Constrained Update Projection Approach to Safe Policy Optimization

Anonymous Author(s)
Affiliation
Address
email

Abstract

Safe reinforcement learning (RL) studies problems where an intelligent agent has to not only maximize reward but also avoid exploring unsafe areas. In this study, we propose CUP, a novel policy optimization method based on **C**onstrained **U**pdate **P**rojection framework that enjoys rigorous safety guarantee. Central to our CUP development is the newly proposed surrogate functions along with the performance bound. Compared to previous safe RL methods, CUP enjoys the benefits of 1) CUP generalizes the surrogate functions to generalized advantage estimator (GAE), leading to strong empirical performance. 2) CUP unifies performance bounds, providing a better understanding and interpretability for some existing algorithms; 3) CUP provides a non-convex implementation via only first-order optimizers, which does not require any strong approximation on the convexity of the objectives. To validate our CUP method, we compared CUP against a comprehensive list of safe RL baselines on a wide range of tasks. Experiments show the effectiveness of CUP both in terms of reward and safety constraint satisfaction.

1 Introduction

Reinforcement learning (RL) [Sutton and Barto, 1998] has achieved significant successes in many fields (e.g., [Mnih *et al.*, 2015; Silver *et al.*, 2017; OpenAI, 2019; Afsar *et al.*, 2021]). However, most RL algorithms improve the performance under the assumption that an agent is free to explore any behaviors. In real-world applications, only considering return maximization is not enough, and we also need to consider safe behaviors. For example, a robot agent should avoid playing actions that irrevocably harm its hardware, and a recommender system should avoid presenting offending items to users. Thus, it is crucial to consider *safe exploration* for RL, which is usually formulated as constrained Markov decision processes (CMDP) [Altman, 1999].

It is challenging to solve CMDP since traditional approaches (e.g., Q-learning [Watkins, 1989] & policy gradient [Williams, 1992]) usually violate the safe exploration constraints, which is undesirable for safe RL. Recently, Achiam *et al.* [2017]; Yang *et al.* [2020]; Bharadhwaj *et al.* [2021] suggest to use some surrogate functions to replace the objective and constraints. However, their implementations involve some convex approximations to the non-convex objective and safe constraints, which leads to many error sources and troubles. Concretely, Achiam *et al.* [2017]; Yang *et al.* [2020]; Bharadhwaj *et al.* [2021] approximate the non-convex objective (or constraints) with first-order or second Taylor expansion, but their implementations still lack a theory to show the error difference between the original objective (or constraints) and its convex approximations. Besides, their approaches involve the inverse of a high-dimension inverse Fisher information matrix, which causes their algorithms require a costly computation for each update when solving high-dimensional RL problems.

To address the above problems, we propose the *constrained update projection* (CUP) algorithm with a theoretical safety guarantee. We derive the CUP bases on the newly proposed surrogate functions

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

with respect to objectives and safety constraints, and provide a practical implementation of CUP that does not depend on any convex approximation to adapt high-dimensional safe RL.

Concretely, in Section 3, Theorem 1 shows generalized difference bounds between two arbitrary policies for the objective and constraints. Those bounds provide principled approximations to the objective and constraints, which are theoretical foundations for us to use those bounds as surrogate functions to replace objective and constraints to design algorithms. Although using difference bounds as surrogate functions to replace the objective has appeared in previous works (e.g., [Schulman *et al.*, 2015; Achiam *et al.*, 2017]), Theorem 1 refines those bounds (or surrogate functions) at least two aspects: **(i)** Firstly, our rigorous theoretical analysis shows a bound with respect to generalized advantage estimator (GAE) [Schulman *et al.*, 2016]. GAE significantly reduces variance empirically while maintaining a tolerable level of bias, the proposed bound involves GAE is one of the critical steps for us to design efficient algorithms. **(ii)** Our new bounds unify the classic result of CPO Achiam *et al.* [2017], i.e., the classic performance bound of CPO is a special case of our bounds. Although existing work (e.g., Zhang *et al.* [2020]; Kang *et al.* [2021]) has applied the key idea of CPO with GAE to solve safe RL problems, their approaches are all empirical and lack a theoretical analysis. Thus, the proposed newly bound partially explains the effectiveness of the above safe RL algorithms. Finally, we should emphasize that although GAE has been empirically applied to extensive reinforcement learning tasks, this work is the first to show a rigorous theoretical analysis to extend the surrogate functions with respect to GAE.

In Section 4, we provide the necessary details of the proposed CUP. The CUP contains two steps: it first performs a policy improvement, which may produce a temporary policy violates the constraint. Then in the second step, CUP projects the policy back onto the safe region to reconcile the constraint violation. Theorem 2 shows the worst-case performance degradation guarantee and approximate satisfaction of safety constraints of CUP, result shows that with a relatively small parameter that controls the penalty of the distance between the old policy and current policy, CUP shares a desirable toleration for both policy improvements and safety constraints. Furthermore, we provide a practical implementation of sample-based CUP. This implementation allows us to use deep neural networks to train a model, which is an efficient iteration without strongly convex approximation of the objective or constraints (e.g., [Achiam *et al.*, 2017; Yang *et al.*, 2020]), and it optimizes the policy according to the first-order optimizer. Finally, extensive high-dimensional experiments on continuous control tasks show the effectiveness of CUP where the agent satisfies safe constraints.

2 Preliminaries

Reinforcement learning (RL) [Sutton and Barto, 1998] is often formulated as a *Markov decision process* (MDP) [Puterman, 2014] that is a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathbb{P}, r, \rho_0, \gamma)$. Here $\mathcal{S}$ is state space, $\mathcal{A}$ is action space. $\mathbb{P}(s^{'}|s,a)$ is probability of state transition from s to $s^{'}$ after playing a. $r(\cdot) : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, and $r(s'|s,a)$ denotes the reward that the agent observes when state transition from s to $s^{'}$ after it plays a. $\rho_0(\cdot) : \mathcal{S} \to [0,1]$ is the initial state distribution and $\gamma \in (0,1)$.

A stationary parameterized policy $\pi_{\boldsymbol{\theta}}$ is a probability distribution defined on $\mathcal{S} \times \mathcal{A}$, $\pi_{\boldsymbol{\theta}}(a|s)$ denotes the probability of playing a in state s. We use $\Pi_{\boldsymbol{\theta}}$ to denote the set of all stationary policies, where $\Pi_{\boldsymbol{\theta}} = \{\pi_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \mathbb{R}^p\}$, and $\boldsymbol{\theta}$ is a parameter needed to be learned. Let $\mathbf{P}_{\pi_{\boldsymbol{\theta}}} \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$ be a state transition probability matrix, and their components are: $\mathbf{P}_{\pi_{\boldsymbol{\theta}}}[s,s'] = \sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s'|s,a) =: \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s^{'}|s)$, which denotes one-step state transformation probability from s to $s^{'}$ by executing $\pi_{\boldsymbol{\theta}}$. Let $\tau = \{s_t, a_t, r_{t+1}\}_{t\geq 0} \sim \pi_{\boldsymbol{\theta}}$ be a trajectory generated by $\pi_{\boldsymbol{\theta}}$, where $s_0 \sim \rho_0(\cdot)$, $a_t \sim \pi_{\boldsymbol{\theta}}(\cdot|s_t)$, $s_{t+1} \sim \mathbb{P}(\cdot|s_t,a_t)$, and $r_{t+1} = r(s_{t+1}|s_t,a_t)$. We use $\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s^{'}|s)$ to denote the probability of visiting the state $s^{'}$ after t time steps from the state s by executing $\pi_{\boldsymbol{\theta}}$. Due to the Markov property in MDP, $\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s^{'}|s)$ is $(s, s^{'})$-th component of the matrix $\mathbf{P}^t_{\pi_{\boldsymbol{\theta}}}$, i.e., $\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s^{'}|s) = \mathbf{P}^t_{\pi_{\boldsymbol{\theta}}}[s, s^{'}]$. Finally, let $d^{s_0}_{\pi_{\boldsymbol{\theta}}}(s) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0)$ be the stationary state distribution of the Markov chain (starting at s_0) induced by policy $\pi_{\boldsymbol{\theta}}$. We define $d^{\rho_0}_{\pi_{\boldsymbol{\theta}}}(s) = \mathbb{E}_{s_0\sim\rho_0(\cdot)}[d^{s_0}_{\pi_{\boldsymbol{\theta}}}(s)]$ as the discounted state visitation distribution on initial distribution $\rho_0(\cdot)$.

The *state value function* of $\pi_{\boldsymbol{\theta}}$ is defined as $V_{\pi_{\boldsymbol{\theta}}}(s) = \mathbb{E}_{\pi_{\boldsymbol{\theta}}}[\sum_{t=0}^{\infty}\gamma^t r_{t+1}|s_0 = s]$, where $\mathbb{E}_{\pi_{\boldsymbol{\theta}}}[\cdot|\cdot]$ denotes a conditional expectation on actions which are selected by $\pi_{\boldsymbol{\theta}}$. Its *state-action value function* is $Q_{\pi_{\boldsymbol{\theta}}}(s,a) = \mathbb{E}_{\pi_{\boldsymbol{\theta}}}[\sum_{t=0}^{\infty}\gamma^t r_{t+1}|s_0 = s, a_0 = a]$, and advantage function is $A_{\pi_{\boldsymbol{\theta}}}(s,a) = Q_{\pi_{\boldsymbol{\theta}}}(s,a) - V_{\pi_{\boldsymbol{\theta}}}(s)$. The goal of reinforcement learning is to maximize $J(\pi_{\boldsymbol{\theta}}) = \mathbb{E}_{s\sim\rho_0(\cdot)}[V_{\pi_{\boldsymbol{\theta}}}(s)]$.

2.1 Policy Gradient and Generalized Advantage Estimator (GAE)

Policy gradient [Williams, 1992; Sutton *et al.*, 2000] is widely used to solve policy optimization, which maximizes the expected total reward by repeatedly estimating the gradient $g = \nabla J(\pi_{\boldsymbol{\theta}})$. Schulman *et al.* [2016] summarize several different related expressions for the policy gradient:

$$g = \nabla J(\pi_{\boldsymbol{\theta}}) = \mathbb{E}\left[\sum_{t=0}^{\infty} \Psi_t \nabla \log \pi_{\boldsymbol{\theta}}(a_t|s_t)\right], \tag{1}$$

where Ψ_t can be total discounted reward of the trajectory, value function, advantage function or temporal difference (TD) error. As stated by Schulman *et al.* [2016], the choice $\Psi_t = A(s_t, a_t)$ yields almost the lowest possible variance, which is consistent with the theoretical analysis [Greensmith *et al.*, 2004; Wu *et al.*, 2018]. Furthermore, Schulman *et al.* [2016] propose generalized advantage estimator (GAE) $\hat{A}_t^{\texttt{GAE}(\gamma,\lambda)}(s_t, a_t)$ to replace Ψ_t: for any $\lambda \in [0, 1]$,

$$\hat{A}_t^{\texttt{GAE}(\gamma,\lambda)}(s_t, a_t) = \sum_{\ell=0}^{\infty} (\gamma\lambda)^{\ell} \delta_{t+\ell}^{V}, \tag{2}$$

where $\delta_t^V = r_{t+1} + \gamma V(s_{t+1}) - V(s_t)$ is TD error, and $V(\cdot)$ is an estimator of value function. GAE is an efficient technique for data efficiency and reliable performance of reinforcement learning.

2.2 Safe Reinforcement Learning

Safe RL is often formulated as a constrained MDP (CMDP) $\mathcal{M} \cup \mathcal{C}$ [Altman, 1999], which is a standard MDP $\mathcal{M}$ augmented with an additional constraint set $\mathcal{C}$. The set $\mathcal{C} = \{(c_i, b_i)\}_{i=1}^m$, where c_i are cost functions: $c_i : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, and limits are b_i, $i = 1, \cdot, m$. The *cost-return* is defined as: $J^{c_i}(\pi_{\boldsymbol{\theta}}) = \mathbb{E}_{\pi_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{\infty} \gamma^t c_i(s_t, a_t)\right]$, then we define the feasible policy set $\Pi_{\mathcal{C}}$ as: $\Pi_{\mathcal{C}} = \cap_{i=1}^m \left\{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}} \text{ and } J^{c_i}(\pi_{\boldsymbol{\theta}}) \leq b_i\right\}$. The goal of CMDP is to search the optimal policy $\pi_\star$:

$$\pi_\star = \arg\max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\mathcal{C}}} J(\pi_{\boldsymbol{\theta}}). \tag{3}$$

Furthermore, we define value functions, action-value functions, and advantage functions for the auxiliary costs in analogy to $V_{\pi_{\boldsymbol{\theta}}}, Q_{\pi_{\boldsymbol{\theta}}}$, and $A_{\pi_{\boldsymbol{\theta}}}$, with c_i replacing r respectively, we denote them as $V_{\pi_{\boldsymbol{\theta}}}^{c_i}, Q_{\pi_{\boldsymbol{\theta}}}^{c_i}$, and $A_{\pi_{\boldsymbol{\theta}}}^{c_i}$. For example, $V_{\pi_{\boldsymbol{\theta}}}^{c_i}(s) = \mathbb{E}_{\pi_{\boldsymbol{\theta}}}\left[\sum_{t=0}^{\infty} \gamma^t c_i(s_t, a_t)|s_0 = s\right]$. Without loss of generality, we will restrict our discussion to the case of one constraint with a cost function c and upper bound b. Finally, we extend the GAE with respect to auxiliary cost function c:

$$\hat{A}_{C,t}^{\texttt{GAE}(\gamma,\lambda)}(s_t, a_t) = \sum_{\ell=0}^{\infty} (\gamma\lambda)^{\ell} \delta_{t+\ell}^{C}, \tag{4}$$

where $\delta_t^C = r_{t+1} + \gamma C(s_{t+1}) - C(s_t)$ is TD error, and $C(\cdot)$ is an estimator of cost function c.

3 Generalized Policy Performance Difference Bounds

In this section, we show some generalized policy optimization performance bounds for $J(\pi_{\boldsymbol{\theta}})$ and $J^c(\pi_{\boldsymbol{\theta}})$. The proposed bounds provide some new surrogate functions with respect to the objective and cost function, which are theoretical foundations for us to design efficient algorithms to improve policy performance and satisfy constraints. Before we present the new performance difference bounds, let us revisit a classic performance difference from Kakade and Langford [2002],

$$J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'}) = (1-\gamma)^{-1} \mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}}}^{\rho_0}(\cdot), a \sim \pi_{\boldsymbol{\theta}}(\cdot|s)}\left[A_{\pi_{\boldsymbol{\theta}'}}(s, a)\right]. \tag{5}$$

Eq.(5) shows a difference between two arbitrary policies $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$ with different parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$. According to (5), we rewrite the policy optimization (3) as follows

$$\pi_\star = \arg\max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\mathcal{C}}} \mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}}}^{\rho_0}(\cdot), a \sim \pi_{\boldsymbol{\theta}}(\cdot|s)}\left[A_{\pi_{\boldsymbol{\theta}'}}(s, a)\right]. \tag{6}$$

However, Eq.(5) or (6) is very intractable for sampling-based policy optimization since it requires the data comes from the (unknown) policy $\pi_{\boldsymbol{\theta}}$ that needed to be learned. In this section, we provide a bound refines the result (5), which provide the sights for surrogate functions to solve problem (3).

3.1 Some Additional Notations

We use a bold lowercase letter to denote a vector, e.g., $\mathbf{a} = (a_1, a_2, \cdots, a_n)$, and its i-th element $\mathbf{a}[i] =: a_i$. Let $\varphi(\cdot) : \mathcal{S} \to \mathbb{R}$ be a function defined on $\mathcal{S}$, $\delta_t^{\varphi} = r(s_{t+1}|s_t, a_t) + \gamma\varphi(s_{t+1}) - \varphi(s_t)$ is TD error with respect to $\varphi(\cdot)$. For two arbitrary policies $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$, we denote $\delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s)$ as the expectation of TD error, and define $\Delta_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}, s)$ as the difference between $\delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s)$ and $\delta_{\pi_{\boldsymbol{\theta}'},t}^{\varphi}(s)$:

$$\delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s) = \mathop{\mathbb{E}}_{\substack{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(\cdot|s)\\ a_t\sim\pi_{\boldsymbol{\theta}}(\cdot|s_t)\\ s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}} \left[\delta_t^{\varphi}\right], \Delta_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}, s) = \mathop{\mathbb{E}}_{\substack{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s)\\ a_t\sim\pi_{\boldsymbol{\theta}'}(\cdot|s_t)\\ s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}} \left[\left(\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}'}(a_t|s_t)} - 1\right)\delta_t^{\varphi}\right].$$

Furthermore, we introduce two vectors $\boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}, \boldsymbol{\Delta}_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \in \mathbb{R}^{|\mathcal{S}|}$, and their components are:

$$\boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}[s] = \delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s), \quad \boldsymbol{\Delta}_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})[s] = \Delta_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}, s). \tag{7}$$

Let matrix $\mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)} = (1-\gamma\lambda)\sum_{t=0}^{\infty}(\gamma\lambda)^t\mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{t+1}$, where $\lambda \in [0, 1]$. It is similar to the normalized discounted distribution $d_{\pi_{\boldsymbol{\theta}}}^{\rho_0}(s)$, we extend it to λ-version and denote it as $d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s)$:

$$d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s) = \mathbb{E}_{s_0\sim\rho_0(\cdot)}\left[(1-\tilde{\gamma})\sum_{t=0}^{\infty}\tilde{\gamma}^t\mathbb{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s_t = s|s_0)\right],$$

where $\tilde{\gamma} = \frac{\gamma(1-\lambda)}{1-\gamma\lambda}$, the probability $\mathbb{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s_t = s|s_0)$ is the (s_0, s)-th component of the matrix product $\left(\mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}\right)^t$. Finally, we introduce a vector $\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} \in \mathbb{R}^{|\mathcal{S}|}$, and its components are: $\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}[s] = d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s)$.

