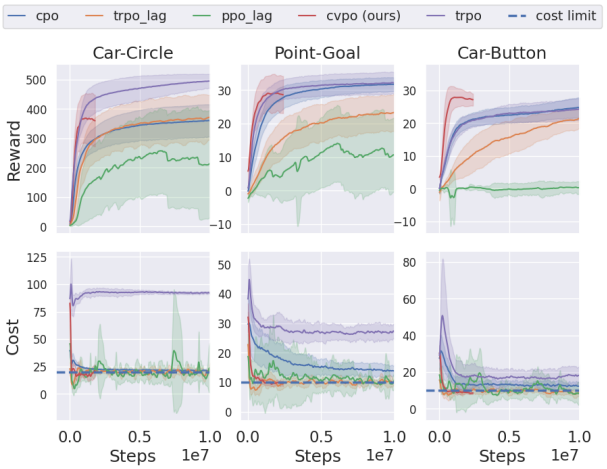


Figure 3. Training curves for on-policy baselines comparison.

baselines, TRPO-Lag and PPO-Lag have better constraint satisfaction performance than CPO, though PPO-Lag suffers from large variance and low task reward. CPO fails to satisfy the constraint for the Goal and Button tasks, which is probably caused by the approximation error during the policy update, as we discussed in section 3.2. Surprisingly, our method can achieve comparable task performance to the *unconstrained RL* baseline – TRPO (purple curves), while maintaining safety with much fewer constraint violations than on-policy safe RL approaches.

Fig. 4 demonstrates the **efficacy** of utilizing each cost – how much task rewards we could obtain given a budget of constraint violations. The curves that approach to the upper left are better because fewer costs are required to achieve high rewards. We can clearly see that CVPO outperforms baselines with large margin among all tasks – we use **100 $\sim$ 1000 times less** cumulative constraint violations to obtain the same task reward. Note that the x-axis is on the log-scale.

Convergence cost comparison. Fig. 5 shows the box plot of each method's constraint satisfaction performance after convergence. We define convergence as the last 20% training steps. The box plot – also called whisker plot – presents a five-number summary of the data. The five-number summary is the minimum (the lower solid line), first quartile (the lower edge of the box), median (the solid line in the box), third quartile (the upper edge of the box), and maximum (the upper solid line). As shown in the figure, on-policy baselines have better constraint satisfaction performance than off-policy baselines, which indicates that extending the primal-dual framework to off-policy settings is non-trivial. We find that our method meets the safety requirement better and with smaller variance than most baselines. Interestingly, we could observe that even **the third quartile of cost of our approach is below the target cost threshold**, which indicates that CVPO satisfies constraints state-wise, rather than in expectation as baselines. The reason is that we sample a mini-batch from the replay buffer to compute the optimal non-parametric distribution for these states respectively, while baseline approaches directly optimize the policy to satisfy the constraints in expectation.

4.4. Ablation study

The role of KL regularizer in M-step. Fig. 6 shows the importance of adding the KL constraint during the policy improvement phase. Since the M-step is essentially a super-

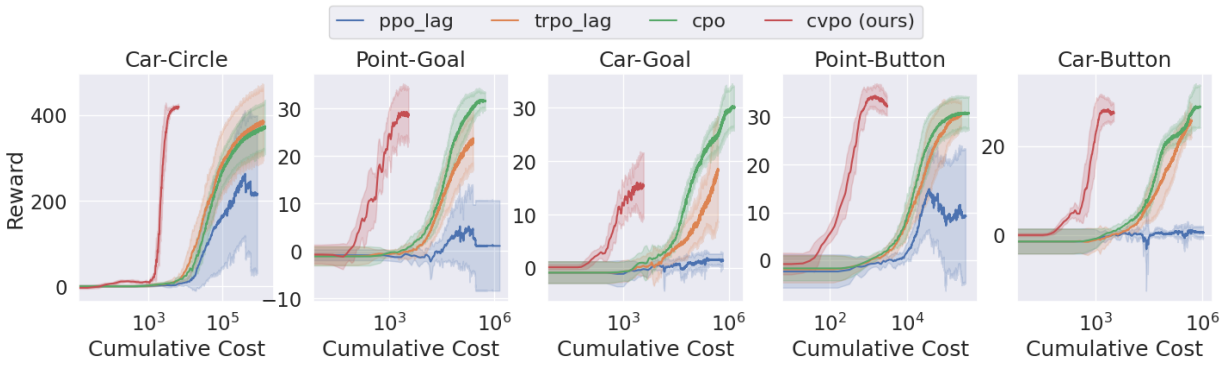


Figure 4. Reward versus cumulative cost (log-scale).

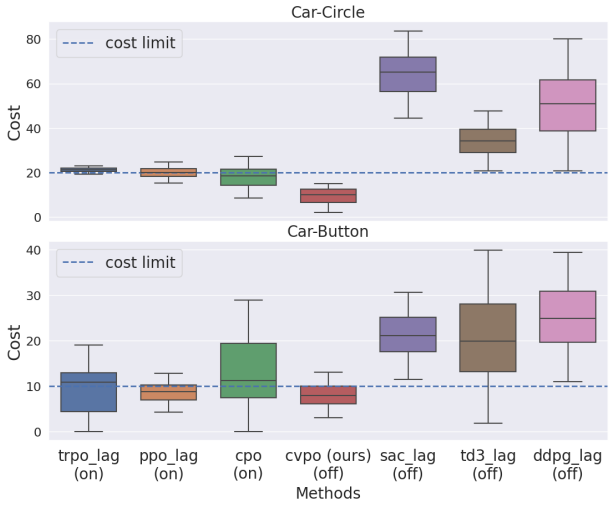


Figure 5. Box plot of the convergence cost. (on) and (off) denotes on-policy and off-policy method, respectively.

vised learning problem – fitting the policy with the optimal distribution solved in the E-step, the policy may easily collapse to local optimum action distributions computed from the current batch of data. Without the KL constraints, the worst-case performance bound in Proposition 2 tends to be infinite and the robustness guarantee in Proposition 3 does not hold, so the agent may fail to satisfy the constraints (Car-Button task). In addition, the policy overfitting prevent the agent from exploring more rewarding trajectories and thus lead to low reward (Car-Circle and Point-Goal tasks).

5. Conclusion

We show the safe RL problem can be decomposed into a convex optimization phase with a non-parametric variational distribution and a supervised learning phase. We show the unique advantages of constrained variational policy optimization: 1) high **sample-efficiency** from the off-policy variant of the approach; 2) **stability** ensured by the monotonic reward improvement guarantee, worst-case constraint

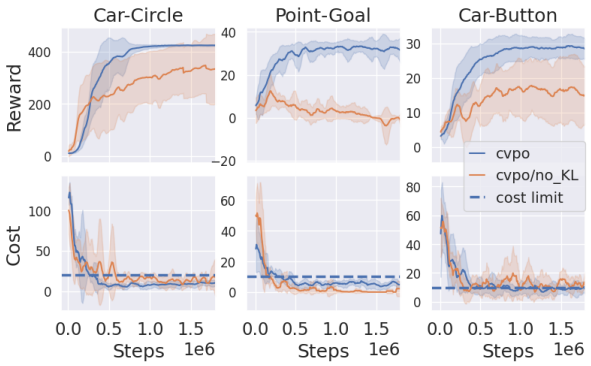


Figure 6. Ablation study of the KL constraint in M-step.

violation bound, and the policy updating robustness; and 3) with theoretical **optimality and feasibility guarantees** by solving a provable and mostly strict convex optimization problem in the E-step. We validate the proposed method with extensive experiments, and the results demonstrate the better empirical performance of our method than previous primal-dual approaches in terms of constraint satisfaction and sample efficiency.

The limitations include that our method would be more computationally expensive than primal-dual approaches since a convex optimization problem need to be solved in the E-step. In addition, the performance would be heavily dependent on the quality of reward and safety critics' estimation due to the off-policy training style. How to improve the critic's estimation of future constraint violations in safe RL is an important and promising topic. We believe our work can provide a new perspective in the safe RL field and inspire more exciting research in this direction.

Acknowledgements

We gratefully acknowledge support from the NSF CAREER CNS-2047454, NSF grant No.1910100, NSF CNS No.2046726, C3 AI, and the Alfred P. Sloan Foundation. We also would like to thank Jiacheng Zhu, Yufeng Zhang, Gokul Swamy for their help in various aspects of this paper.

References

Abdolmaleki, A., Springenberg, J. T., Degrave, J., Bohez, S., Tassa, Y., Belov, D., Heess, N., and Riedmiller, M. Relative entropy regularized policy iteration. *arXiv preprint arXiv:1812.02256*, 2018a.

Abdolmaleki, A., Springenberg, J. T., Tassa, Y., Munos, R., Heess, N., and Riedmiller, M. Maximum a posteriori policy optimisation. *arXiv preprint arXiv:1806.06920*, 2018b.

Achiam, J., Held, D., Tamar, A., and Abbeel, P. Constrained policy optimization. In *International Conference on Machine Learning*, pp. 22–31. PMLR, 2017.

Alshiekh, M., Bloem, R., Ehlers, R., Könighofer, B., Niekum, S., and Topcu, U. Safe reinforcement learning via shielding. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Altman, E. Constrained markov decision processes with total cost criteria: Lagrangian approach and dual linear program. *Mathematical methods of operations research*, 48(3):387–417, 1998.

As, Y., Usmanova, I., Curi, S., and Krause, A. Constrained policy optimization via bayesian world models. *arXiv preprint arXiv:2201.09802*, 2022.

Bohez, S., Abdolmaleki, A., Neunert, M., Buchli, J., Heess, N., and Hadsell, R. Value constrained model-free continuous control. *arXiv preprint arXiv:1902.04623*, 2019.