3.2 Main Results

Theorem 1 (Generalized Policy Performance Difference)**.** *For any function $\varphi(\cdot) : \mathcal{S} \to \mathbb{R}$, for two arbitrary policies $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$, for any $p, q \in [1, \infty)$ such that $\frac{1}{p} + \frac{1}{q} = 1$, we define two error terms:*

$$\epsilon_{p,q,t}^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) =: \|\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} - \mathbf{d}_{\pi_{\boldsymbol{\theta}'}}^{\lambda}\|_p\|\boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}\|_q, \tag{8}$$

$$L_{p,q}^{\varphi,\pm}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) =: \frac{1}{1-\tilde{\gamma}}\sum_{t=0}^{\infty}\gamma^t\lambda^t\mathbb{E}_{s\sim d_{\pi_{\boldsymbol{\theta}'}}^{\lambda}(\cdot)}\left[\Delta_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}, s) \pm \epsilon_{p,q,t}^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})\right]. \tag{9}$$

Then, the following bound with respect to policy performance difference $J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'})$ holds:

$$L_{p,q,}^{\varphi,-}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \le J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'}) \le L_{p,q,}^{\varphi,+}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}). \tag{10}$$

Proof. See Appendix E. □

The bound (10) is well-defined, i.e., if $\pi_{\boldsymbol{\theta}} = \pi_{\boldsymbol{\theta}'}$, all the three terms in Eq.(10) are zero identically. From Eq.(9), we know the performance difference bound $L_{p,q}^{\varphi,\pm}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$ (10) can be interpreted by two distinct difference parts: **(i)** the first difference part, i.e., the expectation $\Delta_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}, s)$, which is determined by the difference between TD errors of $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$; **(ii)** the second difference part, i.e., the discounted distribution difference $\epsilon_{p,q,t}^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$, which is determined by the gap between the normalized discounted distribution of $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$. Thus, the difference of both TD errors and discounted distribution determine the policy difference $J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'})$.

The different choices of p and q lead Eq.(10) to be different bounds. If $p = 1, q = \infty$, we denote $\epsilon_{\pi_{\boldsymbol{\theta}},t}^{\varphi} =: \|\boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}\|_q = \max_{s_t\in\mathcal{S}}\mathbb{E}_{a_t\sim\pi_{\boldsymbol{\theta}}(\cdot|s_t), s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}[|\delta_t^{\varphi}|]$, then, according to Lemma 2 (see Appendix F.2), when $p = 1, q = \infty$, then error $\epsilon_{p,q,t}^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$ is reduced to:

$$\epsilon_{p,q,t}^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})\big|_{p=1,q=\infty} \le \frac{\tilde{\gamma}\left(\gamma\lambda(|\mathcal{S}|-1)+1\right)\epsilon_{\pi_{\boldsymbol{\theta}},t}^{\varphi}}{(1-\tilde{\gamma})(1-\gamma\lambda)}\mathbb{E}_{s\sim d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(\cdot)}\left[2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s]\right],$$

where $D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s]$ is the total variational divergence between action distributions at state s, i.e.,

$$2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s] = \sum_{a\in\mathcal{A}}\left|\pi_{\boldsymbol{\theta}'}(a|s) - \pi_{\boldsymbol{\theta}}(a|s)\right|.$$

Finally, let $\varphi = V_{\pi_{\boldsymbol{\theta}'}}$, the left side of (10) in Theorem 1 implies a lower bound of performance difference, which illustrates the worse case of approximation error, we present it in Proposition 1.

Proposition 1. *For any two policies $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$, let $\epsilon^{V}_{\pi_{\boldsymbol{\theta}}}(\pi_{\boldsymbol{\theta}'}) =: \sup_{t\in\mathbb{N}^+}\{\epsilon^{\varphi}_{\pi_{\boldsymbol{\theta}},t} : \varphi = V_{\pi_{\boldsymbol{\theta}'}}\}$, then*

$$J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'}) \geq \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot),a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}\Bigg[A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}'}}(s,a) - \frac{2\tilde{\gamma}\left(\gamma\lambda(|\mathcal{S}|-1)+1\right)\epsilon^{V}_{\pi_{\boldsymbol{\theta}}}(\pi_{\boldsymbol{\theta}'})}{(1-\tilde{\gamma})(1-\gamma\lambda)}D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\Bigg]. \quad (11)$$

The refined bound (11) contains GAE technique that significantly reduces variance while maintaining a tolerable level of bias empirically [Schulman *et al.*, 2016], which implies using the bound (11) as a surrogate function could improve performance potentially for practice. Although GAE has been empirically applied to extensive reinforcement learning tasks, to the best of our knowledge, the result (11) is the first to show a rigorous theoretical analysis to extend the surrogate functions to GAE.

Remark 1 (Unification of [Achiam *et al.*, 2017])**.** *If $\lambda \to 0$, then the distribution $d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot)$ is reduced to $d^{\rho_0}_{\pi_{\boldsymbol{\theta}'}}(\cdot)$ and the bound (11) is reduced to*

$$J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'}) \geq \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}'}}(\cdot),a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}\left[A_{\pi_{\boldsymbol{\theta}'}}(s,a) - 2\frac{\gamma}{1-\gamma}\epsilon^{V}_{\pi_{\boldsymbol{\theta}}}(\pi_{\boldsymbol{\theta}'})D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\right], \quad (12)$$

which matches the result of [Achiam et al.*, 2017, Corollary 1]. That is to say the proposed bound (11) unifies the classic bound (12)*

Let $\varphi = V^{c}_{\pi_{\boldsymbol{\theta}'}}$, Theorem 1 implies an upper bound of cost function as presented in the next Proposition 2, we will use it to make guarantee for safe policy optimization.

Proposition 2. *For any two policies $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$, let $\epsilon^{C}_{\pi_{\boldsymbol{\theta}}}(\pi_{\boldsymbol{\theta}'}) =: \sup_{t\in\mathbb{N}^+}\{\epsilon^{\varphi}_{\pi_{\boldsymbol{\theta}},t} : \varphi = V^{c}_{\pi_{\boldsymbol{\theta}'}}\}$, then*

$$J^{c}(\pi_{\boldsymbol{\theta}}) - J^{c}(\pi_{\boldsymbol{\theta}'}) \leq \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot),a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}\Bigg[A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}'},C}(s,a) + \frac{2\tilde{\gamma}\left(\gamma\lambda(|\mathcal{S}|-1)+1\right)\epsilon^{C}_{\pi_{\boldsymbol{\theta}}}(\pi_{\boldsymbol{\theta}'})}{(1-\tilde{\gamma})(1-\gamma\lambda)}D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\Bigg]. \quad (13)$$

where we calculate $A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}'},C}(s,a)$ according to the data sampled from $\pi_{\boldsymbol{\theta}'}$ and the estimator (4).

All above bound results (11) and (13) can be extended for a total variational divergence to KL-divergence between policies, which are desirable for policy optimization.

Proposition 3. *All the bounds in (11) and (13) hold if we make the following substitution:*

$$\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot)}\left[D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\right] \leftarrow \sqrt{\frac{1}{2}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot)}\left[\mathrm{KL}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\right]},$$

where $\mathrm{KL}(\cdot,\cdot)$ is KL-divergence, and $\mathrm{KL}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s] = \mathrm{KL}(\pi_{\boldsymbol{\theta}'}(\cdot|s),\pi_{\boldsymbol{\theta}}(\cdot|s))$.

4 CUP: Constrained Update Projection

According to the bounds in Proposition 1-3, we develop new surrogate functions to replace the objective and constraints. We propose the CUP (constrained update projection) algorithm that is a two-step approach contains *performance improvement* and *projection*. Due to the limitation of space, we present all the details of the implementation in Appendix C and Algorithm 1.

Step 1: Performance Improvement. According to Proposition 1 and Proposition 3, for an appropriate coefficient α_k, we update policy as:

$$\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}} = \arg\max_{\pi_{\boldsymbol{\theta}}\in\Pi_{\boldsymbol{\theta}}}\left\{\mathop{\mathbb{E}}_{\substack{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)\\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}}\left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)}A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right] - \alpha_k\sqrt{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}(\pi_{\boldsymbol{\theta}_k},\pi_{\boldsymbol{\theta}})[s]\right]}\right\}. \quad (14)$$

This step is a typical minimization-maximization (MM) algorithm [Hunter and Lange, 2004], it includes return maximization and minimization the distance between old policy and new policy. the expectation (14) by sample averages according to the trajectories collected by $\pi_{\boldsymbol{\theta}_k}$.

Step 2: Projection. According to Proposition 2 and Proposition 3, for an appropriate coefficient β_k, we project the policy $\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}$ onto the safe constraint set,

$$\pi_{\boldsymbol{\theta}_{k+1}} = \arg\min_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} D\left(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right), \text{ s.t. } C_{\pi_{\boldsymbol{\theta}_k}}(\pi_{\boldsymbol{\theta}}, \beta_k) \leq b, \tag{15}$$

where $D(\cdot,\cdot)$ (e.g., KL divergence or ℓ_2-norm) measures distance between $\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}$ and $\pi_{\boldsymbol{\theta}}$,

$$C_{\pi_{\boldsymbol{\theta}_k}}(\pi_{\boldsymbol{\theta}}, \beta) = J^c(\pi_{\boldsymbol{\theta}_k}) + \frac{1}{1-\tilde{\gamma}} \mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)} \left[A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}}, C}(s,a)\right] + \beta\sqrt{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}})[s]\right]}.$$

Until now, the particular choice of surrogate function is heuristically motivated, we show the worst-case performance degradation guarantee and approximate satisfaction of safety constraints of CUP in Theorem 2, and its proof is shown in Appendix G.

Theorem 2. *Let* $\chi_k = \mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}})[s]]$, $\iota =: \frac{\tilde{\gamma}(\gamma\lambda(|\mathcal{S}|-1)+1)}{(1-\tilde{\gamma})(1-\gamma\lambda)}$.*If* $\pi_{\boldsymbol{\theta}_k}$ *and* $\pi_{\boldsymbol{\theta}_{k+1}}$ *are generated according to (14)-(15), then the lower bound on policy improvement, and upper bound on constraint violation are*

$$J(\pi_{\boldsymbol{\theta}_{k+1}}) - J(\pi_{\boldsymbol{\theta}_k}) \geq -\frac{\iota\alpha_k\sqrt{2\chi_k}}{1-\tilde{\gamma}}\epsilon^V_{\pi_{\boldsymbol{\theta}_{k+1}}}(\pi_{\boldsymbol{\theta}_k}), \quad J^c(\pi_{\boldsymbol{\theta}_{k+1}}) \leq b + \frac{\iota\beta_k\sqrt{2\chi_k}}{1-\tilde{\gamma}}\epsilon^C_{\pi_{\boldsymbol{\theta}_{k+1}}}(\pi_{\boldsymbol{\theta}_k}).$$

Remark 2 (Asymptotic Safety Guarantee)**.** *Let* $\alpha_k \to 0, \beta_k \to 0$ *as* $k \to \infty$*, Theorem 2 implies a monotonic policy improvement with an asymptotic safety guarantee.*

Practical Implementation. Let $\{(s_t, a_t, r_{t+1}, c_{t+1})\}_{t=1}^T \sim \pi_{\boldsymbol{\theta}_k}$, we denote the empirical KL-divergence with respect to $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$ as: $\hat{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) = \frac{1}{T}\sum_{t=1}^T \mathrm{KL}(\pi_{\boldsymbol{\theta}}(a_t|s_t), \pi_{\boldsymbol{\theta}'}(a_t|s_t))$. We update performance improvement (14) step as follows,

$$\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}} = \arg\max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\theta}} \left\{\frac{1}{T}\sum_{t=1}^T \frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}_k}(a_t|s_t)}\hat{A}_t - \alpha_k\sqrt{\hat{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}})}\right\},$$

where $\hat{A}_t$ is an estimator of $A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)$.

Then we update the projection step by replacing the distance D by KL-divergence, the next Theorem 3 (for its proof, see Appendix C.2) provides a fundamental way for us to solve projection step (15).

Theorem 3. *The constrained problem (37) is equivalent to the following primal-dual problem:*

$$\max_{\nu\geq 0}\min_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} \left\{D\left(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right) + \nu\left(C_{\pi_{\boldsymbol{\theta}_k}}(\pi_{\boldsymbol{\theta}}, \beta) - b\right)\right\}.$$

According to Theorem 3, we solve the constraint problem (15) by the following primal-dual approach,

$$(\pi_{\boldsymbol{\theta}_{k+1}}, \nu_{k+1}) = \arg\min_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}}\max_{\nu\geq 0}\left\{\hat{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}, \pi_{\boldsymbol{\theta}}) + \nu\hat{C}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})\right\}$$

where $\hat{C}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) = \hat{J}^C + \frac{1}{1-\tilde{\gamma}}\cdot\frac{1}{T}\sum_{t=1}^T \frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}_k}(a_t|s_t)}\hat{A}^C_t + \beta_k\sqrt{\hat{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}})} - b$, $\hat{J}^C$and $\hat{A}^C_t$ are estimators for cost-return and cost-advantage.

Finally, let

$$\hat{\mathcal{L}}_{\mathrm{c}}\left(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}, \boldsymbol{\theta}_{k+\frac{1}{2}}, \nu\right) =: \hat{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}, \pi_{\boldsymbol{\theta}}) + \nu\hat{C}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}), \tag{16}$$

we update the parameters $(\boldsymbol{\theta}_{k+1}, \nu_{k+1})$ as follows,

$$\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k - \eta\frac{\partial}{\partial\boldsymbol{\theta}}\hat{\mathcal{L}}_{\mathrm{c}}\left(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}, \boldsymbol{\theta}_{k+\frac{1}{2}}, \nu\right)\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k, \nu=\nu_k}, \tag{17}$$

$$\nu_{k+1} \leftarrow \left\{\nu_k + \eta\frac{\partial}{\partial\nu}\hat{\mathcal{L}}_{\mathrm{c}}\left(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}, \boldsymbol{\theta}_{k+\frac{1}{2}}, \nu\right)\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k, \nu=\nu_k}\right\}_+, \tag{18}$$

where $\eta > 0$ is step-size, $\{\cdot\}_+$ denotes the positive part, i.e., if $x \leq 0$, $\{x\}_+ = 0$, else $\{x\}_+ = x$. We have shown all the details of the implementation in Algorithm 1.

5 Related Work

Due to the limitation of space, for more discussions, see Appendix B and Table 2.

Local Policy Search and Lagrangian Approach. A direct way to solve CMDP (3) is to apply *local policy search* [Peters and Schaal, 2008; Pirotta *et al.*, 2013] over the policy space $\Pi_{\mathcal{C}}$, i.e.,

$$\pi_{\boldsymbol{\theta}_{k+1}} = \arg\max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} J(\pi_{\boldsymbol{\theta}}), \text{ s.t. } J^c(\pi_{\boldsymbol{\theta}}) \leq b, \text{ and } D(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) < \delta, \tag{19}$$

where δ is a positive scalar, $D(\cdot,\cdot)$ is some distance measure. For practice, the local policy search (19) is challenging to implement because it requires evaluation of the constraint function c to determine whether a proposed point π is feasible [Zhang *et al.*, 2020]. Besides, Li and Belta [2019]; Cheng *et al.* [2019]; Liu *et al.* [2020] provide a local policy search via the barrier function. The key idea of the proposed CUP is parallel to Barrier functions. When updating policy according to samples, local policy search (19) requires off-policy evaluation [Achiam *et al.*, 2017], which is very challenging for high-dimension control problem [Duan *et al.*, 2016; Yang *et al.*, 2018, 2021a].

A way to solve CMDP (3) is Lagrangian approach that is also known as primal-dual problem:

$$(\pi_\star, \lambda_\star) = \arg\min_{\lambda \geq 0} \max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} \left\{ J(\pi_{\boldsymbol{\theta}}) - \lambda (J^c(\pi_{\boldsymbol{\theta}}) - b) \right\}. \tag{20}$$

Although extensive canonical algorithms are proposed to solve problem (20), e.g., [Liang *et al.*, 2018; Tessler *et al.*, 2019; Paternain *et al.*, 2019; Le *et al.*, 2019; Russel *et al.*, 2020; Xu *et al.*, 2020; Satija *et al.*, 2020; Chen *et al.*, 2021], the policy updated by Lagrangian approach may be infeasible w.r.t. CMDP (3). This is hazardous in reinforcement learning when one needs to execute the intermediate policy (which may be unsafe) during training [Chow *et al.*, 2018].

Constrained Policy Optimization (CPO). Recently, Achiam *et al.* [2017] suggest to replace the cost constraint with a surrogate cost function which evaluates the constraint $J^c(\pi_{\boldsymbol{\theta}})$ according to the samples collected from the current policy $\pi_{\boldsymbol{\theta}_k}$, see Eq.(21)-(23), Appendix B.1. Existing recent works (e.g., [Achiam *et al.*, 2017; Vuong *et al.*, 2019; Yang *et al.*, 2020; Han *et al.*, 2020; Bisi *et al.*, 2020; Bharadhwaj *et al.*, 2021]) try to find some convex approximations to replace the term $A_{\pi_{\boldsymbol{\theta}_k}}(s,a)$ and $\bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})$ in Eq.(21)-(23). Such first-order and second-order approximations turn a non-convex problem (21)-(23) to be a convex problem, it seems to make a simple solution, but this approach results in many error sources and troubles in practice. Firstly, it still lacks a theory analysis to show the difference between the non-convex problem (21)-(23) and its convex approximation. Policy optimization is a typical non-convex problem [Yang *et al.*, 2021b]; its convex approximation may introduce some error for its original issue. Secondly, CPO updates parameters according to conjugate gradient [Süli and Mayers, 2003], and its solution involves the inverse Fisher information matrix, which requires expensive computation for each update.

Instead of using a convex approximation for the objective function, the proposed CUP algorithm improves CPO and PCPO at least two aspects. Firstly, the CUP directly optimizes the surrogate objective function via the first-order method, and it does not depend on any convex approximation. Thus, the CUP effectively avoids the expensive computation for the inverse Fisher information matrix. Secondly, CUP extends the surrogate objective function to GAE. Although Zhang *et al.* [2020] has used the GAE technique in experiments, to the best of our knowledge, it still lacks a rigorous theoretical analysis involved GAE before we propose CUP.

6 Experiment

In this section, we aim to answer the following three issues: **(I)**. Does CUP satisfy the safety constraints in different environments? Does CUP performs well with different cost limits? **(II)**. How does CUP compare to the state-of-the-art safe RL algorithms? **(III)**. Does CUP play a sensibility during the hyper-parameters in the tuning processing?

We train different robotic agents using five MuJoCo physical simulators [Todorov *et al.*, 2012] which are open by OpenAI Gym API [Brockman *et al.*, 2016], and Safety Gym [Ray *et al.*, 2019]. For more details, see Appendix H.2. Baselines includes CPO [Achiam *et al.*, 2017], PCPO [Yang *et al.*, 2020], TRPO Lagrangian (TRPO-L), PPO Lagrangian (PPO-L) and FOCOPS [Zhang *et al.*, 2020]. TRPO-L and PPO-L are improved by [Chow *et al.*, 2018; Ray *et al.*, 2019], which are based on TRPO [Schulman *et al.*, 2015] and PPO [Schulman *et al.*, 2017]. These two algorithms use the Lagrangian method [Bertsekas, 1997], which applies adaptive penalty coefficients to satisfy the constraint.

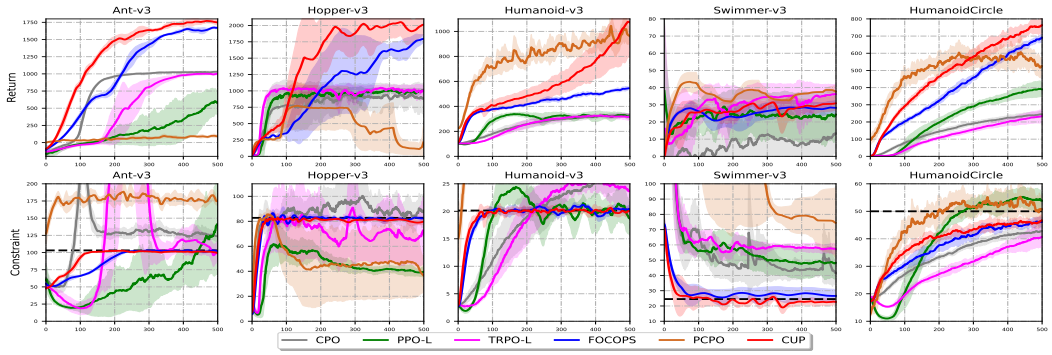


Figure 1: Comparison of CUP to baseline algorithms over 10 seeds on Mujoco.

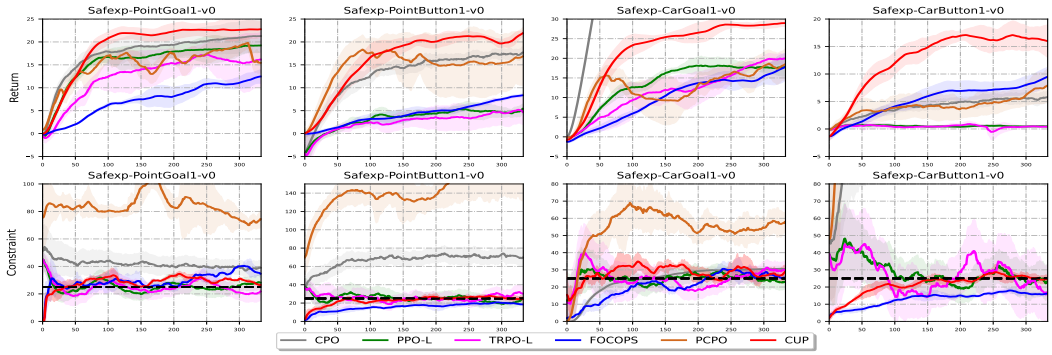


Figure 2: Comparison of CUP to to baseline algorithms over 3 seeds on Satety-Gym.

6.1 Evaluation CUP and Comparison Analysis

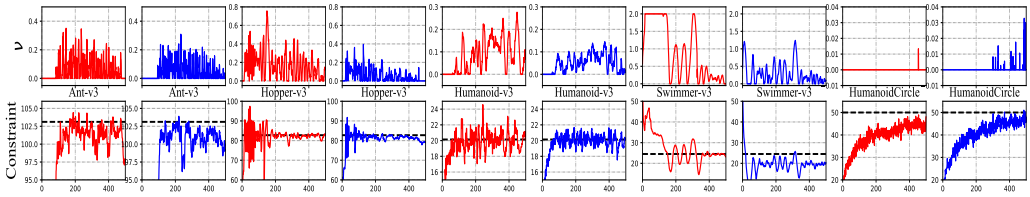
We have shown the Learning curves for CUP, and other baselines in Figure 1-2, and Table 1 summarizes the performance of all algorithms. Results show that CUP quickly stabilizes the constraint return around the limit value while converging the objective return to higher values faster. In most cases, the traces of constraint from CUP almost coincide with the dashed black line of the limit. By contrast, the baseline algorithms frequently suffer from over or under the correction.

From Figure 1, we know initial policies of the baseline algorithms are not guaranteed to be feasible, such as in Swimmer-v3, while CUP performs the best and keeps safety learning in Swimmer-v3 tasks. In the HumanoidCircle task, all the algorithms learn steadily to obtain a safe policy, except PPO-L. Additionally, we observed that CUP brings the policy back to the feasible range faster than other baselines in the HumanoidCircle task. In the Ant-v3 task, only the FOCOPS and the proposed CUP learn safely, and both CPO and TRPO-L violate the safety constraints significantly. Besides, although FOCOPS and CUP converge to a safe policy, CUP obtains a better reward performance than FOCOPS in the Ant-v3 task. The result of Figure 2 is relatively complex, the initial policies of the CPO and PCPO are not guaranteed to be feasible on both Safexp-PointGoal1-v0 and Safexp-PointButton1-v0. We think it is not accidental, but it partially provides corroboration of the previous discussions in Appendix B. Both CPO and PCPO use first-order and second-order approximation to approximate a non-convex problem as a convex problem, which inevitably produces a significant deviation from the original RL problem, and it is more serious in large-scale and complex control systems.

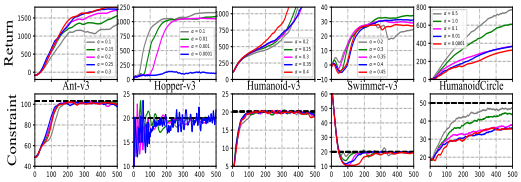
From Table 1, we know although PPO-L achieves a reward of 35.58 ± 5.68 outperforms CUP in Swimmer-v3, PPO-L obtains a cost of 54.91 ± 3.93 that violates the cost limit of 24.5 significantly, which implies PPO-L learns a dangerous policy under this setting. On the other hand, Figure 1 has shown that CUP generally gains higher returns than different baselines while enforcing the cost constraint. Mainly, CUP achieves a reward performance of 2025.56 ± 122.35 that significantly outperforms all the baseline algorithms. Additionally, after equal iterations, CUP performs a greater speed of stabilizing the constraint return around the limit value and is quicker to find feasible policies to gain a more significant objective return.

Environment		CPO	TRPO-L	PPO-L	PCPO	FOCOPS	**CUP**
Ant-v3	Return	1030.17 ± 8.15	480.86 ± 161.05	1012.02 ± 17.26	90.83 ± 17.66	1662.53 ± 17.40	$\mathbf{1743.66 \pm 40.5}$
cost limit: 103	Constraint	120.76 ± 4.80	131.07 ± 67.9	112.45 ± 15.48	174.80 ± 5.53	101.31 ± 0.41	99.11 ± 0.93
Hopper-v3	Return	875.89 ± 285.17	1025.49 ± 10.68	1010.2 ± 61.48	214.90 ± 101.22	1687.72 ± 24.38	$\mathbf{2025.56 \pm 122.35}$
cost limit: 83	Constraint	76.6 ± 10.62	40.36 ± 4.75	83.28 ± 31.19	36.63 ± 12.54	102.3 ± 1.455	79.98 ± 2.306
Swimmer-v3	Return	18.77 ± 6.56	27.35 ± 10.07	$\mathbf{35.58 \pm 5.68}$	37.73 ± 3.56	28.15 ± 4.30	33.38 ± 0.54
cost limit: 24.5	Constraint	42.07 ± 3.31	49.58 ± 7.46	54.91 ± 3.93	74.39 ± 22.71	26.54 ± 4.16	23.31 ± 0.052
Humanoid-v3	Return	326.95 ± 16.00	307.71 ± 24.71	322.11 ± 25.54	962.13 ± 57.94	542.5 ± 4.76	$\mathbf{1066.83 \pm 266.12}$
cost limit: 20.0	Constraint	26.13 ± 2.13	18.22 ± 3.04	22.94 ± 4.54	48.66 ± 3.52	20.04 ± 0.19	19.91 ± 0.36
Humanoid-Circle	Return	237.54 ± 23.20	384.45 ± 47.66	243.35 ± 37.90	525.23 ± 48.32	713.04 ± 9.25	$\mathbf{768.65 \pm 63.70}$
cost limit: 50.0	Constraint	43.64 ± 1.91	53.77 ± 1.48	41.17 ± 3.98	50.80 ± 4.57	47.73 ± 0.64	48.23 ± 0.65

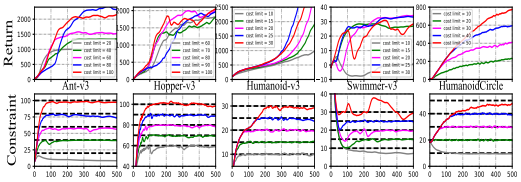
Table 1: Average results for baseline algorithms and CUP over 10 seeds the last 500 iterations.



(a) Cost constraint with respect to hyper-parameter ν (defined in Projection step).



(b) Performance w.r.t. penalty factor α.



(c) Performance w.r.t. cost limit.

Figure 3: Sensitivity analysis for hyper-parameters tuning with respect to ν, α and cost limit.

6.2 Sensitivity Analysis for Hyper-Parameters Tuning

Hyper-parameters tuning is necessary to achieve efficient policy improvement and enforce constraints. We investigate the performance with respect to the parameters: ν, step-size α, and cost limit b.

From Figure 3 (a), we know if the estimated cost under the target threshold b, then ν keeps calm, which implies ν is not activated. Such an empirical phenomenon gives significant expression to the Ant-v3, Humanoid-v3, and Hopper-v3 environments. While if the estimated cost exceeds the target threshold b, ν will be activated, which requires the agent to play a policy on the safe region. Those empirical results are consistent with the update rule of ν: $\nu_{k+1} = \{\nu_k + \eta(\hat{J}_k^C - b)\}_+$, which implies the projection of CUP plays an important role for the agent to learn a safe policy. Furthermore, Figure 3 (b) shows that the performance of CUP is still very stable for different settings of α, and the constraint value of CUP also still fluctuates around the target value. The different value achieved by CUP in different setting α is affected by the simulated environment and constraint thresholds, which are easy to control. Finally, Figure 3 (c) shows that CUP learns safe policies stably under the cost limit thresholds. We compare policy performance and cost under different cost limit settings. For example, in the Swimmer-v3, we set cost limit b among $\{10, 15, 20, 25, 30\}$. Different cost limit setting implies different difficulty for learning, results show that CUP is scalable to various complex tasks, which means CUP is robust to different cost limit settings for various safe RL tasks.

7 Conclusion

This paper proposes the CUP algorithm with a theoretical safety guarantee. We derive the CUP based on the newly proposed surrogate functions with respect to objectives and constraints and provide a practical implementation of CUP that does not depend on any convex approximation. We compared CUP against a comprehensive list of safe RL baselines on a wide range of tasks, which shows the effectiveness of CUP where the agent satisfies safe constraints.

References

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of International Conference on Machine Learning (ICML)*, volume 70, pages 22–31, 2017.

M Mehdi Afsar, Trafford Crump, and Behrouz Far. Reinforcement learning based recommender systems: A survey. *arXiv preprint arXiv:2101.06286*, 2021.

Eitan Altman. *Constrained Markov decision processes*. CRC Press, 1999.

Richard Bellman. A markovian decision process. *Journal of mathematics and mechanics*, 6(5):679–684, 1957.

Dimitri P Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.

Homanga Bharadhwaj, Aviral Kumar, Nicholas Rhinehart, Sergey Levine, Florian Shkurti, and Animesh Garg. Conservative safety critics for exploration. In *International Conference on Learning Representations (ICLR)*, 2021.

Lorenzo Bisi, Luca Sabbioni, Edoardo Vittori, Matteo Papini, and Marcello Restelli. Risk-averse trust region optimization for reward-volatility reduction. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4583–4589, 2020.

Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

Yi Chen, Jing Dong, and Zhaoran Wang. A primal-dual approach to constrained markov decision processes. *arXiv preprint arXiv:2101.10895*, 2021.

Richard Cheng, Gábor Orosz, Richard M. Murray, and Joel W. Burdick. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*. AAAI Press, 2019.

Yinlam Chow, Ofir Nachum, Edgar Duenez-Guzman, and Mohammad Ghavamzadeh. A lyapunov-based approach to safe reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Imre Csiszár and János Körner. *Information theory: coding theorems for discrete memoryless systems*. Cambridge University Press, 2011.

Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning (ICML)*, pages 1329–1338, 2016.

Evan Greensmith, Peter L Bartlett, and Jonathan Baxter. Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research (JMLR)*, 5(Nov):1471–1530, 2004.

Minghao Han, Lixian Tian, Yuanand Zhang, Jun Wang, and Wei Pan. Reinforcement learning control of constrained dynamic systems with uniformly ultimate boundedness stability guarantee. *arXiv preprint arXiv:2011.06882*, 2020.

David R Hunter and Kenneth Lange. A tutorial on mm algorithms. *The American Statistician*, 58(1):30–37, 2004.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of International Conference on Machine Learning (ICML)*, volume 2, pages 267–274, 2002.

Bingyi Kang, Shie Mannor, and Jiashi Feng. Learning safe policies with cost-sensitive advantage estimation, 2021. https://openreview.net/forum?id=uVnhiRaW3J.

Hoang Le, Cameron Voloshin, and Yisong Yue. Batch policy learning under constraints. In *International Conference on Machine Learning (ICML)*, pages 3703–3712, 2019.

Xiao Li and Calin Belta. Temporal logic guided safe reinforcement learning using control barrier functions. *arXiv preprint arXiv:1903.09885*, 2019.

Qingkai Liang, Fanyu Que, and Eytan Modiano. Accelerated primal-dual policy optimization for safe reinforcement learning. *arXiv preprint arXiv:1802.06480*, 2018.

Yongshuai Liu, Jiaxin Ding, and Xin Liu. Ipo: Interior-point policy optimization under constraints. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04), 2020.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

OpenAI. Openai five defeats dota 2 world champions, 2019. https://openai.com/blog/openai-five-defeats-dota-2-world-champions/.

Santiago Paternain, Luiz FO Chamon, Miguel Calvo-Fullana, and Alejandro Ribeiro. Constrained reinforcement learning has zero duality gap. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Jan. Peters and Stefan. Schaal. Reinforcement learning of motor skills with policy gradients. *Neural Netw*, 21(4):682–697, 2008.

M. Pirotta, M. Restelli, A. Pecorino, and D. Calandriello. Safe policy iteration. In *International Conference on Machine Learning (ICML)*, pages 307–315, 2013.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking Safe Exploration in Deep Reinforcement Learning. 2019.

Reazul Hasan Russel, Mouhacine Benosman, and Jeroen Van Baar. Robust constrained-mdps: Soft-constrained robust policy optimization under model uncertainty. *arXiv preprint arXiv:2010.04870*, 2020.

Harsh Satija, Philip Amortila, and Joelle Pineau. Constrained markov decision processes via backward value functions. In *International Conference on Machine Learning (ICML)*, pages 8502–8511, 2020.