Chen, B., Liu, Z., Zhu, J., Xu, M., Ding, W., and Zhao, D. Context-aware safe reinforcement learning for non-stationary environments. *arXiv preprint arXiv:2101.00531*, 2021.

Cheng, R., Orosz, G., Murray, R. M., and Burdick, J. W. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3387–3395, 2019.

Chow, Y., Ghavamzadeh, M., Janson, L., and Pavone, M. Risk-constrained reinforcement learning with percentile risk criteria. *The Journal of Machine Learning Research*, 18(1):6070–6120, 2017.

Chow, Y., Nachum, O., Duenez-Guzman, E., and Ghavamzadeh, M. A lyapunov-based approach to safe reinforcement learning. *arXiv preprint arXiv:1805.07708*, 2018.

Corless, R. M., Gonnet, G. H., Hare, D. E., Jeffrey, D. J., and Knuth, D. E. On the lambertw function. *Advances in Computational mathematics*, 5(1):329–359, 1996.

Dalal, G., Dvijotham, K., Vecerik, M., Hester, T., Paduraru, C., and Tassa, Y. Safe exploration in continuous action spaces. *arXiv preprint arXiv:1801.08757*, 2018.

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

Fellows, M., Mahajan, A., Rudner, T. G., and Whiteson, S. Virel: A variational inference framework for reinforcement learning. *Advances in Neural Information Processing Systems*, 32:7122–7136, 2019.

Fujimoto, S., Hoof, H., and Meger, D. Addressing function approximation error in actor-critic methods. In *International Conference on Machine Learning*, pp. 1587–1596. PMLR, 2018.

Gronauer, S. Bullet-safety-gym: Aframework for constrained reinforcement learning. 2022.

Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, pp. 1352–1361. PMLR, 2017.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. 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.

Levine, S. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*, 2018.

Liang, Q., Que, F., and Modiano, E. Accelerated primal-dual policy optimization for safe reinforcement learning. *arXiv preprint arXiv:1802.06480*, 2018.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Liu, Z., Zhou, H., Chen, B., Zhong, S., Hebert, M., and Zhao, D. Constrained model-based reinforcement learning with robust cross-entropy method. *arXiv preprint arXiv:2010.07968*, 2020.

Luenberger, D. G. *Optimization by vector space methods*, pp. 213–216. John Wiley & Sons, 1997.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Munos, R., Stepleton, T., Harutyunyan, A., and Bellemare, M. G. Safe and efficient off-policy reinforcement learning. *arXiv preprint arXiv:1606.02647*, 2016.

Myung, I. J. Tutorial on maximum likelihood estimation. *Journal of mathematical Psychology*, 47(1):90–100, 2003.

Ray, A., Achiam, J., and Amodei, D. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 7, 2019.

Saunders, W., Sastry, G., Stuhlmueller, A., and Evans, O. Trial without error: Towards safe reinforcement learning via human intervention. *arXiv preprint arXiv:1707.05173*, 2017.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. PMLR, 2015.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

Song, H. F., Abdolmaleki, A., Springenberg, J. T., Clark, A., Soyer, H., Rae, J. W., Noury, S., Ahuja, A., Liu, S., Tirumala, D., et al. V-mpo: On-policy maximum a posteriori policy optimization for discrete and continuous control. *arXiv preprint arXiv:1909.12238*, 2019.

Sriperumbudur, B. K., Fukumizu, K., Gretton, A., Schölkopf, B., and Lanckriet, G. R. On integral probability metrics,\phi-divergences and binary classification. *arXiv preprint arXiv:0901.2698*, 2009.

Stooke, A., Achiam, J., and Abbeel, P. Responsive safety in reinforcement learning by pid lagrangian methods. In *International Conference on Machine Learning*, pp. 9133–9143. PMLR, 2020.

Tessler, C., Mankowitz, D. J., and Mannor, S. Reward constrained policy optimization. *arXiv preprint arXiv:1805.11074*, 2018.

Thananjeyan, B., Balakrishna, A., Nair, S., Luo, M., Srinivasan, K., Hwang, M., Gonzalez, J. E., Ibarz, J., Finn, C., and Goldberg, K. Recovery rl: Safe reinforcement learning with learned recovery zones. *IEEE Robotics and Automation Letters*, 6(3):4915–4922, 2021.

Wagener, N., Boots, B., and Cheng, C.-A. Safe reinforcement learning using advantage-based intervention. *arXiv preprint arXiv:2106.09110*, 2021.

Xu, T., Liang, Y., and Lan, G. Crpo: A new approach for safe reinforcement learning with convergence guarantee. In *International Conference on Machine Learning*, pp. 11480–11491. PMLR, 2021.

Yang, T.-Y., Rosca, J., Narasimhan, K., and Ramadge, P. J. Projection-based constrained policy optimization. *arXiv preprint arXiv:2010.03152*, 2020.

Zhang, Y., Vuong, Q., and Ross, K. First order constrained optimization in policy space. *Advances in Neural Information Processing Systems*, 2020.

Zhang, Y., Liu, W., Chen, Z., Li, K., and Wang, J. On the properties of kullback-leibler divergence between gaussians. *arXiv preprint arXiv:2102.05485*, 2021.

A. Proofs and Discussions

A.1. Proof of the evidence lower bound (ELBO) in section 3.1

Denote the probability of a trajectory τ under the policy π as $p_\pi(\tau) = p(s_0)\prod_{t\geq 0} p(s_{t+1} \mid s_t, a_t)\pi(a_t \mid s_t)$, then the lower bound of the log-likelihood of optimality given the policy π is

$$\log p_\pi(O=1) = \log \int p(O=1 \mid \tau) p_\pi(\tau) d\tau \tag{14}$$

$$= \log \mathbb{E}_{\tau\sim q}\left[\frac{p(O=1 \mid \tau)p_\pi(\tau)}{q(\tau)}\right] \tag{15}$$

$$\geq \mathbb{E}_{\tau\sim q} \log \frac{p(O=1 \mid \tau)p_\pi(\tau)}{q(\tau)} \tag{16}$$

$$= \mathbb{E}_{\tau\sim q} \log p(O=1 \mid \tau) + \mathbb{E}_{\tau\sim q} \log \frac{p_\pi(\tau)}{q(\tau)} \tag{17}$$

$$\propto \mathbb{E}_{\tau\sim q}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right] - \alpha D_{\mathrm{KL}}(q(\tau)\|p_\pi(\tau)) = \mathcal{J}(q,\pi) \tag{18}$$

where inequality (16) follows Jensen's inequality, $q(\tau)$ is an auxiliary trajectory distribution.

A.2. Proof and discussion of Theorem 1 — the optimal variational distribution and the dual function

Recall that the objective in E-step:

$$\begin{aligned} \max_q \quad & \mathbb{E}_{\rho_q}\Big[\int q(a|s) Q_r^{\pi_{\theta_i}}(s,a) da\Big] \\ s.t. \quad & \mathbb{E}_{\rho_q}\Big[\int q(a|s) Q_c^{\pi_{\theta_i}}(s,a) da\Big] \leq \epsilon_1 \\ & \mathbb{E}_{\rho_q}\Big[D_{\mathrm{KL}}(q(a|s)\|\pi_{\theta_i}(a|s))\Big] \leq \epsilon_2 \\ & \int q(a|s) da = 1, \quad \forall s \sim \rho_q. \end{aligned} \tag{19}$$

Then we prove the optimal variational distribution analytical form and its dual function in Theorem 1.

Proof. To solve the constrained optimization problem, we first convert it to the equivalent Lagrangian function:

$$L(q,\lambda,\eta,\kappa) = \int \rho_q(s) \int q(a|s) Q_r^{\pi_{\theta_i}}(s,a) da ds \tag{20}$$

$$+ \lambda\left(\epsilon_1 - \int \rho_q(s) \int q(a|s) Q_c^{\pi_{\theta_i}}(s,a) da ds\right) \tag{21}$$

$$+ \eta\left(\epsilon_2 - \int \rho_q(s) \int q(a|s) \log \frac{q(a|s)}{\pi_{\theta_i}(a|s)} da ds\right) \tag{22}$$

$$+ \kappa\left(1 - \int \rho_q(s) \int q(a|s) da ds\right), \tag{23}$$

where λ, η, κ are the Lagrange multipliers for the constraints. Since the objective is linear and all constraints are convex (note that KL is convex), the E-step optimization problem is convex. Then we obtain the equivalent dual problem:

$$\min_{\lambda,\eta,\kappa} \max_q L(q,\lambda,\eta,\kappa). \tag{24}$$

Take the derivative of Lagrangian function w.r.t q:

$$\frac{\partial L}{\partial q} = Q_r^{\pi_{\theta_i}}(s,a) - \lambda Q_c^{\pi_{\theta_i}}(s,a) - \eta - \kappa - \eta \log \frac{q(a|s)}{\pi_{\theta_i}(a|s)}. \tag{25}$$

Let Eq. (25) be zero, we have the form of the optimal q distribution:

$$q^*(a|s) = \pi_{\theta_i}(a|s) \exp\left(\frac{Q_r^{\pi_{\theta_i}}(s,a) - \lambda Q_c^{\pi_{\theta_i}}(s,a)}{\eta}\right) \exp\left(-\frac{\eta+\kappa}{\eta}\right), \tag{26}$$

where $\exp\left(-\frac{\eta+\kappa}{\eta}\right)$ could be viewed as a normalizer for $q(a|s)$ since it is a constant that is independent of q. Thus, we obtain the following form of the normalizer by integrating the optimal q:

$$\exp\left(\frac{\eta+\kappa}{\eta}\right) = \int \pi_{\theta_i}(a|s) \exp\left(\frac{Q_r^{\pi_{\theta_i}}(s,a) - \lambda Q_c^{\pi_{\theta_i}}(s,a)}{\eta}\right) da, \tag{27}$$

$$\frac{\eta+\kappa}{\eta} = \log \int \pi_{\theta_i}(a|s) \exp\left(\frac{Q_r^{\pi_{\theta_i}}(s,a) - \lambda Q_c^{\pi_{\theta_i}}(s,a)}{\eta}\right) da. \tag{28}$$