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

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *International Conference on Learning Representations (ICLR)*, 2016.

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

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354, 2017.

Endre Süli and David F Mayers. *An introduction to numerical analysis*. Cambridge university press, 2003.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 1998.

Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1057–1063, 2000.

Chen Tessler, Daniel J Mankowitz, and Shie Mannor. Reward constrained policy optimization. *International Conference on Learning Representation (ICLR)*, 2019.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012.

Quan Vuong, Yiming Zhang, and Keith W Ross. Supervised policy update for deep reinforcement learning. In *International Conference on Learning Representation (ICLR)*, 2019.

Christopher John Cornish Hellaby Watkins. Learning from delayed rewards. 1989.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

Cathy Wu, Aravind Rajeswaran, Yan Duan, Vikash Kumar, Alexandre M Bayen, Sham Kakade, Igor Mordatch, and Pieter Abbeel. Variance reduction for policy gradient with action-dependent factorized baselines. *International Conference on Learning Representation (ICLR)*, 2018.

Tengyu Xu, Yingbin Liang, and Guanghui Lan. A primal approach to constrained policy optimization: Global optimality and finite-time analysis. *arXiv preprint arXiv:2011.05869*, 2020.

Long Yang, Minhao Shi, Qian Zheng, Wenjia Meng, and Gang Pan. A unified approach for multi-step temporal-difference learning with eligibility traces in reinforcement learning. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 2984–2990, 2018.

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

Long Yang, Gang Zheng, Yu Zhang, Qian Zheng, Pengfei Li, and Gang Pan. On convergence of gradient expected sarsa (λ). In *AAAI*, 2021.

Long Yang, Qian Zheng, and Gang Pan. Sample complexity of policy gradient finding second-order stationary points. In *AAAI*, 2021.

Yiming Zhang, Quan Vuong, and Keith Ross. First order constrained optimization in policy space. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section **??**.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Abstract and Section 1.
   (b) Did you describe the limitations of your work? [No]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] We ensure our paper to conform to the ethics review guidelines.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 3 and Section 4.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix E and Appendix G.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See the URL in the supplementary material for the code, and see H for environments of experiments.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix H
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Appendix H
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix H
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We provide the code for our implementation of CUP in the supplemental material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] We use open source safe reinforcement learning environments, see Appendix H.2
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] Our data does not contain any personally identifiable information or offensive content.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Notations

A.1 Matrix Index

In this paper, we use a bold capital letter to denote matrix, e.g., $\mathbf{A} = (a_{i,j}) \in \mathbb{R}^{m\times n}$, and its (i,j)-th element denoted as

$$\mathbf{A}[i,j] =: a_{i,j},$$

where $1 \le i \le m, 1 \le j \le n$. Similarly, a bold lowercase letter denotes a vector, e.g., $\mathbf{a} = (a_1, a_2, \cdots, a_n) \in \mathbb{R}^n$, and its i-th element denoted as

$$\mathbf{a}[i] =: a_i,$$

where $1 \le i \le n$.

A.2 Key Notations of Reinforcement Learning

For convenience of reference, we list key notations that have be used in this paper.

A.2.1 Value Function and Dynamic System of MDP.

$\mathbf{r}_{\pi_{\boldsymbol{\theta}}},\ R_{\pi_{\boldsymbol{\theta}}}(s),$	:	$\mathbf{r}_{\pi_{\boldsymbol{\theta}}} \in \mathbb{R}^{\lvert\mathcal{S}\rvert}$ is the expected vector reward according to $\pi_{\boldsymbol{\theta}}$, i.e., their components are: $\mathbf{r}_{\pi_{\boldsymbol{\theta}}}[s] = \sum_{a\in\mathcal{A}}\sum_{s'\in\mathcal{S}} \pi_{\boldsymbol{\theta}}(a\lvert s) r(s'\lvert s,a) =: R_{\pi_{\boldsymbol{\theta}}}(s),\ s\in\mathcal{S}$.
$\mathbf{v}_{\pi_{\boldsymbol{\theta}}},\ V_{\pi_{\boldsymbol{\theta}}}(s),$	:	$\mathbf{v}_{\pi_{\boldsymbol{\theta}}} \in \mathbb{R}^{\lvert\mathcal{S}\rvert}$ is the vector that stores all the state value functions, and its components are: $\mathbf{v}_{\pi_{\boldsymbol{\theta}}}[s] = V_{\pi_{\boldsymbol{\theta}}}(s),\ s\in\mathcal{S}$.
$\rho(\cdot), \boldsymbol{\rho}$	:	$\rho(s)$: the initial state distribution of state s; $\boldsymbol{\rho} \in \mathbb{R}^{\lvert\mathcal{S}\rvert}$, and $\boldsymbol{\rho}[s] = \rho(s)$.
$\mathbf{P}_{\pi_{\boldsymbol{\theta}}}$	:	Single-step state transition matrix by executing $\pi_{\boldsymbol{\theta}}$.
$\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s'\lvert s)$	:	Single-step state transition probability from s to s' by executing $\pi_{\boldsymbol{\theta}}$, and it is the (s,s')-th component of the matrix $\mathbf{P}_{\pi_{\boldsymbol{\theta}}}$, i.e., $\mathbf{P}_{\pi_{\boldsymbol{\theta}}}[s,s'] = \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s'\lvert s)$.
$\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s'\lvert s)$	:	The probability of visiting the state s' after t time steps from the state s by executing $\pi_{\boldsymbol{\theta}}$, and it is the (s,s')-th component of the matrix $\mathbf{P}_{\pi_{\boldsymbol{\theta}}}$, i.e., $\mathbf{P}^t_{\pi_{\boldsymbol{\theta}}}[s,s'] = \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s'\lvert s)$.
$d^{s_0}_{\pi_{\boldsymbol{\theta}}}(s),\ d^{\rho_0}_{\pi_{\boldsymbol{\theta}}}(s)$	:	The normalized discounted distribution of the future state s encountered starting at s_0 by executing $\pi_{\boldsymbol{\theta}}$: $d^{s_0}_{\pi_{\boldsymbol{\theta}}}(s) =: (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s\lvert s_0)$. Since $s_0 \sim \rho(\cdot)$, we define $d^{\rho_0}_{\pi_{\boldsymbol{\theta}}}(s) =: \mathbb{E}_{s_0\sim\rho(\cdot)}[d^{s_0}_{\pi_{\boldsymbol{\theta}}}(s)]$.
$\mathbf{d}^{\rho_0}_{\pi_{\boldsymbol{\theta}}}$	:	It stores all the normalized discounted state distributions $d^{\rho_0}_{\pi_{\boldsymbol{\theta}}}(s), \in \mathcal{S}$, i.e., $\mathbf{d}^{\rho_0}_{\pi_{\boldsymbol{\theta}}} \in \mathbb{R}^{\lvert\mathcal{S}\rvert}$, and its components are: $\mathbf{d}^{\rho_0}_{\pi_{\boldsymbol{\theta}}}[s] = d^{\rho_0}_{\pi_{\boldsymbol{\theta}}}(s)$.

A.2.2 Extend them to λ-version.

$\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}$	:	$\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}} = (1-\gamma\lambda)\sum_{t=0}^{\infty}(\gamma\lambda)^t \mathbf{P}^{t+1}_{\pi_{\boldsymbol{\theta}}}$.
$\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s'\lvert s)$	:	$\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s'\lvert s) =: \mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}[s,s'] = (1-\gamma\lambda)\sum_{t=0}^{\infty}(\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1} = s'\lvert s)$.
$\mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}},\ R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s)$	:	$\mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}} = \sum_{t=0}^{\infty}(\gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}})^t \mathbf{r}_{\pi_{\boldsymbol{\theta}}};\ R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) =: \mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}[s]$.
$\tilde{\gamma}$	:	$\tilde{\gamma} = \dfrac{\gamma(1-\lambda)}{1-\gamma\lambda}$.
$d^{s_0,\lambda}_{\pi_{\boldsymbol{\theta}}}(s)$	:	$d^{s_0,\lambda}_{\pi_{\boldsymbol{\theta}}}(s) = (1-\tilde{\gamma})\sum_{t=0}^{\infty}\tilde{\gamma}^t \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s\lvert s_0)$.
$d^{\lambda}_{\pi_{\boldsymbol{\theta}}}(s),\ \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}}$	:	$d^{\lambda}_{\pi_{\boldsymbol{\theta}}}(s) = \mathbb{E}_{s_0\sim\rho_0(\cdot)}\left[d^{s_0,\lambda}_{\pi_{\boldsymbol{\theta}}}(s)\right],\ \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}}[s] = d^{\lambda}_{\pi_{\boldsymbol{\theta}}}(s)$.

A.2.3 TD error w.r.t. any function $\varphi(\cdot)$.

δ^{φ}_t	:	$\delta^{\varphi}_t = r(s_{t+1}\lvert s_t,a_t) + \gamma\varphi(s_{t+1}) - \varphi(s_t)$.
$\delta^{\varphi}_{\pi_{\boldsymbol{\theta}},t}(s)$	:	$\delta^{\varphi}_{\pi_{\boldsymbol{\theta}},t}(s) = \mathbb{E}_{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(\cdot\lvert s),a_t\sim\pi_{\boldsymbol{\theta}}(\cdot\lvert s_t),s_{t+1}\sim\mathbb{P}(\cdot\lvert s_t,a_t)}\left[\delta^{\varphi}_t\right]$.
$\boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t}$	:	$\boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t}[s] = \delta^{\varphi}_{\pi_{\boldsymbol{\theta}},t}(s)$.
$\Delta^{\varphi}_t(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}'},s)$	:	$\mathbb{E}_{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot\lvert s),a_t\sim\pi_{\boldsymbol{\theta}'}(\cdot\lvert s_t),s_{t+1}\sim\mathbb{P}(\cdot\lvert s_t,a_t)}\left[\left(\dfrac{\pi_{\boldsymbol{\theta}}(a_t\lvert s_t)}{\pi_{\boldsymbol{\theta}'}(a_t\lvert s_t)} - 1\right)\delta^{\varphi}_t\right]$.
$\boldsymbol{\Delta}^{\varphi}_t(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}'})$	:	$\boldsymbol{\Delta}^{\varphi}_t(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}'})[s] = \Delta^{\varphi}_t(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}'},s)$.

B Additional Discussion about Related Work

This section reviews three typical safe reinforcement learning algorithms: CPO [Achiam *et al.*, 2017], PCPO [Yang *et al.*, 2020] and FOCOPS [Zhang *et al.*, 2020]. Those algorithms also use new surrogate functions to replace the objective and constraints, which resembles the proposed CUP algorithm. The goal is to present the contribution of our work.

B.1 CPO [Achiam *et al.*, 2017]

For a given policy $\pi_{\boldsymbol{\theta}_k}$, CPO updates new policy $\pi_{\boldsymbol{\theta}_{k+1}}$ as follows:

$$\pi_{\boldsymbol{\theta}_{k+1}} = \arg\max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} \quad \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)} \left[A_{\pi_{\boldsymbol{\theta}_k}}(s,a) \right] \tag{21}$$

$$\text{s.t.} \;\; J^c(\pi_{\boldsymbol{\theta}_k}) + \frac{1}{1-\gamma} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)} \left[A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a) \right] \leq b, \tag{22}$$

$$\bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) = \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})[s]] \leq \delta. \tag{23}$$

It is impractical to solve the problem (21) directly due to the computational cost. [Achiam *et al.*, 2017] suggest to find some convex approximations to replace the term $A_{\pi_{\boldsymbol{\theta}_k}}(s,a)$ and $\bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})$ Eq.(21)-(23).

Concretely, according to (5), Achiam *et al.* [2017] suggest to use first-order Taylor expansion of $J(\pi_{\boldsymbol{\theta}})$ to replace the objective (21) as follows,

$$\frac{1}{1-\gamma} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A_{\pi_{\boldsymbol{\theta}_k}}(s,a) \right] = J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}_k}) \approx (\boldsymbol{\theta} - \boldsymbol{\theta}_k)^\top \nabla_{\boldsymbol{\theta}} J(\pi_{\boldsymbol{\theta}}).$$

Similarly, Achiam *et al.* [2017] use the following approximations to turn the constrained policy optimization (21)-(23) to be a convex problem,

$$\frac{1}{1-\gamma} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a) \right] \approx (\boldsymbol{\theta} - \boldsymbol{\theta}_k)^\top \nabla_{\boldsymbol{\theta}} J^c(\pi_{\boldsymbol{\theta}}), \tag{24}$$

$$\bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) \approx (\boldsymbol{\theta} - \boldsymbol{\theta}_k)^\top \mathbf{H} (\boldsymbol{\theta} - \boldsymbol{\theta}_k), \tag{25}$$

where $\mathbf{H}$ is Hessian matrix of $\bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})$, i.e.,

$$\mathbf{H}[i,j] =: \frac{\partial^2}{\partial \boldsymbol{\theta}_i \partial \boldsymbol{\theta}_j} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)} \left[\mathrm{KL}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})[s] \right],$$

Eq.(25) is the second-oder approximation of (23).

Let $\lambda_\star, \nu_\star$ is the dual solution of the following problem

$$\lambda_\star, \nu_\star = \arg\max_{\lambda\geq 0, \nu \geq 0} \left\{ \frac{-1}{2\lambda} \left(\mathbf{g}^\top \mathbf{H}^{-1} \mathbf{g} - 2\nu r + s v^2 \right) + \nu c - \frac{\lambda\delta}{2} \right\};$$

where $\mathbf{g} = \nabla_{\boldsymbol{\theta}} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)} \left[A_{\pi_{\boldsymbol{\theta}_k}}(s,a) \right]$, $\mathbf{a} = \nabla_{\boldsymbol{\theta}} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)} \left[A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a) \right]$, $r = \mathbf{g}^\top \mathbf{H} \mathbf{a}$, $s = \mathbf{a}^\top \mathbf{H}^{-1} \mathbf{a}$, and $c = J^c(\pi_{\boldsymbol{\theta}_k}) - b$.

Finally, CPO updates parameters according to conjugate gradient as follows: if approximation to CPO is feasible:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \frac{1}{\lambda_\star} \mathbf{H}^{-1} (\mathbf{g} - \nu_\star \mathbf{a}),$$

else,

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \sqrt{\frac{2\delta}{\mathbf{a}^\top \mathbf{H}^{-1} \mathbf{a}}} \mathbf{H}^{-1} \mathbf{a}.$$

B.2 PCPO [Yang *et al.*, 2020]

Projection-Based Constrained Policy Optimization (PCPO) is an iterative method for optimizing policies in a two-step process: the first step performs a local reward improvement update, while the second step reconciles any constraint violation by projecting the policy back onto the constraint set.

Reward Improvement:

$$\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}} = \arg\max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)} \left[A_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right],$$
$$\text{s.t.} \bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) = \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})[s]] \leq \delta;$$

Projection:

$$\pi_{\boldsymbol{\theta}_{k+1}} = \arg\min_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} D\left(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right),$$
$$\text{s.t. } J^c(\pi_{\boldsymbol{\theta}_k}) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)} \left[A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right] \leq b.$$

Then, Yang *et al.* [2020] follows CPO [Achiam *et al.*, 2017] uses convex approximation to original problem, and calculate the update rule as follows,

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \sqrt{\frac{2\delta}{\mathbf{g}^\top \mathbf{H}^{-1}\mathbf{g}}}\mathbf{H}^{-1}\mathbf{g} - \max\left(0, \frac{\sqrt{\frac{2\delta}{\mathbf{g}^\top \mathbf{H}^{-1}\mathbf{g}}}\mathbf{a}^\top \mathbf{H}^{-1}\mathbf{g} + c}{\mathbf{a}^\top \mathbf{L}^{-1}\mathbf{a}}\right)\mathbf{L}^{-1}\mathbf{a},$$

where $\mathbf{L} = \mathbf{I}$ if D is ℓ_2-norm, and $\mathbf{L} = \mathbf{H}$ if D is KL-divergence.

B.3 FOCOPS [Zhang *et al.*, 2020]

Zhang *et al.* [2020] propose the First Order Constrained Optimization in Policy Space (FOCOPS) that is a two-step approach. We present it as follows.

Step1: Finding the optimal update policy. Firstly, for a given policy $\pi_{\boldsymbol{\theta}k}$, we find an optimal update policy $\pi^\star$ by solving the optimization problem (21)-(23) in the non-parameterized policy space.

$$\pi^\star = \arg\max_{\pi\in\Pi} \quad \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi(\cdot|s)} \left[A_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right] \tag{26}$$
$$\text{s.t. } J^c(\pi_{\boldsymbol{\theta}_k}) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi(\cdot|s)} \left[A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right] \leq b, \tag{27}$$
$$\bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) = \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi, \pi_{\boldsymbol{\theta}_k})[s]] \leq \delta. \tag{28}$$

If $\pi_{\boldsymbol{\theta}_k}$ is feasible, then the optimal policy for (26)-(28) takes the following form:

$$\pi^\star(a|s) = \frac{\pi_{\boldsymbol{\theta}_k}(a|s)}{Z_{\lambda,\nu}(s)} \exp\left(\frac{1}{\lambda}\left(A_{\pi_{\boldsymbol{\theta}_k}}(s,a) - \nu A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right)\right), \tag{29}$$

where $Z_{\lambda,\nu}(s)$ is the partition function which ensures (29) is a valid probability distribution, λ and ν are solutions to the optimization problem:

$$\min_{\lambda,\nu\geq 0} \lambda\nu + \nu\tilde{b} + \lambda\mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi^\star(\cdot|s)}\left[Z_{\lambda,\nu}(s)\right],$$

the term $\tilde{b} = (1-\gamma)(b - J^c(\pi_{\boldsymbol{\theta}_k}))$.

Step 2: Projection Then, we project the policy found in the previous step back into the parameterized policy space $\Pi_{\boldsymbol{\theta}}$ by solving for the closest policy $\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}$ to $\pi^\star$ in order to obtain $\pi_{\boldsymbol{\theta}_{k+1}}$:

$$\boldsymbol{\theta}_{k+1} = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi_{\boldsymbol{\theta}}, \pi^\star)[s]].$$

B.4 Comparison to CUP

Comparing to CPO and PCPO, the implementation of CUP does not depend on any convex approximations. CPO learns its objective with the deep neural network via the first-order method (see Appendix C).

Concretely, CPO and PCPO approximate the non-convex objective (or constraints) with first-order or second Taylor expansion, but their implementations still lack a theory to show the error difference between the original objective (or constraints) and its convex approximations. Additionally, their approaches involve the inverse of a high-dimension Fisher information matrix, which causes their algorithms to require a costly computation for each update when solving high-dimensional RL problems. While the proposed CUP does not depend on any convex approximations, it learns the policy via first-order optimization approaches. Thus, CUP does not involve the inverse of a high-dimension Fisher information matrix, which implies CUP requires less memory than CPO and PCPO.

Although FOCOPS is also a non-convex implementation, it heavily depends on the current best-satisfied policy. It is known that the current best policy may not be the optimal policy, and FOCOPS requires to project this policy back into the parametric policy space, which implies FOCOPS reduce the chances for an agent to explore the environment since it may lose in a locally optimal solution. While the proposed CUP does not depend on the current optimal policy, in fact, CUP requires the agent to learn the policy according to (14), the numerical solution is not the current optimal policy, which helps CUP to explore the environment.

Table 2: Comparison of some safe reinforcement algorithms.

Algorithm	Optimization problem	Implementation	Remark
CPO [Achiam *et al.*, 2017]	$\pi_{\boldsymbol{\theta}_{k+1}} = \arg\max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot\|s)} \left[A_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right]$, s.t. $J^c(\pi_{\boldsymbol{\theta}_k}) + \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot\|s)} \left[A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right] \leq b$, $\bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) = \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})[s]] \leq \delta$.	$\boldsymbol{\theta}_{k+1} = \arg\max_{\boldsymbol{\theta}} \ \mathbf{g}^\top(\boldsymbol{\theta} - \boldsymbol{\theta}_k)$, s.t. $c + \mathbf{b}^\top(\boldsymbol{\theta} - \boldsymbol{\theta}_k) \leq 0$, $\frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_k)^\top \mathbf{H}(\boldsymbol{\theta} - \boldsymbol{\theta}_k) \leq \delta$.	Convex Implementation
PCPO [Yang *et al.*, 2020]	Reward Improvement $\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}} = \arg\max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot\|s)} \left[A_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right]$, s.t. $\bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) = \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})[s]] \leq \delta$; Projection $\pi_{\boldsymbol{\theta}_{k+1}} = \arg\min_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} \ D\left(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)$, s.t. $J^c(\pi_{\boldsymbol{\theta}_k}) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot\|s)} \left[A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right] \leq b$.	Reward Improvement $\boldsymbol{\theta}_{k+\frac{1}{2}} = \arg\max_{\boldsymbol{\theta}} \ \mathbf{g}^\top(\boldsymbol{\theta} - \boldsymbol{\theta}_k)$, s.t. $\frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_k)^\top \mathbf{H}(\boldsymbol{\theta} - \boldsymbol{\theta}_k) \leq \delta$; Projection $\boldsymbol{\theta}_{k+1} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_k)^\top \mathbf{L}(\boldsymbol{\theta} - \boldsymbol{\theta}_k)$, s.t. $c + \mathbf{b}^\top(\boldsymbol{\theta} - \boldsymbol{\theta}_k) \leq 0$.	Convex Implementation
FOCOPS [Zhang *et al.*, 2020]	Optimal update policy $\pi^\star = \arg\max_{\pi \in \Pi} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi(\cdot\|s)} \left[A_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right]$, s.t. $J^c(\pi_{\boldsymbol{\theta}_k}) + \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi(\cdot\|s)} \left[A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right] \leq b$, $\bar{D}_{\mathrm{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) = \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi, \pi_{\boldsymbol{\theta}_k})[s]] \leq \delta$; Projection $\pi_{\boldsymbol{\theta}_{k+1}} = \arg\min_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} \ \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi_{\boldsymbol{\theta}}, \pi^\star)[s]]$.	Optimal update policy $\pi^\star(a\|s) = \frac{\pi_{\boldsymbol{\theta}_k}(a\|s)}{Z_{\lambda,\nu}(s)} \exp\left(\frac{1}{\lambda}\left(A_{\pi_{\boldsymbol{\theta}_k}}(s,a) - \nu A^c_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right)\right)$; Projection $\boldsymbol{\theta}_{k+1} = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_{s\sim d^{\rho_0}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}[\mathrm{KL}(\pi_{\boldsymbol{\theta}}, \pi^\star)[s]]$.	Non-Convex Implementation
CUP (Our Work)	Policy Improvement $\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}} = \arg\max_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} \Big\{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot\|s)} \left[A^{\mathrm{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right] - \alpha_k \sqrt{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}})[s]\right]}\Big\}$, Projection $\pi_{\boldsymbol{\theta}_{k+1}} = \arg\min_{\pi_{\boldsymbol{\theta}} \in \Pi_{\boldsymbol{\theta}}} D\left(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)$, s.t. $J^c(\pi_{\boldsymbol{\theta}_k}) + \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}}(\cdot\|s)} \left[A^{\mathrm{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k},C}(s,a)\right] + \beta_k \sqrt{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}})[s]\right]} \leq b$.	Policy Improvement $\boldsymbol{\theta}_{k+\frac{1}{2}} = \arg\max_{\boldsymbol{\theta}} \Big\{\frac{1}{T}\sum_{t=1}^{T} \frac{\pi_{\boldsymbol{\theta}}(a_t\|s_t)}{\pi_{\boldsymbol{\theta}_k}(a_t\|s_t)}\hat{A}_t - \alpha\sqrt{\frac{1}{T}\sum_{t=1}^{T}\mathrm{KL}(\pi_{\boldsymbol{\theta}_k}(\cdot\|s_t), \pi_{\boldsymbol{\theta}}(\cdot\|s_t))}\Big\}$; Projection $\boldsymbol{\theta}_{k+1} = \arg\min_{\boldsymbol{\theta}} \frac{1}{T}\sum_{t=1}^{T}\Big\{\mathrm{KL}\left(\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}(\cdot\|s_t), \pi_{\boldsymbol{\theta}}(\cdot\|s_t)\right) + \nu_k \frac{1-\gamma\lambda}{1-\gamma}\frac{\pi_{\boldsymbol{\theta}}(a_t\|s_t)}{\pi_{\boldsymbol{\theta}_k}(a_t\|s_t)}\hat{A}^C_t\Big\}$.	Non-Convex Implementation

C Constrained Update Projection Algorithm

C.1 Practical Implementation of Performance Improvement

C.1.1 Sample-based Performance Improvement

Let the trajectory $\{(s_t, a_t, r_{t+1}, c_{t+1})\}_{t=1}^{T}$ be sampled according to $\pi_{\boldsymbol{\theta}_k}$, then we denote the empirical KL-divergence with respect to $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$ as follows,

$$\hat{D}_{\mathtt{KL}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) = \frac{1}{T}\sum_{t=1}^{T} \mathtt{KL}(\pi_{\boldsymbol{\theta}}(a_t|s_t), \pi_{\boldsymbol{\theta}'}(a_t|s_t)).$$

We defined the following $\hat{\mathcal{L}}_{\mathrm{R}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})$,

$$\hat{\mathcal{L}}_{\mathrm{R}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}) = \frac{1}{T}\sum_{t=1}^{T} \frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}_k}(a_t|s_t)}\hat{A}_t - \alpha_k\sqrt{\hat{D}_{\mathtt{KL}}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}})}, \tag{30}$$

where $\hat{A}_t$ is an estimator of $A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)$. The term $\hat{\mathcal{L}}_{\mathrm{R}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})$ (30) is an estimator of the next expectation that appears in (14)

$$\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot),\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}\left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)}A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right] - \alpha_k\sqrt{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathtt{KL}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}})[s]\right]}.$$

Then we implement the performance improvement as follows,

$$\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}} = \arg\max_{\pi_{\boldsymbol{\theta}}\in\Pi_{\theta}} \left\{\hat{\mathcal{L}}_{\mathrm{R}}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k})\right\}. \tag{31}$$

C.1.2 Clipped Surrogate Objective

How can the implementation (31) take the biggest possible improvement step on a policy using the data we currently have, without stepping so far that we accidentally cause performance collapse? Now, we present a clip implementation for policy improvement, which is very efficient in practice.

Instead of the previous policy improvement (14), according to PPO Schulman *et al.* [2017], we update the policy as follows,

$$\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}} = \arg\max_{\pi_{\boldsymbol{\theta}}\in\Pi_{\boldsymbol{\theta}}} \left\{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot),\ a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}\left[\mathcal{L}_{\text{clip}}\left(s, a, \pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}, \epsilon\right)\right]\right\},$$

where the the objective $\mathcal{L}_{\text{clip}}$ is defined as follows,

$$\mathcal{L}_{\text{clip}}\left(s, a, \pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}, \epsilon\right) = \min\left\{\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)}A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a),\ \text{clip}\left(\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)}, 1-\epsilon, 1+\epsilon\right)A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right\}, \tag{32}$$

ϵ is a hyperparameter which roughly says how far away the policy $\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}$ is allowed to go from the current policy $\pi_{\boldsymbol{\theta}_k}$. The objective $\mathcal{L}_{\text{clip}}\left(s, a, \pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}, \epsilon\right)$ is complex, we present the insights of this clip mechanism Schulman *et al.* [2017] to make CUP learn stably.

Positive GAE: $A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a) > 0$. Firstly, we consider the positive advantage, which implies the objective $\mathcal{L}_{\text{clip}}\left(s, a, \pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}, \epsilon\right)$ reduces to

$$\mathcal{L}_{\text{clip}}\left(s, a, \pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}_k}, \epsilon\right) = \min\left\{\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)}, 1+\epsilon\right\}A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a). \tag{33}$$

Since $A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a) > 0$, to improve the performance, we need to increase $\pi_{\boldsymbol{\theta}}$. The $\min\{\cdot\}$ operator determines the quantization how much the CUP improves. If the policy improves too much such that

$$\pi_{\boldsymbol{\theta}}(a|s) > (1+\epsilon)\pi_{\boldsymbol{\theta}_k}(a|s),$$

The $\min\{\cdot\}$ operator hit the objective with a ceiling of $(1+\epsilon)A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)$. The clip technique requires CUP learns a policy $\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}$ does not benefit by going far away from the current policy $\pi_{\boldsymbol{\theta}_k}$.

Negative GAE: $A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a) < 0$. Let us consider the negative advantage, which implies the objective $\mathcal{L}_{\text{clip}}\left(s,a,\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\epsilon\right)$ reduces to

$$\mathcal{L}_{\text{clip}}\left(s,a,\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\epsilon\right) = \max\left\{\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)}, 1-\epsilon\right\} A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a). \tag{34}$$

Since $A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a) < 0$, to improve the performance, we need to decrease the policy $\pi_{\boldsymbol{\theta}}(a|s)$. The $\max\{\cdot\}$ operator determines the quantization how much the CUP improves. If the policy decrease too much such that

$$\pi_{\boldsymbol{\theta}}(a|s) < (1-\epsilon)\pi_{\boldsymbol{\theta}_k}(a|s),$$

The $\max\{\cdot\}$ operator hit the objective with a ceiling of $(1-\epsilon)A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)$. Thus, similar to the positive GAE the clip technique requires CUP learns a policy $\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}$ does not benefit by going far away from the current policy $\pi_{\boldsymbol{\theta}_k}$.

C.1.3 Learning from Sampling

To short the expression, we introduce a function $g(\epsilon, A)$ as follows,

$$g(\epsilon, A) = \begin{cases}(1+\epsilon)A & A \geq 0\\ (1+\epsilon)A & A < 0.\end{cases}$$

Then we rewrite the objective (32) as follows,

$$\mathcal{L}_{\text{clip}}\left(s,a,\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\epsilon\right) = \min\left\{\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a), g\left(\epsilon, A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)\right)\right\}. \tag{35}$$

Recall the trajectory $\{(s_t,a_t,r_{t+1},c_{t+1})\}_{t=1}^{T}$ be sampled according to $\pi_{\boldsymbol{\theta}_k}$, we defined the following $\hat{\mathcal{L}}_{\text{clip}}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k})$,

$$\hat{\mathcal{L}}_{\text{clip}}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\epsilon) = \min\left\{\frac{1}{T}\sum_{t=1}^{T}\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}_k}(a_t|s_t)}\hat{A}_t - g\left(\epsilon, \frac{1}{T}\sum_{t=1}^{T}\hat{A}_t\right)\right\}, \tag{36}$$

where $\hat{A}_t$ is an estimator of $A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a)$. The term $\hat{\mathcal{L}}_{\text{clip}}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\epsilon)$ (36) is an estimator of the next expectation that appears in (35).

Then we implement the performance improvement as follows,

$$\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}} = \arg\max_{\pi_{\boldsymbol{\theta}}\in\Pi_{\theta}}\left\{\hat{\mathcal{L}}_{\text{clip}}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\epsilon)\right\},$$

i.e., we obtain the parameter $\boldsymbol{\theta}_{k+\frac{1}{2}}$ according to

$$\boldsymbol{\theta}_{k+\frac{1}{2}} = \boldsymbol{\theta}_k + \eta_1\frac{\partial}{\partial\boldsymbol{\theta}}\hat{\mathcal{L}}_{\text{clip}}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\epsilon)\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k},$$

where η_1 is step-size.

C.2 Practical Implementation of Projection

Recall (15), we introduce the new surrogate function with respected to cost function as follows,

$$C_{\pi_{\boldsymbol{\theta}'}}(\pi_{\boldsymbol{\theta}},\beta) = J^c(\pi_{\boldsymbol{\theta}'}) + \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot),a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}\left[A^{\texttt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}'},C}(s,a) + \beta\sqrt{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot)}\left[\texttt{KL}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\right]}\right],$$

where β is adaptive to the term $\frac{\sqrt{2}\tilde{\gamma}(\gamma\lambda(|\mathcal{S}|-1)+1)\epsilon^{V}_{\pi_{\boldsymbol{\theta}}}(\pi_{\boldsymbol{\theta}'})}{(1-\gamma\lambda)}$. Now, we rewrite the projection step (15) as follows,

$$\pi_{\boldsymbol{\theta}_{k+1}} = \arg\min_{\pi_{\boldsymbol{\theta}}\in\Pi_{\boldsymbol{\theta}}} D\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right), \text{ s.t. } C_{\pi_{\boldsymbol{\theta}_k}}(\pi_{\boldsymbol{\theta}},\beta) \leq b. \tag{37}$$

We update the projection step (15) by replacing the distance function $D(\cdot,\cdot)$ by KL-divergence, and we solve the constraint problem (15) by the primal-dual approach.

Theorem 4. *The constrained problem (37) is equivalent to the following primal-dual problem:*

$$\max_{\nu\geq 0}\min_{\pi_{\boldsymbol{\theta}}\in\Pi_{\boldsymbol{\theta}}}\left\{D\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)+\nu\left(C_{\pi_{\boldsymbol{\theta}_k}}(\pi_{\boldsymbol{\theta}},\beta)-b\right)\right\}.$$

Proof. This result is a direct application of [Boyd and Vandenberghe, 2004, Chapter 5.9], and we also present it in D.1. Firstly, we notice if $D\left(\cdot,\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)$ is KL divergence or ℓ_2-norm, then the constrained problem (37) is a convex problem [1]. In fact, for a given policy $\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}$, $D\left(\cdot,\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)$ is convex over the policy $\Pi_{\boldsymbol{\theta}}$, and $C_{\pi_{\boldsymbol{\theta}'}}(\cdot,\beta)$ is also convex over the policy $\Pi_{\boldsymbol{\theta}}$. Additionally, Slater's condition alway holds since $C_{\pi_{\boldsymbol{\theta}'}}(\pi_{\boldsymbol{\theta}'},\beta)=0$. □

According to Theorem 4, we turn the projection step (37) as the following unconstrained problem,

$$\max_{\nu\geq 0}\min_{\pi_{\boldsymbol{\theta}}\in\Pi_{\boldsymbol{\theta}}}\left\{D\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)+\nu\left(C_{\pi_{\boldsymbol{\theta}_k}}(\pi_{\boldsymbol{\theta}},\beta)-b\right)\right\}. \tag{38}$$

In our implementation, we use KL-divergence as the distance $D(\cdot,\cdot)$ to measure the difference between two policies, then

$$D\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)=\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathtt{KL}\left(\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}},\pi_{\boldsymbol{\theta}}\right)[s]\right], \tag{39}$$

which implies we can rewrite the problem (38) as follows,

$$\max_{\nu\geq 0}\min_{\pi_{\boldsymbol{\theta}}}\left\{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathtt{KL}\left(\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}},\pi_{\boldsymbol{\theta}}\right)[s]\right]+\nu\left(C_{\pi_{\boldsymbol{\theta}_k}}(\pi_{\boldsymbol{\theta}},\beta)-b\right)\right\}. \tag{40}$$

Furthermore, we update the projection step as follows,

$$(\pi_{\boldsymbol{\theta}_{k+1}},\nu_{k+1})=\arg\min_{\pi_{\boldsymbol{\theta}}\in\Pi_{\boldsymbol{\theta}}}\max_{\nu\geq 0}\left\{\hat{\mathcal{L}}_{\mathrm{c}}\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\boldsymbol{\theta}_{k+\frac{1}{2}},\nu\right)\right\},$$

where

$$\hat{\mathcal{L}}_{\mathrm{c}}\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\boldsymbol{\theta}_{k+\frac{1}{2}},\nu\right)=\hat{D}_{\mathtt{KL}}(\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}},\pi_{\boldsymbol{\theta}})+\nu\hat{C}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k}),$$

$$\hat{C}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k})=\hat{J}^C+\frac{1}{1-\tilde{\gamma}}\cdot\frac{1}{T}\sum_{t=1}^{T}\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}_k}(a_t|s_t)}\hat{A}^C_t+\beta_k\sqrt{\hat{D}_{\mathtt{KL}}(\pi_{\boldsymbol{\theta}_k},\pi_{\boldsymbol{\theta}})}-b,$$

$\hat{J}^C$and $\hat{A}^C_t$ are estimators for cost-return and cost-advantage.