Take the optimal q distribution in Equation (26) and $\frac{\eta+\kappa}{\eta}$ in Equation (28) back to the Lagrangian function (23), we can find that most of the terms are cancelled out, and obtain the dual function $g(\eta, \lambda)$,

$$g(\eta,\lambda) = \lambda\epsilon_1 + \eta\epsilon_2 + \eta \int \rho_q(s) \log \int \pi_{\theta_i}(a|s) \exp\left(\frac{Q_r^{\pi_{\theta_i}}(s,a) - \lambda Q_c^{\pi_{\theta_i}}(s,a)}{\eta}\right) dads. \tag{29}$$

The optimal dual variables are calculated by

$$\eta^*, \lambda^* = \arg\min_{\eta,\lambda} g(\eta,\lambda). \tag{30}$$

□

A good property is that the dual function is convex (as we will prove in Appendix A.3), so we could use off-the-shelf convex optimization tools to solve the dual problem. Also, under the Slater's assumption (1) — there exists a feasible distribution $\bar{q} \in \Pi_{\mathcal{Q}}$ within the trust region of the old policy π_{θ_i}: $D_{\mathrm{KL}}(\bar{q}\|\pi_{\theta_i}) \leq \epsilon_2$, we could solve the optimal dual variables efficiently.

Lemma 1. *(Strong duality). If the Slater condition in Assumption (1) holds, then the strong duality holds for the original problem (7) and the dual problem (24).*

Lemma 1 implies that the optimality of our closed form non-parametric distribution in Eq. (26) could be guaranteed, since there is zero duality gap between the primal and dual problem.

Based on above proofs and lemma, we could easily obtain Theorem 1. Note that Theorem 1 is similar to the one in FOCOPS (Zhang et al., 2020) but with several major differences: 1) they use a parametric policy as the optimization objective in (8) instead of our non-parametric variational distribution. The parametrized form makes the problem hard to solve analytically, and thus first-order approximation over the policy parameters is required; 2) we provide an analytical form of the dual function and prove it is convex, such that the dual variables η, λ are guaranteed to be optimal by convex optimization, while they use a gradient-based method to solve the dual variables, which is similar to the Lagrangian-based methods in the literature (Stooke et al., 2020); 3) they consider the on-policy advantage estimations in their objective (8), while we allow off-policy evaluation to update the policy, which should be more sample-efficient.

A.3. Proof and discussion of Theorem 2 — the dual function's convexity and the condition of strong convexity

Note the dual function is the supremum of Lagrangian function w.r.t q:

$$g(\eta,\lambda) = \max_q L(q,\lambda,\eta,\kappa), \tag{31}$$

where κ can be calculated by Eq. (27) given q. While the convexity of dual function g has been proven in previous literature (Proposition 1, section 8.3 in (Luenberger, 1997)), we further derive the conditions of strong convexity.

Recall the dual function (we omit θ to simplify the notation):

$$g(\eta, \lambda) = \lambda\epsilon_1 + \eta\epsilon_2 + \eta \int \rho_q(s) \log \int \pi(a|s) \exp\left(\frac{Q_r(s,a) - \lambda Q_c(s,a)}{\eta}\right) dads. \tag{32}$$

Observe that the outer integral is independent of η, λ, so the convexity of $g(\eta, \lambda)$ is the same as:

$$\bar{g}(\eta, \lambda) = \lambda\epsilon_1 + \eta\epsilon_2 + \eta \log \int \pi(a|s) \exp\left(\frac{Q_r(s,a) - \lambda Q_c(s,a)}{\eta}\right) da. \tag{33}$$

Remark 4. *Interestingly, both dual functions could be used in off-policy implementation by tuning the batch size hyper-parameter. We could view $g(\eta, \lambda)$ as solving the dual variables through a batch of data, while $\bar{g}(\eta, \lambda)$ is a per-state variant — batch size equals one. Namely, $\bar{g}(\eta, \lambda)$ aims to compute the optimal dual variables for each state. Practically, we prefer using the batch version because computing the dual for each state is computational expensive and sensitive to the Q-value estimation error.*

To further prove the strong convexity conditions of $\bar{g}$, we first introduce the following lemmas.

Lemma 2. *(Lagrange's Identity) Given two functions f, g such that $\int_a^b f^2(x)dx < +\infty, \int_a^b g^2(x)dx < +\infty$, the following equation holds,*

$$\int_a^b f^2(x)dx \int_a^b g^2(x)dx - \left(\int_a^b f(x)g(x)dx\right)^2 = \frac{1}{2}\int_a^b \int_a^b \left(f(x)g(y) - g(x)f(y)\right)^2 dxdy. \tag{34}$$

Proof.

$$\begin{aligned}
\text{LHS} &= \int_a^b \int_a^b f^2(x)g^2(y)dxdy - \int_a^b f(x)g(x)dx \int_a^b f(y)g(y)dy && (35)\\
&= \frac{1}{2}\int_a^b \int_a^b f^2(x)g^2(y)dxdy + \frac{1}{2}\int_a^b \int_a^b f^2(y)g^2(x)dxdy - \int_a^b \int_a^b f(x)g(x)f(y)g(y)dxdy && (36)\\
&= \frac{1}{2}\int_a^b \int_a^b \left(f(x)g(y) - g(x)f(y)\right)^2 dxdy = \text{RHS}. && (37)
\end{aligned}$$

□

According to the convexity of dual function, the Hessian matrix of $\bar{g}(\lambda, \eta)$ is always positive semi-definite. Extending to this generic property, the following lemma gives the conditions of strict positive definiteness of Hessian matrix in constrained RL problem.

Lemma 3. *The Hessian matrix $\mathbf{H}$ of $\bar{g}(\lambda, \eta)$ is strictly positive definite when (1) $Q_r^{\theta_i}(s, \cdot), Q_c^{\theta_i}(s, \cdot)$ are not constant functions; (2) $\forall C \in \mathbb{R}, \exists a_0, s.t. Q_r^{\theta_i}(s, a_0) \neq C \cdot Q_c^{\theta_i}(s, a_0)$; and (3) $\lambda^* < +\infty$, where λ^* is the optimal dual variables in Eq. (30).*

Proof. For simplicity, let $M(a) \doteq \pi(a|s) \exp\left(\frac{Q_r(s,a) - \lambda Q_c(s,a)}{\eta}\right)$, then $\bar{g}(\eta, \lambda) = \lambda\epsilon_1 + \eta\epsilon_2 + \eta \log \int M(a)da$,

$$\begin{cases}
\dfrac{\partial M(a)}{\partial \lambda} &= M(a)\left(-\dfrac{Q_c(s,a)}{\eta}\right)\\
\dfrac{\partial M(a)}{\partial \eta} &= M(a)\left(-\dfrac{Q_r(s,a) - \lambda Q_c(s,a)}{\eta^2}\right)
\end{cases} \tag{38}$$

$$\Rightarrow \begin{cases}
\dfrac{\partial \bar{g}(\lambda, \eta)}{\partial \lambda} &= \epsilon_1 - \dfrac{\int M(a)Q_c(s,a)da}{\int M(a)da}\\
\dfrac{\partial \bar{g}(\lambda, \eta)}{\partial \eta} &= \epsilon_2 + \log \int M(a)da - \dfrac{\int M(a)[Q_r(s,a) - \lambda Q_c(s,a)]da}{\eta \int M(a)da}
\end{cases} \tag{39}$$

$$\Rightarrow \begin{cases} \mathbf{H}_{1,1} = \dfrac{\partial^2 \bar{g}(\lambda,\eta)}{\partial \lambda^2} & = \dfrac{\int M(a)(Q_c(s,a))^2 da \int M(a) da - \left(\int M(a) Q_c(s,a) da\right)^2}{\eta (\int M(a) da)^2} \\ \mathbf{H}_{2,2} = \dfrac{\partial^2 \bar{g}(\lambda,\eta)}{\partial \eta^2} & = \dfrac{\int M(a)[Q_r(s,a) - \lambda Q_c(s,a)]^2 da \int M(a) da - \left(\int M(a)[Q_r(s,a) - \lambda Q_c(s,a)] da\right)^2}{\eta^3 (\int M(a) da)^2} \\ \mathbf{H}_{1,2} = \dfrac{\partial^2 \bar{g}(\lambda,\eta)}{\partial \lambda \partial \eta} & = \mathbf{H}_{2,1} = \dfrac{\partial^2 \bar{g}(\lambda,\eta)}{\partial \eta \partial \lambda} = \dfrac{\int M(a) da \int M(a)[Q_r(s,a) - \lambda Q_c(s,a)] Q_c(s,a) da}{\eta^2 (\int M(a) da)^2} \\ & \quad - \dfrac{\int M(a) Q_c(s,a) da \int M(a)[Q_r(s,a) - \lambda Q_c(s,a)] da}{\eta^2 (\int M(a) da)^2}. \end{cases} \tag{40}$$

Therefore, $\mathbf{H}$ is positive semi-definite with following two conditions,

$$\mathbf{H}_{1,1} \geq 0, \quad \mathbf{H}_{1,1}\mathbf{H}_{2,2} - \mathbf{H}_{1,2}^2 \geq 0. \tag{41}$$

Note $\mathbf{H}$ is strictly positive definite when the equality does not hold. We will first prove the inequality holds and further discuss the condition of strict positive definiteness later.