Remark 3 (Track for Learning ν)**.** *Particularly, after some simple algebra, we obtain the derivation of $\hat{\mathcal{L}}_c(\cdot)$ with respect to ν as follows,*

$$\frac{\partial\hat{\mathcal{L}}_c\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\boldsymbol{\theta}_{k+\frac{1}{2}},\nu\right)}{\partial\nu}=J^c(\pi_{\boldsymbol{\theta}_k})+\frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot),a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}\left[A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k},C}(s,a)\right]-b. \tag{41}$$

But recall (14) is a minimization-maximization iteration, i.e., we require to minimize the distance $\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\mathrm{KL}\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k}\right)[s]$, which implies $\pi_{\boldsymbol{\theta}}$ is close to $\pi_{\boldsymbol{\theta}_k}$. Thus it is reasonable to consider

$$\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot),a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}\left[A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k},C}(s,a)\right]\approx 0.$$

Thus, in practice, we update ν following a simple way

$$\nu\leftarrow\left\{\nu+\eta(J^c(\pi_{\boldsymbol{\theta}_k})-b)\right\}_+.$$

Finally, we obtain the parameters $(\boldsymbol{\theta}_{k+1},\nu_{k+1})$ as follows,

$$\boldsymbol{\theta}_{k+1}\leftarrow\boldsymbol{\theta}_k-\eta_2\frac{\partial}{\partial\boldsymbol{\theta}}\hat{\mathcal{L}}_{\mathrm{c}}\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\boldsymbol{\theta}_{k+\frac{1}{2}},\nu\right)\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k,\nu=\nu_k}, \tag{42}$$

$$\nu_{k+1}\leftarrow\left\{\nu_k+\eta_2\frac{\partial}{\partial\nu}\hat{\mathcal{L}}_{\mathrm{c}}\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_k},\boldsymbol{\theta}_{k+\frac{1}{2}},\nu\right)\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k,\nu=\nu_k}\right\}_+, \tag{43}$$

where $\{\cdot\}_+$ denotes the positive part, i.e., if $x\leq 0$, $\{x\}_+=0$, else $\{x\}_+=x$. We have shown all the details of the implementation in Algorithm 1.

[1] It is worth noting that $\min_{\pi_{\boldsymbol{\theta}}\in\Pi_{\boldsymbol{\theta}}}\ D\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)$ is a convex problem, while $\min_{\boldsymbol{\theta}\in\mathbb{R}^p}\ D\left(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)$ can be a non-convex problem.

Algorithm 1 Constrained Update Projection (CUP)

Initialize: policy network parameters $\boldsymbol{\theta}_0$; value network parameter $\boldsymbol{\omega}_0$; cost value function parameter $\boldsymbol{\nu}_0$, step-size ν_0;
Hyper-parameters: trajectory horizon T; discount rate γ; episode number M, N, mini-batch size B, positive constant α, η;
for $k = 0, 1, 2, \ldots$ **do**
Collect batch data of M episodes of horizon T in $\cup_{i=1}^{M} \cup_{t=0}^{T} \{(s_{i,t}, a_{i,t}, r_{i,t+1}, c_{i,t+1})\}$ according to current policy $\pi_{\boldsymbol{\theta}_k}$;
Estimate c-return by discount averaging on each episode: $\hat{J}_i^C = \sum_{t=0}^{T} \gamma^t c_{i,t+1}$;
Compute TD errors $\cup_{i=1}^{M} \cup_{t=0}^{T} \{\delta_{i,t}\}$, cost TD errors $\cup_{i=1}^{M} \cup_{t=0}^{T} \{\delta_{i,t}^C\}$:

$$\delta_{i,t} = r_{i,t} + \gamma V_{\boldsymbol{\omega}_k}(s_{i,t}) - V_{\boldsymbol{\omega}_k}(s_{i,t-1}),\ \delta_{i,t}^C = c_{i,t} + \gamma V_{\boldsymbol{\nu}_k}^C(s_{i,t}) - V_{\boldsymbol{\nu}_k}^C(s_{i,t-1});$$

Compute GAE: $\cup_{i=1}^{M} \cup_{t=0}^{T} \{\hat{A}_{i,t}, \hat{A}_{i,t}^C\}$: $\hat{A}_{i,t} = \sum_{j=t}^{T} (\gamma\lambda)^{j-t} \delta_{i,j},\ \hat{A}_{i,t}^C = \sum_{j=t}^{T} (\gamma\lambda)^{j-t} \delta_{i,j}^C$;
Compute target function for value function and cost value function as follows,

$$V_{i,t}^{\text{target}} = \hat{A}_{i,t} + V_{\boldsymbol{\omega}_k}(s_{i,t}),\ \ V_{i,t}^{\text{target},C} = \hat{A}_{i,t}^C + V_{\boldsymbol{\nu}_k}^C(s_{i,t});$$

Store data: $\mathcal{D}_k = \cup_{i=1}^{M} \cup_{t=0}^{T} \left\{(a_{i,t}, s_{i,t}, \hat{A}_{i,t}, \hat{A}_{i,t}^C, V_{i,t}^{\text{target}}, V_{i,t}^{\text{target},C})\right\}$;

$\pi_{\text{old}} \leftarrow \pi_{\boldsymbol{\theta}_k}$; Policy Improvement
for $i = 0, 1, 2, \ldots, M$ **do**

$$\boldsymbol{\theta}_{k+\frac{1}{2}} = \arg\max_{\boldsymbol{\theta}} \left\{ \frac{1}{T} \sum_{t=1}^{T} \frac{\pi_{\boldsymbol{\theta}}(a_{i,t}|s_{i,t})}{\pi_{\text{old}}(a_{i,t}|s_{i,t})} \hat{A}_{i,t} - g\left(\epsilon, \frac{1}{T} \sum_{t=1}^{T} \hat{A}_{i,t}\right) \right\};$$

end for

$\pi_{\text{old}} \leftarrow \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}$; Projection
$\nu_{k+1} = (\nu_k + \eta(\hat{J}_i^C - b))_+$;
for $i = 0, 1, 2, \ldots, M$ **do**

$$\boldsymbol{\theta}_{k+1} = \arg\min_{\boldsymbol{\theta}} \frac{1}{T} \sum_{t=1}^{T} \left\{ \text{KL}(\pi_{\boldsymbol{\theta}_{\text{old}}}(\cdot|s_{i,t}), \pi_{\boldsymbol{\theta}}(\cdot|s_{i,t})) + \nu_k \frac{1-\gamma\lambda}{1-\gamma} \frac{\pi_{\boldsymbol{\theta}}(a_{i,t}|s_{i,t})}{\pi_{\boldsymbol{\theta}_k}(a_{i,t}|s_{i,t})} \hat{A}_{i,t}^C \right\};$$

end for

for each mini-batch $\{(a_j, s_j, \hat{A}_j, \hat{A}_j^C, V_j^{\text{target}}, V_j^{\text{target},C})\}$ of size B from $\mathcal{D}_k$ **do**

$$\boldsymbol{\omega}_{k+1} = \arg\min_{\boldsymbol{\omega}} \sum_{j=1}^{B} \left(V_{\boldsymbol{\omega}}(s_j) - V_j^{\text{target}}\right)^2, \boldsymbol{\nu}_{k+1} = \arg\min_{\boldsymbol{\nu}} \sum_{j=1}^{B} \left(V_{\boldsymbol{\nu}}^c(s_j) - V_j^{\text{target},C}\right)^2;$$

end for
end for

D Preliminaries

In this section, we introduce some new notations and results about convex optimization, state distribution, policy optimization and λ-returns.

D.1 Strong Duality via Slater's Condition

We consider a convex optimization problem:

$$p_\star = \min_x f_0(x), \tag{44}$$

$$\text{s.t. } f_i(x) \leq 0,\ i = 1, 2, \cdots, m, \tag{45}$$

$$h_i(x) = 0,\ i = 1, 2, \cdots, p, \tag{46}$$

where the functions $f_0, f_1, \cdots, f_m$ are convex, and $h_1, \cdots, h_p$ are affine. We denote by $\mathcal{D}$ the domain of the problem (which is the intersection of the domains of all the functions involved), and by $\mathcal{X} \subset \mathcal{D}$ its feasible set.

To the problem we associate the Lagrangian $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$, with values

$$\mathcal{L}(x, \lambda, \nu) = f_0 + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x). \tag{47}$$

The dual function is $g : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$, with values

$$g(\lambda, \nu) = \min_x \mathcal{L}(x, \lambda, \nu). \tag{48}$$

The associated dual problem is

$$d_\star = \max_{\lambda \succeq 0, \nu} g(\lambda, \nu). \tag{49}$$

Slater's condition. We say that the problem satisfies Slater's condition if it is strictly feasible, that is:

$$\exists x_0 \ \in \mathcal{D} : f_i(x_0) < 0, i = 1, \cdots, m,\ h_i(x_0) = 0, i = 1, \cdots, p. \tag{50}$$

Theorem 5 (Strong duality via Slater condition)**.** *If the primal problem (8.1) is convex, and satisfies the weak Slater's condition, then strong duality holds, that is, $p_\star = d_\star$.*

We omit the proof of Theorem 5, for more discussions, please refer to [Boyd and Vandenberghe, 2004, Chapter 5.9].

D.2 State Distribution

We use $\mathbf{P}_{\pi_{\boldsymbol{\theta}}} \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|}$ to denote the state transition matrix by executing $\pi_{\boldsymbol{\theta}}$, and their components are:

$$\mathbf{P}_{\pi_{\boldsymbol{\theta}}}[s, s'] = \sum_{a \in \mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s) \mathbb{P}(s'|s, a) =: \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s^{'}|s),\ \ s, s^{'} \in \mathcal{S},$$

which denotes one-step state transformation probability from s to $s^{'}$.

We use $\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0)$ to denote the probability of visiting s after t time steps from the initial state s_0 by executing $\pi_{\boldsymbol{\theta}}$. Particularly, we notice if $t = 0$, $s_t \neq s_0$, then $\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0) = 0$, i.e.,

$$\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0) = 0,\ \ t = 0 \text{ and } s \neq s_0. \tag{51}$$

Then for any initial state $s_0 \sim \rho(\cdot)$, the following holds,

$$\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0) = \sum_{s' \in \mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_{t-1} = s^{'}) \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t-1} = s^{'}|s_0). \tag{52}$$

Recall $d_{\pi_{\boldsymbol{\theta}}}^{s_0}(s)$ denotes the normalized discounted distribution of the future state s encountered starting at s_0 by executing $\pi_{\boldsymbol{\theta}}$,

$$d_{\pi_{\boldsymbol{\theta}}}^{s_0}(s) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0).$$

Furthermore, since $s_0 \sim \rho_0(\cdot)$, we define

$$d_{\pi_{\boldsymbol{\theta}}}^{\rho_0}(s) = \mathbb{E}_{s_0 \sim \rho_0(\cdot)}[d_{\pi_{\boldsymbol{\theta}}}^{s_0}(s)] = \int_{s_0 \in \mathcal{S}} \rho_0(s_0) d_{\pi_{\boldsymbol{\theta}}}^{s_0}(s) \mathrm{d}s_0$$

as the discounted state visitation distribution over the initial distribution $\rho_0(\cdot)$. We use $\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\rho_0} \in \mathbb{R}^{|\mathcal{S}|}$ to store all the normalized discounted state distributions, and its components are:

$$\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\rho_0}[s] = d_{\pi_{\boldsymbol{\theta}}}^{\rho_0}(s), \quad s \in \mathcal{S}.$$

We use $\boldsymbol{\rho}_0 \in \mathbb{R}^{|\mathcal{S}|}$ to denote initial state distribution vector, and their components are:

$$\boldsymbol{\rho}_0[s] = \rho_0(s), \quad s \in \mathcal{S}.$$

Then, we rewrite $\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\rho_0}$ as the following matrix version,

$$\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\rho_0} = (1-\gamma) \sum_{t=0}^{\infty} (\gamma \mathbf{P}_{\pi_{\boldsymbol{\theta}}})^t \boldsymbol{\rho}_0 = (1-\gamma)(\mathbf{I} - \gamma \mathbf{P}_{\pi_{\boldsymbol{\theta}}})^{-1} \boldsymbol{\rho}_0. \tag{53}$$

D.3 Objective of MDP

Recall $\tau = \{s_t, a_t, r_{t+1}\}_{t \geq 0} \sim \pi_{\boldsymbol{\theta}}$, according to τ, we define the expected return $J(\pi_{\boldsymbol{\theta}}|s_0)$ as follows,

$$J(\pi_{\boldsymbol{\theta}}|s_0) = \mathbb{E}_{\tau \sim \pi_{\boldsymbol{\theta}}}[R(\tau)] = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}}}^{s_0}(\cdot), a \sim \pi_{\boldsymbol{\theta}}(\cdot|s), s' \sim \mathbb{P}(\cdot|s,a)} \left[r(s'|s,a) \right], \tag{54}$$

where $R(\tau) = \sum_{t \geq 0} \gamma^t r_{t+1}$, and the notation $J(\pi_{\boldsymbol{\theta}}|s_0)$ is "conditional" on s_0 is to emphasize the trajectory τ starting from s_0.

Since $s_0 \sim \rho_0(\cdot)$, we define the objective of MDP as follows,

$$J(\pi_{\boldsymbol{\theta}}) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}}}^{\rho_0}(\cdot), a \sim \pi_{\boldsymbol{\theta}}(\cdot|s), s' \sim \mathbb{P}(\cdot|s,a)} \left[r(s'|s,a) \right]. \tag{55}$$

The goal of reinforcement learning is to solve the following optimization problem:

$$\boldsymbol{\theta}_\star = \arg\max_{\boldsymbol{\theta} \in \mathbb{R}^p} J(\pi_{\boldsymbol{\theta}}). \tag{56}$$

D.4 λ-Return

Let $\mathcal{B}_{\pi_{\boldsymbol{\theta}}}$ be the *Bellman operator*:

$$\mathcal{B}_{\pi_{\boldsymbol{\theta}}} : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}, \quad v \mapsto \mathbf{r}_{\pi_{\boldsymbol{\theta}}} + \gamma \mathbf{P}_{\pi_{\boldsymbol{\theta}}} v, \tag{57}$$

where $\mathbf{r}_{\pi_{\boldsymbol{\theta}}} \in \mathbb{R}^{|\mathcal{S}|}$ is the expected reward according to $\pi_{\boldsymbol{\theta}}$, i.e., their components are:

$$\mathbf{r}_{\pi_{\boldsymbol{\theta}}}[s] = \sum_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} \pi_{\boldsymbol{\theta}}(a|s) r(s'|s,a) =: R_{\pi_{\boldsymbol{\theta}}}(s), \quad s \in \mathcal{S}.$$

Let $\mathbf{v}_{\pi_{\boldsymbol{\theta}}} \in \mathbb{R}^{|\mathcal{S}|}$ be a vector that stores all the state value functions, and its components are:

$$\mathbf{v}_{\pi_{\boldsymbol{\theta}}}[s] = V_{\pi_{\boldsymbol{\theta}}}(s), \quad s \in \mathcal{S}.$$

Then, according to Bellman operator (57), we rewrite Bellman equation [Bellman, 1957] as the following matrix version:

$$\mathcal{B}_{\pi_{\boldsymbol{\theta}}} \mathbf{v}_{\pi_{\boldsymbol{\theta}}} = \mathbf{v}_{\pi_{\boldsymbol{\theta}}}. \tag{58}$$

Furthermore, we define λ-*Bellman operator* $\mathcal{B}_{\pi_{\boldsymbol{\theta}}}^{\lambda}$ as follows,

$$\mathcal{B}_{\pi_{\boldsymbol{\theta}}}^{\lambda} = (1-\lambda) \sum_{t=0}^{\infty} \lambda^t (\mathcal{B}_{\pi_{\boldsymbol{\theta}}})^{t+1},$$

which implies

$$\mathcal{B}^{\lambda}_{\pi_{\boldsymbol{\theta}}} : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}, \quad v \mapsto \mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}} + \tilde{\gamma}\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}} v, \tag{59}$$

where

$$\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}} = (1-\gamma\lambda)\sum_{t=0}^{\infty}(\gamma\lambda)^t \mathbf{P}^{t+1}_{\pi_{\boldsymbol{\theta}}}, \;\; \mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}} = \sum_{t=0}^{\infty}(\gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}})^t \mathbf{r}_{\pi_{\boldsymbol{\theta}}}, \;\; \tilde{\gamma} = \frac{\gamma(1-\lambda)}{1-\gamma\lambda}. \tag{60}$$

Let

$$\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s^{'}|s) = \mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}[s, s^{'}] =: (1-\gamma\lambda)\sum_{t=0}^{\infty}(\gamma\lambda)^t \left(\mathbf{P}^{t+1}_{\pi_{\boldsymbol{\theta}}}[s, s^{'}]\right), \tag{61}$$

where $\mathbf{P}^{t+1}_{\pi_{\boldsymbol{\theta}}}[s, s^{'}]$ is the $(s, s^{'})$-th component of matrix $\mathbf{P}^{t+1}_{\pi_{\boldsymbol{\theta}}}$, which is the probability of visiting $s^{'}$ after $t+1$ time steps from the state s by executing $\pi_{\boldsymbol{\theta}}$, i.e.,

$$\mathbf{P}^{t+1}_{\pi_{\boldsymbol{\theta}}}[s, s^{'}] = \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1} = s^{'}|s). \tag{62}$$

Thus, we rewrite $\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s^{'}|s)$ (61) as follows

$$\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s^{'}|s) = (1-\gamma\lambda)\sum_{t=0}^{\infty}(\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1} = s^{'}|s), \;\; s \in \mathcal{S}. \tag{63}$$

Remark 4. *Furthermore, recall the following visitation sequence $\tau = \{s_t, a_t, r_{t+1}\}_{t\geq 0}$ induced by $\pi_{\boldsymbol{\theta}}$, it is similar to the probability $\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s^{'}|s_0)$, we introduce $\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s^{'}|s_0)$ as the probability of transition from state s to state $s^{'}$ after t time steps under the dynamic transformation matrix $\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}$. Then, the following equity holds*

$$\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0) = \sum_{s^{'}\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_{t-1} = s^{'})\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_{t-1} = s^{'}|s_0). \tag{64}$$

Similarly, let

$$\begin{aligned} R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) =: \mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}[s] &= \sum_{t=0}^{\infty}(\gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}})^t \mathbf{r}_{\pi_{\boldsymbol{\theta}}}[s] = \sum_{t=0}^{\infty}(\gamma\lambda)^t \left(\sum_{s^{'}\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s^{'}|s) R_{\pi_{\boldsymbol{\theta}}}(s^{'})\right) \\ &= \sum_{t=0}^{\infty}\sum_{s^{'}\in\mathcal{S}} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t = s^{'}|s) R_{\pi_{\boldsymbol{\theta}}}(s^{'}). \end{aligned} \tag{65}$$

It is similar to normalized discounted distribution $d^{\rho_0}_{\pi_{\boldsymbol{\theta}}}(s)$, we introduce λ-return version of discounted state distribution $d^{\lambda}_{\pi_{\boldsymbol{\theta}}}(s)$ as follows: $\forall s \in \mathcal{S}$,

$$d^{s_0,\lambda}_{\pi_{\boldsymbol{\theta}}}(s) = (1-\tilde{\gamma})\sum_{t=0}^{\infty}\tilde{\gamma}^t \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0), \tag{66}$$

$$d^{\lambda}_{\pi_{\boldsymbol{\theta}}}(s) = \mathbb{E}_{s_0\sim\rho_0(\cdot)}\left[d^{s_0,\lambda}_{\pi_{\boldsymbol{\theta}}}(s)\right], \tag{67}$$

$$\mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}}[s] = d^{\lambda}_{\pi_{\boldsymbol{\theta}}}(s), \tag{68}$$

where $\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0)$ is the (s_0, s)-th component of the matrix $\left(\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right)^t$, i.e.,

$$\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0) =: \left(\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right)^t [s_0, s].$$

Similarly, $\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s^{'}|s)$ is the $(s, s^{'})$-th component of the matrix $\left(\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right)^t$, i.e.,

$$\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s^{'}|s) =: \left(\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right)^t [s, s^{'}].$$

Finally, we rewrite $\mathbf{d}^{\rho_0,\lambda}_{\pi_{\boldsymbol{\theta}}}$ as the following matrix version,

$$\mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}} = (1-\tilde{\gamma})\sum_{t=0}^{\infty}\left(\gamma\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right)^t \boldsymbol{\rho}_0 = (1-\tilde{\gamma})\left(\mathbf{I} - \tilde{\gamma}\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right)^{-1}\boldsymbol{\rho}_0. \tag{69}$$

Remark 5 (λ-Return Version of Bellman Equation)**.** *According to Bellman equation (58),* $\mathbf{v}_{\pi_{\boldsymbol{\theta}}}$ *is fixed point of* λ*-operator* $\mathcal{B}^{\lambda}_{\pi_{\boldsymbol{\theta}}}$*, i.e.,*

$$\mathbf{v}_{\pi_{\boldsymbol{\theta}}} = \mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}} + \tilde{\gamma}\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\mathbf{v}_{\pi_{\boldsymbol{\theta}}}. \tag{70}$$

Recall $\tau = \{s_t, a_t, r_{t+1}\}_{t\geq 0} \sim \pi_{\boldsymbol{\theta}}$*, according to (70), the value function of initial state* s_0 *is*

$$\begin{aligned} V_{\pi_{\boldsymbol{\theta}}}(s_0) &= \mathbf{v}_{\pi_{\boldsymbol{\theta}}}[s_0] = \mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}[s_0] + \tilde{\gamma}\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\mathbf{v}_{\pi_{\boldsymbol{\theta}}}[s_0] \\ &= R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_0) + \tilde{\gamma}\sum_{s^{'}\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_1 = s^{'}|s_0)V_{\pi_{\boldsymbol{\theta}}}(s^{'}). \end{aligned} \tag{71}$$

We unroll the expression of (71) repeatedly, then we have

$$\begin{aligned}
&V_{\pi_{\boldsymbol{\theta}}}(s_0) \\
=&R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_0) + \tilde{\gamma}\sum_{s^{'}\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_1 = s^{'}|s_0)\underbrace{\left(R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s^{'}) + \tilde{\gamma}\sum_{s^{''}\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_2 = s^{''}|s_1 = s^{'})V_{\pi_{\boldsymbol{\theta}}}(s^{''})\right)}_{=V_{\pi_{\boldsymbol{\theta}}}(s^{'})} \\
=&R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_0) + \tilde{\gamma}\sum_{s^{'}\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_1 = s^{'}|s_0)R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s^{'}) \\
&\qquad + \tilde{\gamma}^2\sum_{s^{''}\in\mathcal{S}}\underbrace{\left(\sum_{s^{'}\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_1 = s^{'}|s_0)\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_2 = s^{''}|s_1 = s^{'})\right)}_{\overset{(64)}{=}:\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\left(s_2=s^{''}|s_0\right)} V_{\pi_{\boldsymbol{\theta}}}(s^{''}) \\
=&R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_0) + \tilde{\gamma}\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_1 = s|s_0)R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \tilde{\gamma}^2\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_2 = s|s_0)V_{\pi_{\boldsymbol{\theta}}}(s) \\
=&R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_0) + \tilde{\gamma}\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_1 = s|s_0)R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) \\
&\qquad + \tilde{\gamma}^2\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_2 = s|s_0)\left(R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \tilde{\gamma}\sum_{s^{'}\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_3 = s^{'}|s_2 = s)V_{\pi_{\boldsymbol{\theta}}}(s^{'})\right) \\
=&R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_0) + \tilde{\gamma}\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_1 = s|s_0)R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \tilde{\gamma}^2\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_2 = s|s_0)R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) \\
&\qquad + \tilde{\gamma}^3\sum_{s^{'}\in\mathcal{S}}\underbrace{\left(\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_2 = s|s_0)\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_3 = s^{'}|s_2 = s)\right)}_{=\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_3=s^{'}|s_0)} V_{\pi_{\boldsymbol{\theta}}}(s^{'}) \\
=&R^{(\lambda)}(s_0) + \tilde{\gamma}\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_1 = s|s_0)R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \tilde{\gamma}^2\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_2 = s|s_0)R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) \\
&\qquad + \tilde{\gamma}^3\sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_3 = s|s_0)V_{\pi_{\boldsymbol{\theta}}}(s) \\
=&\cdots \\
=&\sum_{s\in\mathcal{S}}\sum_{t=0}^{\infty}\tilde{\gamma}^t\mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s_t = s|s_0)R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) \overset{(66)}{=} \frac{1}{1-\tilde{\gamma}}\sum_{s\in\mathcal{S}} d^{s_0,\lambda}_{\pi_{\boldsymbol{\theta}}}(s)R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s).
\end{aligned} \tag{72}$$

According to (54) and (72), we have

$$
\begin{aligned}
J(\pi_{\boldsymbol{\theta}}) = \sum_{s_0 \in \mathcal{S}} \rho_0(s_0) V_{\pi_{\boldsymbol{\theta}}}(s_0) \overset{(72)}{=} & \frac{1}{1-\tilde{\gamma}} \sum_{s_0 \in \mathcal{S}} \rho_0(s_0) \sum_{s \in \mathcal{S}} d_{\pi_{\boldsymbol{\theta}}}^{s_0,\lambda}(s) R_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s) \\
= & \frac{1}{1-\tilde{\gamma}} \sum_{s \in \mathcal{S}} \underbrace{\left(\sum_{s_0 \in \mathcal{S}} \rho_0(s_0) d_{\pi_{\boldsymbol{\theta}}}^{s_0,\lambda}(s) \right)}_{= d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s)} R_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s) \\
= & \frac{1}{1-\tilde{\gamma}} \sum_{s \in \mathcal{S}} d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s) R_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s) = \frac{1}{1-\tilde{\gamma}} \mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(\cdot)} \left[R_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s) \right].
\end{aligned} \tag{73}
$$

Finally, we summarize above results in the following Lemma 1.

Lemma 1. *The objective $J(\pi_{\boldsymbol{\theta}})$ (55) can be rewritten as the following version:*

$$
J(\pi_{\boldsymbol{\theta}}) = \frac{1}{1-\tilde{\gamma}} \sum_{s \in \mathcal{S}} d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s) R_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s) = \frac{1}{1-\tilde{\gamma}} \mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(\cdot)} \left[R_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s) \right].
$$

E Proof of Theorem 1

We need the following Proposition 4 to prove Theorem 1, which illustrates an identity for the objective function of policy optimization.

Proposition 4. *For any function $\varphi(\cdot) : \mathcal{S} \to \mathbb{R}$, for any policy $\pi_{\boldsymbol{\theta}}$, for any trajectory satisfies $\tau = \{s_t, a_t, r_{t+1}\}_{t\geq 0} \sim \pi_{\boldsymbol{\theta}}$, let*

$$\delta_t^{\varphi} = r(s_{t+1}|s_t, a_t) + \gamma\varphi(s_{t+1}) - \varphi(s_t),$$
$$\delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s) = \mathbb{E}_{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(\cdot|s), a_t\sim\pi_{\boldsymbol{\theta}}(\cdot|s_t), s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}\left[\delta_t^{\varphi}\right],$$

then, the objective $J(\pi_{\boldsymbol{\theta}})$ (73) can be rewritten as the following version:

$$\begin{aligned} J(\pi_{\boldsymbol{\theta}}) =& \mathbb{E}_{s_0\sim\rho_0(\cdot)}[\varphi(s_0)] + \frac{1}{1-\tilde{\gamma}}\sum_{s\in\mathcal{S}} d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s)\left(\sum_{t=0}^{\infty}\gamma^t\lambda^t\delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s)\right) \\ =& \mathbb{E}_{s_0\sim\rho_0(\cdot)}[\varphi(s_0)] + \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(\cdot)}\left[\sum_{t=0}^{\infty}\gamma^t\lambda^t\delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s)\right]. \end{aligned} \tag{74}$$

We present the proof of of Proposition 4 at the end of this section, see Section E.2.

We introduce a vector $\boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi} \in \mathbb{R}^{|\mathcal{S}|}$ and its components are: for any $s \in \mathcal{S}$

$$\boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}[s] = \delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s). \tag{75}$$

Then, we rewrite the objective as the following vector version

$$J(\pi_{\boldsymbol{\theta}}) = \mathbb{E}_{s_0\sim\rho_0(\cdot)}[\varphi(s_0)] + \frac{1}{1-\tilde{\gamma}}\sum_{t=0}^{\infty}\gamma^t\lambda^t\langle\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}\rangle, \tag{76}$$

where $\langle\cdot,\cdot\rangle$ denotes inner production between two vectors.

E.1 Proof of Theorem 1

Theorem 1 (Generalized Policy Performance Difference) *For any function $\varphi(\cdot) : \mathcal{S} \to \mathbb{R}$, for two arbitrary policy $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$, for any $p, q \in [1, \infty)$ such that $\frac{1}{p} + \frac{1}{q} = 1$, The following bound holds:*

$$\frac{1}{1-\tilde{\gamma}}\sum_{t=0}^{\infty}\gamma^t\lambda^t M_{p,q,t}^{\varphi,-}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \leq J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'}) \leq \frac{1}{1-\tilde{\gamma}}\sum_{t=0}^{\infty}\gamma^t\lambda^t M_{p,q,t}^{\varphi,+}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}), \tag{77}$$

where the terms $M_{p,q,t}^{\varphi,-}$ and $M_{p,q,t}^{\varphi,+}$ are defined in (93)-(94).

Proof. (of Theorem 1)

We consider two arbitrary policies $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$ with different parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$, let

$$D_t^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) =: \langle\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}\rangle - \langle\mathbf{d}_{\pi_{\boldsymbol{\theta}'}}^{\lambda}, \boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}'},t}^{\varphi}\rangle. \tag{78}$$

According to (76), we obtain performance difference as follows,

$$\begin{aligned} J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'}) =& \frac{1}{1-\tilde{\gamma}}\sum_{t=0}^{\infty}\gamma^t\lambda^t\left(\langle\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}\rangle - \langle\mathbf{d}_{\pi_{\boldsymbol{\theta}'}}^{\lambda}, \boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}'},t}^{\varphi}\rangle\right) \\ =& \frac{1}{1-\tilde{\gamma}}\sum_{t=0}^{\infty}\gamma^t\lambda^t D_t^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}), \end{aligned} \tag{79}$$

which requires us to consider the boundedness of the difference $D_t^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$ (78) .

Step 1: Bound the term $D_t^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$ (78).

We rewrite the first term of (78) as follows,

$$\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle = \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle + \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}} - \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle, \tag{80}$$

which is bounded by applying Hölder's inequality to the term $\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}} - \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle$, we rewrite (80) as follows,

$$\begin{aligned} &\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle - \|\mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}} - \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}\|_p \|\boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t}\|_q \\ \leq &\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle \leq \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle + \|\mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}} - \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}\|_p \|\boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t}\|_q, \end{aligned} \tag{81}$$

where $p, q \in [1, \infty)$ and $\frac{1}{p} + \frac{1}{q} = 1$. Let

$$\epsilon^{\varphi,(\lambda)}_{p,q,t}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) =: \|\mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}} - \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}\|_p \|\boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t}\|_q,$$

then we rewrite Eq.(81) as follows,

$$\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle - \epsilon^{\varphi,(\lambda)}_{p,q,t}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \leq \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle \leq \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle + \epsilon^{\varphi,(\lambda)}_{p,q,t}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}). \tag{82}$$

Let

$$M^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) =: \underbrace{\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle}_{\text{Term-I}} - \underbrace{\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}'},t} \rangle}_{\text{Term-II}}, \tag{83}$$

combining the (78) and (82), we achieve the boundedness of $D^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$ as follows

$$M^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) - \epsilon^{\varphi,(\lambda)}_{p,q,t}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \leq D^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \leq M^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) + \epsilon^{\varphi,(\lambda)}_{p,q,t}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}). \tag{84}$$

Step 2: Analyze the term $M^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$ (83).