By Cauchy–Schwarz inequality,

$$\int M(a)(Q_c(s,a))^2 da \int M(a) da \geq \left(\int M(a) Q_c(s,a) da\right)^2 \Rightarrow \mathbf{H}_{1,1} \geq 0. \tag{42}$$

i.e., the first condition in Eq. (41) holds. To prove the second condition, we use shorthand $M = M(a), Q_r = Q_r(s,a), Q_c = Q_c(s,a)$ for simplicity,

$$\begin{aligned} & \mathbf{H}_{1,1}\mathbf{H}_{2,2} - \mathbf{H}_{1,2}^2 \geq 0 \\ \Leftrightarrow & \left[\int M Q_c^2 da \int M da - \left(\int M Q_c da\right)^2\right] \left[\int M (Q_r - \lambda Q_c)^2 da \int M da - \left(\int M (Q_r - \lambda Q_c) da\right)^2\right] \geq \\ & \left[\int M Q_c da \int M (Q_r - \lambda Q_c) da - \int M da \int M (Q_r - \lambda Q_c) Q_c da\right]^2. \end{aligned} \tag{43}$$

By lemma 2, we have

$$\int M Q_c^2 da \int M da - \left(\int M Q_c da\right)^2 \tag{44}$$

$$= \frac{1}{2} \iint \left(\sqrt{M(a_1)} Q_c(s,a_1) \sqrt{M(a_2)} - \sqrt{M(a_2)} Q_c(s,a_2) \sqrt{M(a_1)}\right)^2 da_1 da_2 \tag{45}$$

$$= \frac{1}{2} \iint M(a_1) M(a_2) \left(Q_c(s,a_1) - Q_c(s,a_2)\right)^2 da_1 da_2, \tag{46}$$

and

$$\int M (Q_r - \lambda Q_c)^2 da \int M da - \left(\int M (Q_r - \lambda Q_c) da\right)^2 \tag{47}$$

$$= \frac{1}{2} \iint \left(\sqrt{M(a_1)} (Q_r(s,a_1) - \lambda Q_c(s,a_1)) \sqrt{M(a_2)} - \sqrt{M(a_2)} (Q_r(s,a_2) - \lambda Q_c(s,a_2)) \sqrt{M(a_1)}\right)^2 da_1 da_2 \tag{48}$$

$$= \frac{1}{2} \iint M(a_1) M(a_2) \Big[(Q_r(s,a_1) - \lambda Q_c(s,a_1)) - (Q_r(s,a_2) - \lambda Q_c(s,a_2))\Big]^2 da_1 da_2. \tag{49}$$

Therefore, apply Cauchy–Schwarz inequality to the LHS of Eq. (43), we have

$$\text{LHS} = \frac{1}{2}\iint M(a_1)M(a_2)\left(Q_c(s,a_1)-Q_c(s,a_2)\right)^2 da_1 da_2 \tag{50}$$

$$\cdot\frac{1}{2}\iint M(a_1)M(a_2)\Big[(Q_r(s,a_1)-\lambda Q_c(s,a_1))-(Q_r(s,a_2)-\lambda Q_c(s,a_2))\Big]^2 da_1 da_2 \tag{51}$$

$$\geq \frac{1}{4}\Bigg[\iint M(a_1)M(a_2)(Q_c(s,a_1)-Q_c(s,a_2)) \tag{52}$$

$$\cdot\left((Q_r(s,a_1)-\lambda Q_c(s,a_1))-(Q_r(s,a_2)-\lambda Q_c(s,a_2))\right)da_1 da_2\Bigg]^2, \text{(by Cauchy–Schwarz inequality)} \tag{53}$$

$$=\frac{1}{4}\Big[\iint M(a_1)M(a_2)\big[Q_c(s,a_1)(Q_r(s,a_1)-\lambda Q_c(s,a_1))-Q_c(s,a_2)(Q_r(s,a_1)-\lambda Q_c(s,a_1)) \tag{54}$$

$$-Q_c(s,a_1)(Q_r(s,a_2)-\lambda Q_c(s,a_2))+Q_c(s,a_2)(Q_r(s,a_2)-\lambda Q_c(s,a_2))\big]da_1 da_2\Big]^2 \tag{55}$$

$$=\frac{1}{4}\left[2\iint M(a_1)M(a_2)\left(Q_c(s,a_1)(Q_r(s,a_1)-\lambda Q_c(s,a_1))-Q_c(s,a_1)(Q_r(s,a_2)-\lambda Q_c(s,a_2))\right)da_1 da_2\right]^2 \tag{56}$$

$$=\Bigg[\int M(a_1)Q_c(s,a_1)(Q_r(s,a_1)-\lambda Q_c(s,a_1))da_1\int M(a_2)da_2 \tag{57}$$

$$-\int M(a_1)Q_c(s,a_1)da_1\int M(a_2)(Q_r(s,a_2)-\lambda Q_c(s,a_2))da_2\Bigg]^2 = \text{RHS}. \tag{58}$$

Note that we use the Cauchy-Schwarz inequality in a middle step. We could easily check that when Q_c is a constant function or $\lambda\to+\infty$, then the equality holds. Otherwise, the equality holds if and only if:

$$\frac{(Q_r(s,a_1)-\lambda Q_c(s,a_1))-(Q_r(s,a_2)-\lambda Q_c(s,a_2))}{Q_c(s,a_1)-Q_c(s,a_2)}=\frac{Q_r(s,a_1)-Q_r(s,a_2)}{Q_c(s,a_1)-Q_c(s,a_2)}-\lambda=C, \tag{59}$$

where C is a constant. It could be achieved by setting Q_r be a constant function or let Q_r be proportional to Q_c. We summarize the above conditions as follows: (1) $Q_r(s,\cdot)$ or $Q_c(s,\cdot)$ is a constant function; or (2) $\exists C\in\mathbb{R}, s.t.\forall a, Q_r(s,a)=CQ_c(s,a)$; or (3) $\lambda\to+\infty$. Each condition has corresponding meaning, as we explain in the following remarks. □

Remark 5. *For the first condition, if $Q_r(s,\cdot)$ is a constant function, then the objective in (7) will always be a constant — the optimization becomes meaningless since all the distributions that within the trust region should be the same; if $Q_c(s,\cdot)$ is a constant function, then the safety constraint will be inactive, since no policy could change the feasibility status. Note that the Hessian element $\mathbf{H}_{1,1}$ will be 0 if $Q_c(s,\cdot)$ is a constant, which means that the gradient of the safety constraint dual variable λ will either be a positive constant when the problem is feasible (tends to make $\lambda\to 0$) or a negative constant when the problem is infeasible (tends to make $\lambda\to+\infty$).*

Remark 6. *The second condition indicates that if the task reward Q_r value is proportional to the cost Q_c value everywhere, then multiple optimal solutions may exist on the KL constraint boundary or the safety constraint boundary. Intuitively, the problem becomes a bounded convex optimization — maximizing the risks $\mathbb{E}_q[Q_c(s,a)]$ within the trust region defined by ϵ_2 and a ball defined by ϵ_1. Though the solutions may not be unique, the optimality could still be guaranteed as long as the Slater's condition 1 holds.*

Remark 7. *The third condition is equivalent to the violation of our Slater's condition assumption 1 — no distribution exists within the trust region that satisfies the safety constraints. In that sense, the gradient of the dual variable λ will always be negative, and thus $\lambda\to+\infty$, which tends to impose huge penalties on safety critics Q_c. In practice, this rarely happens since we could select a large KL threshold for E-step and choose a much smaller trust region size for M-step, such that n-step robustness could be guaranteed and the Slater's condition could be easily met, as we show in Appendix A.7.*

In summary, $\bar{g}(\lambda,\eta)$ is convex by Lemma 3 and strictly convex when the equality does not hold in Eq. (59). The strict convexity analysis ensures the uniqueness of the optimal solution in most situations and provides some other theoretical implications of the E-M procedure, such as the convergence to a stationary point (Sriperumbudur et al., 2009). In addition, as long as the Slater's condition in Assumption 1 holds, we could obtain the optimal solution of the dual problem efficiently via any convex optimization techniques, though multiple optimal solutions may exist if Eq. (59) holds.

A.4. Proof of the M-step — KL regularized policy improvement

As shown in section 3.1 and (Abdolmaleki et al., 2018b), given the optimal variational distribution q_i^* from the E-step, the M-step objective is

$$\theta_{i+1} = \arg\max_{\theta} \mathbb{E}_{\rho_q}\Big[\alpha \mathbb{E}_{q_i^*(\cdot|s)}\big[\log \pi_\theta(a|s)\big]\Big] + \log p(\theta) \tag{60}$$

which is a Maximum A-Posteriori (MAP) problem. Consider a Gaussian prior around the old policy parameter θ_i, we have

$$\theta \sim \mathcal{N}(\theta_i, \frac{F_{\theta_i}}{\alpha\beta})$$

where F_{θ_i} is the Fisher information matrix and β is a positive constant. With the Gaussian prior, the objective (60) becomes

$$\begin{aligned}\theta_{i+1} &= \arg\max_{\theta} \alpha \mathbb{E}_{\rho_q}\Big[\mathbb{E}_{q_i^*(\cdot|s)}\big[\log \pi_\theta(a|s)\big]\Big] - \alpha\beta(\theta-\theta_i)^T F_{\theta_i}^{-1}(\theta-\theta_i) \\ &= \arg\max_{\theta} \mathbb{E}_{\rho_q}\Big[\mathbb{E}_{q_i^*(\cdot|s)}\big[\log \pi_\theta(a|s)\big]\Big] - \beta(\theta-\theta_i)^T F_{\theta_i}^{-1}(\theta-\theta_i)\end{aligned} \tag{61}$$

where we could observe that $(\theta-\theta_i)^T F_{\theta_i}^{-1}(\theta-\theta_i)$ is the second order Taylor expansion of $\mathbb{E}_{\rho_q}\big[D_{\mathrm{KL}}(\pi_{\theta_i}(a|s)\|\pi_\theta(a|s))\big]$. Thus, we could generalize the above objective to the KL-regularized one:

$$\max_{\theta} \mathbb{E}_{\rho_q}\Big[\mathbb{E}_{q_i^*(\cdot|s)}\big[\log \pi_\theta(a|s)\big] - \beta D_{\mathrm{KL}}(\pi_{\theta_i}\|\pi_\theta)\Big]. \tag{62}$$

Similar to the E-step, we could convert the soft KL regularizer to a hard KL constraint:

$$\begin{aligned}\max_{\theta} \quad & \mathbb{E}_{\rho_q}\Big[\mathbb{E}_{q_i^*(\cdot|s)}\big[\log \pi_\theta(a|s)\big]\Big] \\ s.t. \quad & \mathbb{E}_{\rho_q}\big[D_{\mathrm{KL}}(\pi_{\theta_i}(a|s)\|\pi_\theta(a|s))\big] \le \epsilon.\end{aligned} \tag{63}$$

A.5. Proof of Proposition 1 — the ELBO improvement guarantee

Recall that the optimality of policy π is lower bounded by the following ELBO objective

$$\mathcal{J}(q,\theta) = \mathbb{E}_{\tau\sim q}\left[\sum_{t=0}^{\infty}\big(\gamma^t r_t - \alpha D_{\mathrm{KL}}(q(\cdot|s_t)\|\pi_\theta(\cdot|s_t))\big)\right] + \log p(\theta).$$

We improve the policy by optimizing the ELBO alternatively via EM. Thus, we will prove the monotonic improvement guarantee of ELBO at the i-th training iteration with the following assumption.

Assumption 2. *The Slater's condition holds for both $\pi_{\theta_{i-1}}$ and π_{θ_i}.*

The above assumption indicates a well-optimized policy in the M-step. In addition, it ensures the variational distributions q_{i-1} and q_i are feasible. Note that an infeasible variational distribution q_{i-1} may lead to arbitrarily high reward return. With this assumption, we prove the ELBO improvement for E-step and M-step separately.

Proof. **E-step**: By the definition of E-step, we improve the ELBO w.r.t q. Since q_{i-1} is feasible (reward return is bounded) and π_{θ_i} satisfies the Slater's condition, we can prove the E-step update will increase ELBO by Theorem 1:

$$\begin{cases} q_i = \arg\max\limits_{q\in\Pi_{\mathcal{Q}}^{\epsilon_1}} \mathbb{E}_{\rho_q}\left[\mathbb{E}_{a\sim q(\cdot|s)}\left[Q_r^{q_i}(s,a)\right] - \alpha D_{\mathrm{KL}}[q(\cdot|s)\|\pi_{\theta_i}(\cdot|s)]\right] & \text{(By Slater's condition for } \pi_{\theta_i}) \\ \quad = \arg\max\limits_{q\in\Pi_{\mathcal{Q}}^{\epsilon_1}} \mathbb{E}_{\tau\sim q}\left[\sum\limits_{t=0}^{\infty}\big(\gamma^t r_t - \alpha D_{\mathrm{KL}}(q(\cdot|s_t)\|\pi_\theta(\cdot|s_t))\big)\right] \\ \quad = \arg\max\limits_{q\in\Pi_{\mathcal{Q}}^{\epsilon_1}} \mathcal{J}(q,\theta_i) \\ q_{i-1} \in \Pi_{\mathcal{Q}}^{\epsilon_1} \quad \text{(By Slater's condition for } \pi_{\theta_{i-1}}) \end{cases}$$

$$\Rightarrow \mathcal{J}(q_i,\theta_i) \ge \mathcal{J}(q_{i-1},\theta_i).$$

Therefore, as long as assumption 2 holds, the ELBO will increase monotonically in terms of q.

M-step: By definition in Eq. (10), we update θ by

$$\begin{aligned}\theta_{i+1} &= \arg\max_{\theta} \mathbb{E}_{\rho_{q_i}}\left[\alpha \mathbb{E}_{a\sim q_i(\cdot|s)}\left[\log \pi_\theta(a|s)\right]\right] + \log p(\theta) \\ &= \arg\max_{\theta} \mathbb{E}_{\rho_{q_i}}\left[-\alpha D_{\mathrm{KL}}[q_i(\cdot|s_t)\|\pi_\theta(\cdot|s)]\right] + \log p(\theta) \\ &= \arg\max_{\theta} \mathbb{E}_{\rho_{q_i}}\left[\mathbb{E}_{a\sim q_i(\cdot|s)}\left[Q_r^{q_i}(s,a)\right] - \alpha D_{\mathrm{KL}}[q_i(\cdot|s_t)\|\pi_\theta(\cdot|s)]\right] + \log p(\theta) \\ &= \arg\max_{\theta} \mathcal{J}(q_i, \pi_\theta).\end{aligned}$$

Therefore, we have: $\mathcal{J}(q_i, \pi_{\theta_{i+1}}) \geq \mathcal{J}(q_i, \pi_{\theta_i})$. Combining all the above together, we have

$$\mathcal{J}(q_i, \pi_{\theta_{i+1}}) \geq \mathcal{J}(q_i, \pi_{\theta_i}) \geq \mathcal{J}(q_{i-1}, \pi_{\theta_i}).$$

□

Remark 8. *The Slater's condition assumption is critical for monotonic policy improvement guarantee, since as we have shown in Appendix A.2 Remark 4, violation of Slater's condition may lead to large penalty for constraint violations, and then in E-step the variational distribution is updated to reduce the cost or reconcile to the feasible set instead of improving the reward return. However, the n-step robustness (Appendix A.7) guarantees the updated policies in n consecutive iterations satisfy the Slater's condition if we choose proper KL constraint thresholds and the initial policy is feasible, which indicates a monotonic ELBO improvement guarantee during these n policy updating iterations.*

A.6. Proof of Proposition 2 — worst-case constraint satisfaction bound

The Corollary 2 and 3 in CPO (Achiam et al., 2017) connect the difference in cost returns between two policies to the divergence between them,

$$J_c(\pi') - J_c(\pi) \leq \frac{1}{1-\gamma}\mathbb{E}_{s\sim\rho_\pi, a\sim\pi'}[A_c^\pi(s,a)] + \frac{2\gamma\delta_c^{\pi'}}{(1-\gamma)^2}\sqrt{\frac{1}{2}\mathbb{E}_{s\sim\rho_\pi}\left[D_{\mathrm{KL}}[\pi'(\cdot|s)\|\pi(\cdot|s)]\right]} \tag{64}$$

where $\delta_c^{\pi'} = \max_s \left|\mathbb{E}_{a\sim\pi'}[A_c^{\theta_i}(s,a)]\right|$, and $A_c^{\theta_i}(s,a)$ denotes the advantage function of cost.

Note that the M-step does not involve cost constraint when updating θ. Therefore, we obtain the following worse-case bound:

$$\begin{aligned}J_c(\pi_{\theta_{i+1}}) &\leq J_c(\pi_{\theta_i}) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim\rho_{\pi_{\theta_i}}, a\sim\pi_{\theta_{i+1}}}[A_c^{\pi_{\theta_i}}(s,a)] + \frac{2\gamma\delta_c^{\pi_{\theta_{i+1}}}}{(1-\gamma)^2}\sqrt{\frac{1}{2}\mathbb{E}_{s\sim\rho_{\pi_{\theta_i}}}\left[D_{\mathrm{KL}}[\pi_{\theta_{i+1}}(\cdot|s)\|\pi_{\theta_i}(\cdot|s)]\right]} \\ &\leq J_c(\pi_{\theta_i}) + \frac{1}{1-\gamma}\delta_c^{\pi_{\theta_{i+1}}} + \frac{2\gamma\delta_c^{\pi_{\theta_{i+1}}}}{(1-\gamma)^2}\sqrt{\frac{1}{2}\epsilon} \\ &= J_c(\pi_{\theta_i}) + \frac{[(1-\gamma)+\sqrt{2\epsilon}\gamma]\delta_c^{\pi_{\theta_{i+1}}}}{(1-\gamma)^2}.\end{aligned} \tag{65}$$

When the old policy $\pi_{\theta_i} \in \Pi_{\mathcal{Q}}^{\epsilon_1}$, we further have

$$J_c(\pi_{\theta_{i+1}}) \leq \epsilon_1 + \frac{[(1-\gamma)+\sqrt{2\epsilon}\gamma]\delta_c^{\pi_{\theta_{i+1}}}}{(1-\gamma)^2}. \tag{66}$$

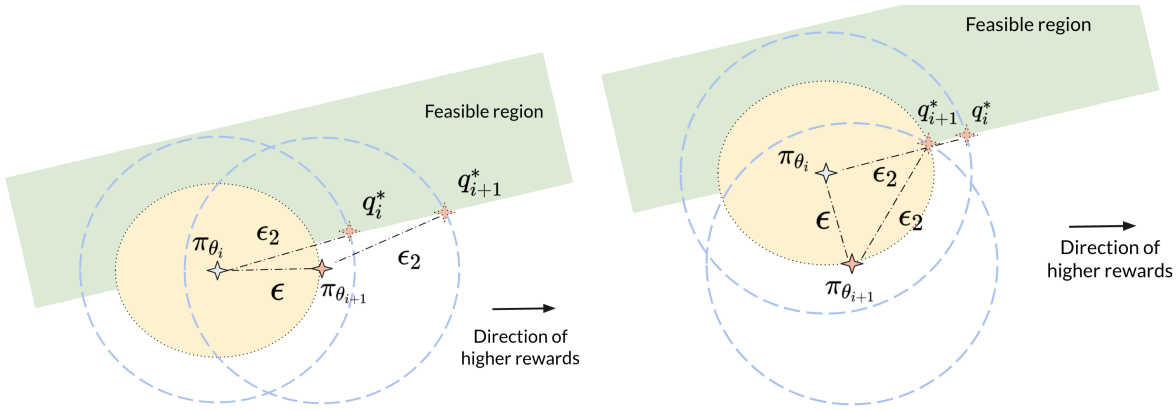
A.7. Proof and discussion of Proposition 3 — policy updating robustness

Recall the Proposition 3 — Suppose $\pi_{\theta_i} \in \Pi_{\mathcal{Q}}^{\epsilon_1}$. $\pi_{\theta_{i+1}}$ and π_{θ_i} are related by the M-step. If $\epsilon < \epsilon_2$, where ϵ, ϵ_2 are the KL threshold in M-step and E-step respectively, then the variational distribution q_{i+1}^* in the next iteration is guaranteed to be feasible and optimal.

Proof. Since $\epsilon < \epsilon_2$, the KL divergence between $\pi_{\theta_{i+1}}$ and π_{θ_i} $D_{\mathrm{KL}}(\pi_{\theta_i} \| \pi_{\theta_{i+1}}) \leq \epsilon < \epsilon_2$. Thus, the Slater condition 1 holds for $\pi_{\theta_{i+1}}$ as long as π_{θ_i} is feasible, because at least one feasible solution π_{θ_i} within the trust region exists. By Theorem 1, we know that q^*_{i+1} in the E-step is guaranteed to be feasible and optimal. □

Remark 9. *Proposition 3 is useful in practice, since it allows the policy to recover to the feasible region from perturbations in M-step. The perturbations may come from bad function approximation errors in M-step or noisy Q estimations, which may happen occasionally for black-box function approximators such as neural network. The robustness guarantee holds even under the worst-case scenario with adversarial perturbations.*

Figure illustrations. Fig. 7(a) demonstrates the example of one EM iteration that is subject to approximation errors in the M-step. The green area represents the feasible region in the parameter space, the yellow ellipsoid is the trust region in the M-step, and dashed blue circles are the trust region size in the E-step. The righter region has the higher reward in this counter example. Ideally, the updated policy should be the intersection point of the $\pi_{\theta_i} - q^*_i$ line and the yellow ellipsoid. However, due to the approximation errors in the M-step, the updated policy $\pi_{\theta_{i+1}}$ might be away from the correction updating direction. Fig. 7(b) shows the worst-case scenario, where the updating direction is totally orthogonal to the correct one. However, due to the smaller trust region size of M-step than E-step, the Slater condition still holds – a feasible policy exist within the trust region of $\pi_{\theta_{i+1}}$ that is specified by ϵ_2. So, we could guarantee to obtain an optimal and feasible non-parametric variational distribution at the $(i+1)$-th iteration – the policy still has the chance to recover to the feasible region.



(a) Regular policy updating (b) Worst-case policy updating

Figure 7. Illustration of the policy updating at the i-th iteration under the M-step approximation error.

Extending the one-step robustness guarantee to multiple steps. A natural follow-up question for Proposition 3 is that whether we can achieve multiple steps policy updating robustness guarantees – can the policy recover to feasible region with $n > 1$ steps adversarial/worst-case M-step policy updating? Since the n-step ($n > 1$) guarantees may require the Triangular inequality to be satisfied for consecutive updated policies, but it may not hold for KL divergence in general cases. However, we found that if the policy π_θ is of a multivariate Gaussian form, which is commonly used for continuous action space tasks in practice, then a relaxed Triangular inequality holds – see Theorem 4 in (Zhang et al., 2021). Thus, we hypothesize that with a Gaussian policy, n-step robustness could be achieved with sufficiently small $\epsilon = f(\epsilon_2, n)$ with the function f. Next, we provide a proof for two steps robustness.

Proposition 4. *Suppose $\pi_{\theta_i} \in \Pi^{\epsilon_1}_{\mathcal{Q}}$ and π is a Multivariate Gaussian policy. If $\epsilon < \frac{\epsilon_2}{8}$, where ϵ, ϵ_2 are the KL threshold in M-step and E-step respectively, then the variational distribution q^*_{i+2} in the $(i+2)$-th iteration is guaranteed to be feasible and optimal.*

Proof. Denote W_0 and W_{-1} be the 0, -1 branches of the Lambert W function, respectively. We use π_i denote π_{θ_i} for simplicity. Since $D_{\mathrm{KL}}(\pi_i \| \pi_{i+1}) \leq \epsilon$ and $D_{\mathrm{KL}}(\pi_{i+1} \| \pi_{i+2}) \leq \epsilon$, we have the following relaxed Triangular inequality holds (Theorem 4 in (Zhang et al., 2021)):

$$D_{\mathrm{KL}}(\pi_i \| \pi_{i+2}) < 2\epsilon + \frac{1}{2}\left(\left(W_{-1}(-e^{(-1-2\epsilon)}) + 1 \right)^2 - W_{-1}(-e^{(-1-2\epsilon)}) \left(\sqrt{\frac{2\epsilon}{-W_0(-e^{(-1-2\epsilon)})}} + \sqrt{2\epsilon} \right)^2 \right). \quad (67)$$

Note that $W_0(-1/e) = W_{-1}(-1/e) = -1$ and for sufficiently small ϵ, $W_{-1}(-e^{(-1-2\epsilon)})$ and $W_0(-e^{(-1-2\epsilon)})$ are arbitrarily close to -1 by the following series (Corless et al., 1996)

$$W_{-1}(-e^{(-1-2\epsilon)}) = -1 - 2\sqrt{\epsilon} + O(\epsilon); \quad W_0(-e^{(-1-2\epsilon)}) = -1 + 2\sqrt{\epsilon} - O(\epsilon). \tag{68}$$

So we have

$$\left(W_{-1}(-e^{(-1-2\epsilon)}) + 1\right)^2 = (-2\sqrt{\epsilon} + O(\epsilon))^2 = 4\epsilon - O(\epsilon^{1.5}) \tag{69}$$

$$-W_{-1}(-e^{(-1-2\epsilon)}) \left(\sqrt{\frac{2\epsilon}{-W_0(-e^{(-1-2\epsilon)})}} + \sqrt{2\epsilon}\right)^2 = (1 + 2\sqrt{\epsilon} + O(\epsilon)) \left(\sqrt{\frac{2\epsilon}{1 - 2\sqrt{\epsilon} + O(\epsilon)}} + \sqrt{2\epsilon}\right)^2 \tag{70}$$

$$\leq (1 + 2\sqrt{\epsilon} + O(\epsilon)) \left(\frac{4\epsilon}{1 - 2\sqrt{\epsilon} + O(\epsilon)} + 4\epsilon\right) \tag{71}$$

$$= 8\epsilon + O(\epsilon^1) + O(\epsilon^{1.5}). \tag{72}$$

Then we obtain the following bound by ignoring the high order terms for sufficiently small ϵ:

$$D_{\mathrm{KL}}(\pi_i \| \pi_{i+2}) < 2\epsilon + \frac{1}{2}(4\epsilon + 8\epsilon) = 8\epsilon. \tag{73}$$

Therefore, we could conclude that for sufficiently small trust-region size, and $\epsilon < \frac{\epsilon_2}{8}$, we could obtain two steps robustness guarantee – no matter how worse are the M-step for two iterations, an optimal and feasible variational distribution could be solved, and the policy could be recovered to the safe region. □

While the above proof could be generalized to n-step robustness, we found that as the certified step n increases, the trust region size ϵ in M-step has to be shrunk with a faster rate than n. In addition, we could observe that though smaller trust region size in M-step could improve the robustness, it will also reduce the training efficiency – more training steps and samples are required to improve the policy. Therefore, there exists a natural robustness and efficiency trade-off in this context. As we have shown in the experiment section, we believe that one or two steps robustness should be enough to handle mild approximation errors in practice, since the MLE style objective in M-step will converge to the true distribution in probability (consistency) and achieve the lowest-possible variance of parameters (efficiency) asymptotically as the sample size increases (Myung, 2003). This principle also guides us to select reasonable batch size and particle size practically.

B. Implementation Details

B.1. Full algorithm

Due to the page limit, we omit some implementation details in the main content. We will present the full algorithm and some implementation tricks in this section. Without otherwise statement, the critics' and policies' parametrization is assumed to be neural networks (NN), while we believe other parametrization form should also work well.

Critics update. Denote ϕ_r as the parameters for the task reward critic Q_r, and ϕ_c as the parameters for the constraint violation cost critic Q_c. Similar to many other off-policy algorithms (Lillicrap et al., 2015), we use a target network for each critic and the polyak smoothing trick to stabilize the training. Other off-policy critics training methods, such as Re-trace (Munos et al., 2016), could also be easily incorporated with CVPO training framework. Denote ϕ_r' as the parameters for the **target** reward critic Q_r', and ϕ_c' as the parameters for the **target** cost critic Q_c'. Define $\mathcal{D}$ as the replay buffer and (s, a, s', r, c) as the state, action, next state, reward, and cost respectively. The critics are updated by minimizing the following mean-squared Bellman error (MSBE):

$$L(\phi_r) = \mathbb{E}_{(s,a,s',r,c)\sim\mathcal{D}}\Big[\left(Q_r(s,a) - (r + \gamma \mathbb{E}_{a'\sim\pi}[Q_r'(s',a')])\right)^2\Big] \tag{74}$$

$$L(\phi_c) = \mathbb{E}_{(s,a,s',r,c)\sim\mathcal{D}}\Big[\left(Q_c(s,a) - (c + \gamma \mathbb{E}_{a'\sim\pi}[Q_c'(s',a')])\right)^2\Big]. \tag{75}$$

Denote α_c as the critics' learning rate, we have the following updating equations:

$$\phi_r \leftarrow \phi_r - \alpha_c \nabla_{\phi_r} L(\phi_r) \tag{76}$$

$$\phi_c \leftarrow \phi_c - \alpha_c \nabla_{\phi_c} L(\phi_c). \tag{77}$$

M-step regularized policy improvement trick. We use a Multivariate Gaussian policy in our implementation. As shown in MPO (Abdolmaleki et al., 2018b;a), decoupling the KL constraint into two separate terms – mean ϵ_μ and covariance ϵ_Σ, can yield better empirical performance. Denote $C_\mu = \mathbb{E}_{\rho_q}\big[\frac{1}{2}\mathbf{tr}(\Sigma^{-1}\Sigma_i) - n + \ln(\frac{\Sigma}{\Sigma_i})\big]$ and $C_\mu = \mathbb{E}_{\rho_q}\big[\frac{1}{2}(\mu-\mu_i)^T\Sigma^{-1}(\mu-\mu_i)\big]$, we have:

$$\mathbb{E}_{\rho_q}\big[D_{\mathrm{KL}}(\pi_{\theta_i}(a|s)\|\pi_\theta(a|s))\big] = C_\mu + C_\Sigma. \tag{78}$$

And thus the M-step objective could be written as the following Lagrangian function:

$$L(\theta, \beta_\mu, \beta_\Sigma) = \mathbb{E}_{\rho_q}\Big[\mathbb{E}_{q_i^*(\cdot|s)}\big[\log \pi_\theta(a|s)\big]\Big] + \beta_\mu(\epsilon_\mu - C_\mu) + \beta_\Sigma(\epsilon_\Sigma - C_\Sigma). \tag{79}$$

By performing the gradient descend ascend algorithm over the dual variables β_μ, β_Σ and the policy parameters θ in Eq. (79) iteratively yields the KL-constrained policy improvement in a supervised learning fashion:

$$\max_\theta \min_{\beta_\mu>0, \beta_\Sigma>0} \quad L(\theta, \beta_\mu, \beta_\Sigma). \tag{80}$$

Denote $\alpha_\mu, \alpha_\Sigma, \alpha_\theta$ as the learning rate for $\beta_\mu, \beta_\Sigma, \theta$ respectively, we have the following updating equations:

$$\beta_\mu \leftarrow \beta_\mu - \alpha_\mu \frac{\partial L(\theta, \beta_\mu, \beta_\Sigma)}{\partial \beta_\mu} = \beta_\mu - \alpha_\mu(\epsilon_\mu - C_\mu) \tag{81}$$

$$\beta_\Sigma \leftarrow \beta_\Sigma - \alpha_\Sigma \frac{\partial L(\theta, \beta_\mu, \beta_\Sigma)}{\partial \beta_\Sigma} = \beta_\Sigma - \alpha_\Sigma(\epsilon_\Sigma - C_\Sigma) \tag{82}$$

$$\theta \leftarrow \theta - \alpha_\theta \frac{\partial L(\theta, \beta_\mu, \beta_\Sigma)}{\partial \theta}. \tag{83}$$

In practice, we also use a target policy network $\pi_{\theta'}$ to generate the covariance matrix Σ and the current policy network π_θ to generate the mean vector μ, such that the policy will not be easily collapsed to a local optimum.

Polyak averaging for the target networks. The polyak averaging is specified by a weight parameter $\rho \in (0, 1)$ and updates the parameters with:

$$\begin{aligned} \phi_r' &= \rho\phi_r' + (1-\rho)\phi_r \\ \phi_c' &= \rho\phi_c' + (1-\rho)\phi_c \\ \theta' &= \rho\theta' + (1-\rho)\theta. \end{aligned} \tag{84}$$

With all the implementation tricks mentioned above, we present the full CVPO algorithm:

Algorithm 2 CVPO Algorithm

Input: rollouts T, M-step iteration number M, batch size B, particle size K, discount factor γ, polyak weight ρ, critics learning rate α_c, policy learning rate α_θ, M-step dual variables' learning rates $\alpha_\mu, \alpha_\Sigma$, thresholds $\epsilon_\mu, \epsilon_\Sigma$
Output: policy π_θ

Initialize policy parameters θ, θ', critics parameters $\phi_r, \phi_r', \phi_c, \phi_c'$ and replay buffer $\mathcal{D} = \{\}$
for each training iteration **do**
  Rollout T trajectories by π_θ from the environment $\mathcal{D} = \mathcal{D} \cup \{(s, a, s', r, c)\}$
  Sample B transitions $\{(s_b, a_b, s_{b+1}, r_b, c_b)_{b=1,...,B}\}$ from the replay buffer $\mathcal{D}$
  ▷ *E-step begins*
  Update reward critic by Eq. (74): $\phi_r \leftarrow \phi_r - \alpha_c \nabla_{\phi_r} L(\phi_r)$
  Update cost critic by Eq. (75): $\phi_c \leftarrow \phi_c - \alpha_c \nabla_{\phi_c} L(\phi_c)$
  for $b = 1, ..., B$ **do**
    Sample K actions $\{a_1, ..., a_K\}$ for s_b
    Compute $\{Q_r^{\theta_i}(s_b, a_k), Q_c^{\theta_i}(s_b, a_k); k = 1, ..., K\}$
  end for
  Compute optimal dual variables η^*, λ^* by solving the convex optimization problem (9)
  Compute the optimal variational distribution for each state $\{q^*(\cdot|s_b); b = 1, ..., B\}$ by Eq. (8)
  Normalize the variational distribution $\{q^*(\cdot|s_b); b = 1, ..., B\}$ for each state
  ▷ *M-step begins*
  for M-step iterations $m = 1, ..., M$ **do**
    Perform one gradient step for β_μ via Eq. (81) and for β_Σ via Eq. (82)
    Perform one gradient step for policy parameters via Eq. (83): $\theta \leftarrow \theta - \alpha_\theta \frac{\partial L(\theta, \beta_\mu, \beta_\Sigma)}{\partial \theta}$
  end for
  Polyak averaging target networks by Eq. (84)
end for

Note that for off-policy methods, we need to convert the episodic-wise constraint violation threshold to a state-wise threshold for the Q_c functions. Denote T as the episode length, the target cost limit for one episode is ϵ_T. Denote the discounting factor as γ. Then, if we assume that at each time step we have equal probability to violate the constraint, the target constraint value ϵ_1 for safety critic $Q_c^{\pi_\theta}$ could be approximated by:

$$\epsilon_1 = \epsilon_T \times \frac{1 - \gamma^T}{T(1 - \gamma)}$$

The converted threshold ϵ_1 will be used to compute the Lagrangian multipliers for the baselines, and also be used as one of the constraint threshold in the E-step of our method:

$$\int \pi(a|s) Q_c^{\pi_{\theta_i}}(s, a) \leq \epsilon_1, \quad \forall s, a$$

B.2. Experiment environments

The task environment implementations are built upon SafetyGym (based on Mujoco) (Ray et al., 2019) and its PyBullet implementation (Gronauer, 2022). We modified the original environment parameters for all the safe RL algorithms to make the training faster and save computational resources. Particularly, we increase the simulation time-step and decrease the timeout steps for each environment, such that the agent can finish the tasks with fewer steps. In addition, the Goal task in this paper is modified from the Button task, since we can then fix the layout of the goal buttons and obstacles to make the environment more deterministic. Note that the original SafetyGym implementation will random sample the layout for each episode, which greatly increase the training time and variance. The proposed CVPO implementation also works for the original SafetyGym environments, but may require different set of hyper-parameters and much longer training time. Though CVPO is sample-efficient, it is not very computational efficient, since we need to sample many particles in the E-step and solve a convex optimization problem, which is currently done on CPU via SciPy.

B.3. Hyper-parameters

The hyperparameters are shown in Table 1. More details can be found in the code.

Common Hyperparameters	
Policy network sizes	[256, 256]
Q network sizes	[256, 256]
Network activation	ReLU
Discount factor gamma γ	0.99
Polyak weight ρ:	0.995
Batch size B:	300
Rollout trajectory number T	20
Critics learning rate α_c	0.001
NN Optimizer	Adam

CVPO Hyperparameter	
Particle size K	32
M-step iterations M	6
Learning rate α_μ	1
Learning rate α_Σ	100
Learning rate α_θ	0.002
E-step KL threshold ϵ_2:	0.1
M-step KL threshold ϵ_μ:	0.001
M-step KL threshold ϵ_Σ:	0.0001
E-step solver	SLSQP

Table 1. Hyperparameters. Left: common hyperparameters for all methods. Right: hyperparameters that are specifically for CVPO.

C. Complete Experiment Results

We present all the experiment results in this section.

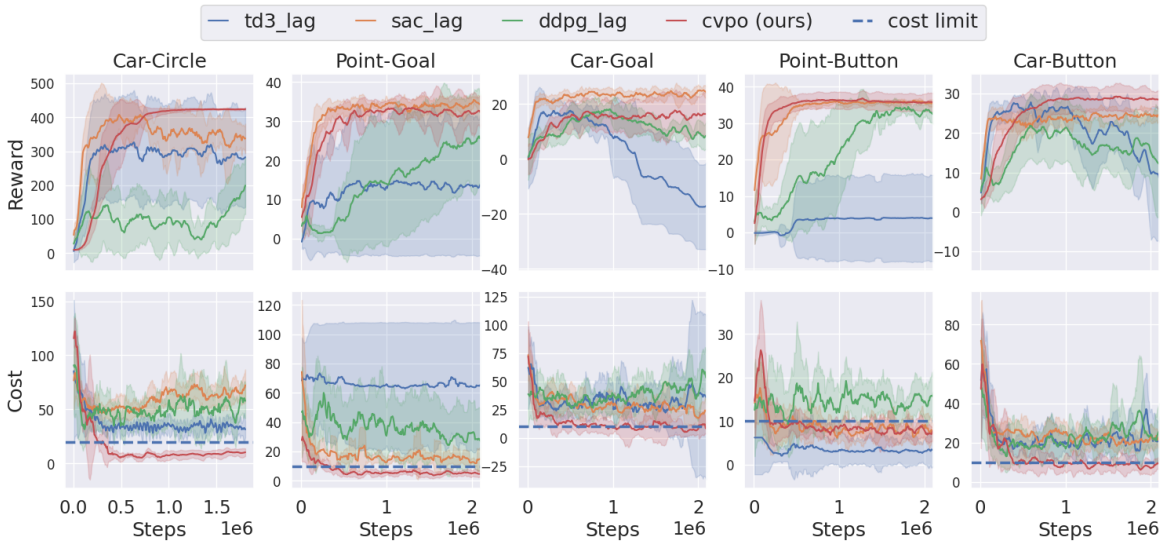


Figure 8. Training curves for off-policy baselines comparison. Each column corresponds to an environment. The curves are averaged over 10 random seeds, where the solid lines are the mean and the shadowed areas are the standard deviation.

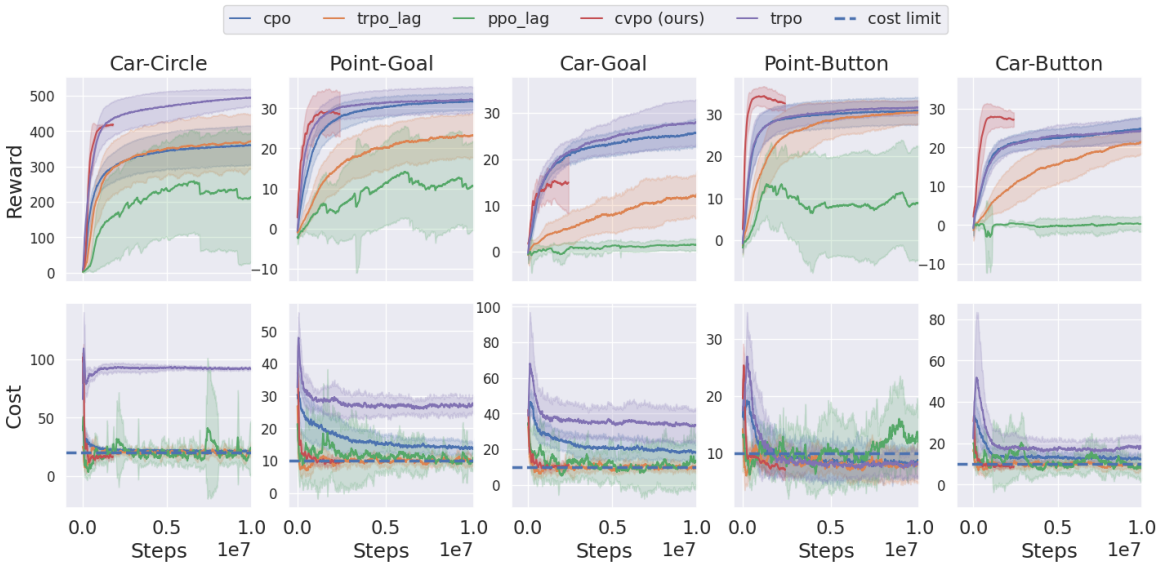


Figure 9. Training curves for on-policy baselines comparison. Each column corresponds to an environment. The curves are averaged over 10 random seeds, where the solid lines are the mean and the shadowed areas are the standard deviation.

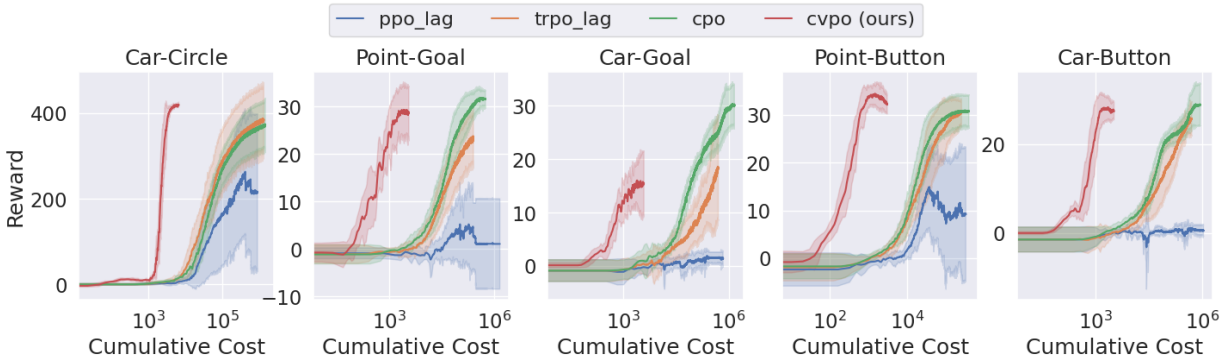
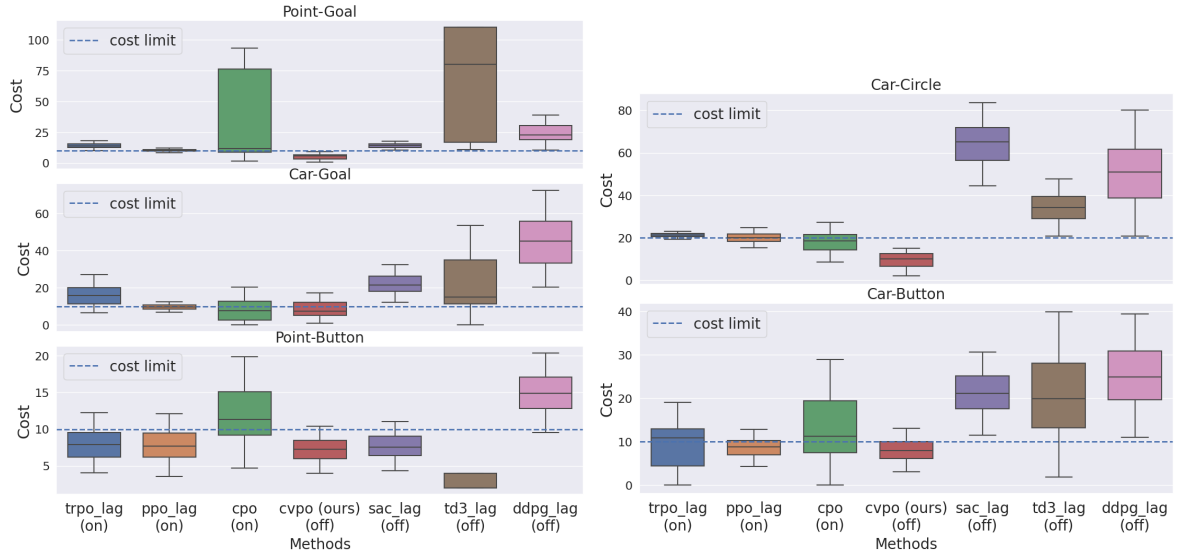


Figure 10. Reward versus cumulative cost (log-scale).



(a) (b)

Figure 11. Box plot of the convergence cost. (on) and (off) denotes on-policy and off-policy method, respectively.

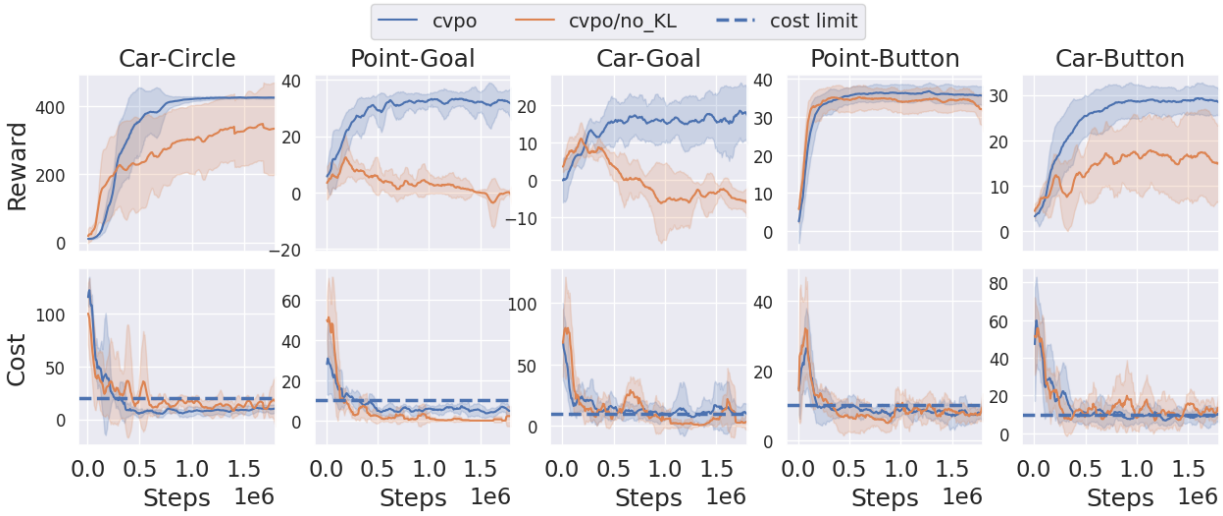


Figure 12. Ablation study of the KL constraint in the M-step.