To analyze (84) further, we need to consider the first term appears in $M^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$ (83):

$$\begin{aligned} \text{Term-I (83)} =& \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle \\ =& \sum_{s \in \mathcal{S}} d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(s) \delta^{\varphi}_{\pi_{\boldsymbol{\theta}},t}(s) = \mathbb{E}_{s \sim d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot)} \left[\delta^{\varphi}_{\pi_{\boldsymbol{\theta}},t}(s) \right] \qquad (85) \\ \overset{(75)}{=}& \mathbb{E}_{s \sim d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot)} \left[\mathbb{E}_{s_t \sim \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(\cdot|s)} [\delta^{\varphi}_{\pi_{\boldsymbol{\theta}}}(s_t)] \right]. \qquad (86) \end{aligned}$$

We notice the following relationship

$$\delta^{\varphi}_{\pi_{\boldsymbol{\theta}},t}(s) = \mathop{\mathbb{E}}_{\substack{s_t \sim \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(\cdot|s) \\ a_t \sim \pi_{\boldsymbol{\theta}}(\cdot|s_t) \\ s_{t+1} \sim \mathbb{P}(\cdot|s_t, a_t)}} [\delta^{\varphi}_t] = \mathop{\mathbb{E}}_{\substack{s_t \sim \mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s) \\ a_t \sim \pi_{\boldsymbol{\theta}'}(\cdot|s_t) \\ s_{t+1} \sim \mathbb{P}(\cdot|s_t, a_t)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}'}(a_t|s_t)} \delta^{\varphi}_t \right], \tag{87}$$

which holds since we use importance sampling: for any distribution $p(\cdot)$ and $q(\cdot)$, for any random variable function $f(\cdot)$,

$$\mathbb{E}_{x \sim p(x)}[f(x)] = \mathbb{E}_{x \sim q(x)} \left[\frac{p(x)}{q(x)} f(x) \right].$$

According to (85), (87), we rewrite the term $\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle$ in Eq.(83) as follows,

$$\text{Term-I (83)} = \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle = \sum_{s \in \mathcal{S}} d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(s) \left(\mathop{\mathbb{E}}_{\substack{s_t \sim \mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s) \\ a_t \sim \pi_{\boldsymbol{\theta}'}(\cdot|s_t) \\ s_{t+1} \sim \mathbb{P}(\cdot|s_t, a_t)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}'}(a_t|s_t)} \delta^{\varphi}_t \right] \right). \tag{88}$$

Now, we consider the second term appears in $M_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$ (83):

$$\text{Term-II (83)} = \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}'},t} \rangle$$

$$= \sum_{s\in\mathcal{S}} d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(s)\delta^{\varphi}_{\pi_{\boldsymbol{\theta}'},t}(s) = \sum_{s\in\mathcal{S}} d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(s) \left(\mathop{\mathbb{E}}_{\substack{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s)\\ a_t\sim\pi_{\boldsymbol{\theta}'}(\cdot|s_t)\\ s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}} [\delta^{\varphi}_t] \right). \tag{89}$$

Finally, take the results (88) and (89) to (83), we obtain the difference between $\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle$ and $\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}'},t} \rangle$, i.e., we achieve a identity for $M_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})$ (83) as follows,

$$M_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \overset{(83)}{=} \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle - \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}'},t} \rangle$$

$$\overset{(88),(89)}{=} \sum_{s\in\mathcal{S}} d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(s) \left(\mathop{\mathbb{E}}_{\substack{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s)\\ a_t\sim\pi_{\boldsymbol{\theta}'}(\cdot|s_t)\\ s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}} \left[\left(\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}'}(a_t|s_t)} - 1 \right) \delta^{\varphi}_t \right] \right). \tag{90}$$

To simplify expression, we introduce a notation as follows,

$$\Delta^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}, s) =: \mathop{\mathbb{E}}_{\substack{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s)\\ a_t\sim\pi_{\boldsymbol{\theta}'}(\cdot|s_t)\\ s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}} \left[\left(\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}'}(a_t|s_t)} - 1 \right) \delta^{\varphi}_t \right], \tag{91}$$

and we use a vector $\boldsymbol{\Delta}^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \in \mathbb{R}^{|\mathcal{S}|}$ to store all the values $\{\Delta^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}, s)\}_{s\in\mathcal{S}}$:

$$\boldsymbol{\Delta}^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'})[s] = \Delta^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}, s).$$

Then we rewrite $\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle - \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}'},t} \rangle$ (90) as follows,

$$M_t^{\varphi}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) = \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t} \rangle - \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}'},t} \rangle$$

$$\overset{(90)}{=} \sum_{s\in\mathcal{S}} d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(s)\Delta^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}, s) = \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\Delta}^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \rangle.$$

Step 3: Bound on $J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'})$.

Recall (84), taking above result in it, we obtain

$$\langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\Delta}^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \rangle - \epsilon^{\varphi,(\lambda)}_{p,q,t}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \leq D^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \leq \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\Delta}^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \rangle + \epsilon^{\varphi,(\lambda)}_{p,q,t}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}). \tag{92}$$

Finally, let

$$M^{\varphi,-}_{p,q,t}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) = \langle \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}, \boldsymbol{\Delta}^{\varphi}_t(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \rangle - \epsilon^{\varphi,(\lambda)}_{p,q,t}(\pi_{\boldsymbol{\theta}}, \pi_{\boldsymbol{\theta}'}) \tag{93}$$

$$= \sum_{s\in\mathcal{S}} d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(s) \left(\mathop{\mathbb{E}}_{\substack{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s)\\ a_t\sim\pi_{\boldsymbol{\theta}'}(\cdot|s_t)\\ s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}} \left[\left(\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}'}(a_t|s_t)} - 1 \right) \delta^{\varphi}_t \right] \right) - \|\mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}} - \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}\|_p \|\boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t}\|_q$$

$$= \mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}'}}(\cdot)} \left[\mathop{\mathbb{E}}_{\substack{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s)\\ a_t\sim\pi_{\boldsymbol{\theta}'}(\cdot|s_t)\\ s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}} \left[\left(\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}'}(a_t|s_t)} - 1 \right) \delta^{\varphi}_t \right] \right] - \|\mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}} - \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}'}}\|_p \|\boldsymbol{\delta}^{\varphi}_{\pi_{\boldsymbol{\theta}},t}\|_q.$$

and

$$M_{p,q,t}^{\varphi,+}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}'}) = \langle \mathbf{d}_{\pi_{\boldsymbol{\theta}'}}^{\lambda}, \boldsymbol{\Delta}_t^{\varphi}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}'})\rangle + \epsilon_{p,q,t}^{\varphi,(\lambda)}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}'}) \tag{94}$$

$$= \sum_{s\in\mathcal{S}} d_{\pi_{\boldsymbol{\theta}'}}^{\lambda}(s) \left(\mathop{\mathbb{E}}_{\substack{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s)\\ a_t\sim\pi_{\boldsymbol{\theta}'}(\cdot|s_t)\\ s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}} \left[\left(\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}'}(a_t|s_t)} - 1 \right) \delta_t^{\varphi} \right] \right) + \|\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} - \mathbf{d}_{\pi_{\boldsymbol{\theta}'}}^{\lambda}\|_p \|\boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}\|_q$$

$$= \mathbb{E}_{s\sim d_{\pi_{\boldsymbol{\theta}'}}^{\lambda}(\cdot)} \left[\mathop{\mathbb{E}}_{\substack{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(\cdot|s)\\ a_t\sim\pi_{\boldsymbol{\theta}'}(\cdot|s_t)\\ s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)}} \left[\left(\frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}'}(a_t|s_t)} - 1 \right) \delta_t^{\varphi} \right] \right] + \|\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} - \mathbf{d}_{\pi_{\boldsymbol{\theta}'}}^{\lambda}\|_p \|\boldsymbol{\delta}_{\pi_{\boldsymbol{\theta}},t}^{\varphi}\|_q.$$

According to (79) and (92), we achieve the boundedness of performance difference between two arbitrary policies $\pi_{\boldsymbol{\theta}}$ and $\pi_{\boldsymbol{\theta}'}$:

$$\underbrace{\frac{1}{1-\tilde{\gamma}} \sum_{t=0}^{\infty} \gamma^t \lambda^t M_{p,q,t}^{\varphi,-}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}'})}_{=:L_{p,q,}^{\varphi,-}} \le J(\pi_{\boldsymbol{\theta}}) - J(\pi_{\boldsymbol{\theta}'}) \le \underbrace{\frac{1}{1-\tilde{\gamma}} \sum_{t=0}^{\infty} \gamma^t \lambda^t M_{p,q,t}^{\varphi,+}(\pi_{\boldsymbol{\theta}},\pi_{\boldsymbol{\theta}'})}_{=:L_{p,q,}^{\varphi,+}}. \tag{95}$$

□

E.2 Proof of Proposition 4

Proof. (of Proposition 4).

Step 1: Rewrite the objective $J(\pi_{\boldsymbol{\theta}})$ in Eq.(73).

We rewrite the discounted distribution $\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}$ (69) as follows,

$$\boldsymbol{\rho}_0 - \frac{1}{1-\tilde{\gamma}} \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} + \frac{\tilde{\gamma}}{1-\tilde{\gamma}} \mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)} \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} = \mathbf{0}. \tag{96}$$

Let $\varphi(\cdot)$ be a real number function defined on the state space $\mathcal{S}$, i.e., $\varphi : \mathcal{S} \to \mathbb{R}$. Then we define a vector function $\boldsymbol{\phi}(\cdot) \in \mathbb{R}^{|\mathcal{S}|}$ to collect all the values $\{\varphi(s)\}_{s\in\mathcal{S}}$, and its components are

$$\boldsymbol{\phi}[s] = \varphi(s), \quad s \in \mathcal{S}.$$

Now, we take the inner product between the vector $\boldsymbol{\phi}$ and (96), we have

$$\begin{aligned} 0 &= \langle \boldsymbol{\rho}_0 - \frac{1}{1-\tilde{\gamma}} \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} + \frac{\tilde{\gamma}}{1-\tilde{\gamma}} \mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)} \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\phi} \rangle \\ &= \langle \boldsymbol{\rho}_0, \boldsymbol{\phi} \rangle - \frac{1}{1-\tilde{\gamma}} \langle \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\phi} \rangle + \frac{\tilde{\gamma}}{1-\tilde{\gamma}} \langle \mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)} \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\phi} \rangle. \end{aligned} \tag{97}$$

We express the first term $\langle \boldsymbol{\rho}_0, \boldsymbol{\phi} \rangle$ of (97) as follows,

$$\langle \boldsymbol{\rho}_0, \boldsymbol{\phi} \rangle = \sum_{s\in\mathcal{S}} \rho_0(s)\varphi(s) = \mathbb{E}_{s\sim\rho_0(\cdot)}[\varphi(s)]. \tag{98}$$

We express the second term $\langle \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\phi} \rangle$ of (97) as follows,

$$-\frac{1}{1-\tilde{\gamma}} \langle \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\phi} \rangle = -\frac{1}{1-\tilde{\gamma}} \sum_{s\in\mathcal{S}} d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s)\varphi(s) = -\frac{1}{1-\tilde{\gamma}} \mathbb{E}_{s\sim d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(\cdot)}[\varphi(s)]. \tag{99}$$

We express the third term $\langle \tilde{\gamma} \mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)} \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\phi} \rangle$ of (97) as follows,

$$\begin{aligned} \frac{\tilde{\gamma}}{1-\tilde{\gamma}} \langle \mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)} \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}, \boldsymbol{\phi} \rangle &= \frac{\tilde{\gamma}}{1-\tilde{\gamma}} \sum_{s'\in\mathcal{S}} \left(\mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)} \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} \right)[s']\varphi(s') \\ &= \frac{\tilde{\gamma}}{1-\tilde{\gamma}} \sum_{s'\in\mathcal{S}} \left(\sum_{s\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s'|s) d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s) \right) \varphi(s'). \end{aligned} \tag{100}$$

According to Lemma 1, put the results (73) and (97) together, we have

$$J(\pi_{\boldsymbol{\theta}}) \overset{(73),(97)}{=} \frac{1}{1-\tilde{\gamma}} \sum_{s\in\mathcal{S}} d^{\lambda}_{\pi_{\boldsymbol{\theta}}}(s) R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \langle \boldsymbol{\rho}_0 - \frac{1}{1-\tilde{\gamma}} \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}} + \frac{\tilde{\gamma}}{1-\tilde{\gamma}} \mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}} \mathbf{d}^{\lambda}_{\pi_{\boldsymbol{\theta}}}, \boldsymbol{\phi} \rangle$$

$$= \mathbb{E}_{s_0\sim\rho_0(\cdot)}[\varphi(s_0)] + \frac{1}{1-\tilde{\gamma}} \sum_{s\in\mathcal{S}} d^{\lambda}_{\pi_{\boldsymbol{\theta}}}(s) \left(R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \tilde{\gamma} \sum_{s'\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s'|s)\varphi(s') - \varphi(s) \right), \tag{101}$$

where the last equation holds since we unfold (97) according to (98)-(100).

Step 2: Rewrite the term $\left(R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \tilde{\gamma} \sum_{s'\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s'|s)\varphi(s') - \varphi(s) \right)$ in Eq.(101).

Then, we unfold the second term of (101) as follows,

$$R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \tilde{\gamma} \sum_{s'\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s'|s)\varphi(s') - \varphi(s) \tag{102}$$

$$\overset{(63),(65)}{=} \sum_{t=0}^{\infty} (\gamma\lambda \mathbf{P}_{\pi_{\boldsymbol{\theta}}})^t \mathbf{r}_{\pi_{\boldsymbol{\theta}}}[s] + \tilde{\gamma}(1-\gamma\lambda) \sum_{s'\in\mathcal{S}} \sum_{t=0}^{\infty} (\gamma\lambda)^t \left(\mathbf{P}^{t+1}_{\pi_{\boldsymbol{\theta}}}[s,s'] \right) \varphi(s') - \varphi(s)$$

$$\overset{(60)}{=} \sum_{t=0}^{\infty} (\gamma\lambda \mathbf{P}_{\pi_{\boldsymbol{\theta}}})^t \mathbf{r}_{\pi_{\boldsymbol{\theta}}}[s] + \gamma(1-\lambda) \sum_{s'\in\mathcal{S}} \sum_{t=0}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\varphi(s') - \varphi(s). \tag{103}$$

Recall the terms $\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}$, $\mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}[s]$ defined in (60)-(65),

$$R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \gamma(1-\lambda) \sum_{s'\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s'|s)\varphi(s') - \varphi(s) \tag{104}$$

We consider the first term $R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s)$ of (102) as follows,

$$R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) \overset{(60)-(65)}{=} \mathbf{r}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}[s] = \sum_{t=0}^{\infty} (\gamma\lambda)^t \mathbf{P}^t_{\pi_{\boldsymbol{\theta}}} \mathbf{r}_{\pi_{\boldsymbol{\theta}}}[s] = \sum_{t=0}^{\infty} \sum_{s_t\in\mathcal{S}} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s) R_{\pi_{\boldsymbol{\theta}}}(s_t). \tag{105}$$

We consider the second term $\tilde{\gamma} \sum_{s\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s'|s)\varphi(s) - \varphi(s)$ of (102) as follows,

$$\tilde{\gamma} \sum_{s'\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s'|s)\varphi(s') - \varphi(s)$$

$$\overset{(63)}{=} \tilde{\gamma}(1-\gamma\lambda) \sum_{s'\in\mathcal{S}} \sum_{t=0}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\varphi(s') - \varphi(s) \tag{106}$$

$$\overset{(60)}{=} \gamma(1-\lambda) \sum_{s'\in\mathcal{S}} \sum_{t=0}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\varphi(s') - \varphi(s) \tag{107}$$

$$= \gamma \sum_{s'\in\mathcal{S}} \sum_{t=0}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\varphi(s') - \sum_{s'\in\mathcal{S}} \underbrace{\left(\sum_{t=0}^{\infty} (\gamma\lambda)^{t+1} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\varphi(s') \right)}_{=\sum_{t=1}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t=s'|s)\varphi(s')} - \varphi(s)$$

$$= \gamma \sum_{s'\in\mathcal{S}} \sum_{t=0}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\varphi(s') - \underbrace{\left(\sum_{s'\in\mathcal{S}} \sum_{t=1}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t=s'|s)\varphi(s') + \varphi(s) \right)}_{=\sum_{s'\in\mathcal{S}} \sum_{t=0}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t=s'|s)\varphi(s')} \tag{108}$$

$$= \gamma \sum_{s'\in\mathcal{S}} \sum_{t=0}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\varphi(s') - \sum_{s_t\in\mathcal{S}} \sum_{t=0}^{\infty} (\gamma\lambda)^t \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s)\varphi(s), \tag{109}$$

where the equation from Eq.(108) to Eq.(109) holds since: according to (51), we use the following identity

$$\sum_{s^{'}\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_0 = s^{'}|s)\varphi(s^{'}) = \varphi(s).$$

Furthermore, take the result (105) and (109) to (104), we have

$$\begin{aligned}
&R^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s) + \tilde{\gamma}\sum_{s^{'}\in\mathcal{S}} \mathbb{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}(s^{'}|s)\varphi(s^{'}) - \varphi(s) \\
&= \sum_{t=0}^{\infty}(\gamma\lambda)^t \Bigg(\sum_{s_t\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s) R_{\pi_{\boldsymbol{\theta}}}(s_t) + \gamma \sum_{s^{'}\in\mathcal{S}} \underbrace{\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1} = s^{'}|s)\varphi(s^{'})}_{\overset{(52)}{=}\sum_{s_t\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s^{'}|s_t)\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s)\varphi(s^{'})} \\
&\qquad\qquad - \sum_{s_t\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s)\varphi(s_t) \Bigg) \qquad (110) \\
&= \sum_{t=0}^{\infty}(\gamma\lambda)^t \Bigg(\sum_{s_t\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s) R_{\pi_{\boldsymbol{\theta}}}(s_t) + \gamma \sum_{s_t\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s) \sum_{s_{t+1}\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}|s_t)\varphi(s_{t+1}) \\
&\qquad\qquad - \sum_{s_t\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s)\varphi(s_t) \Bigg) \qquad (111) \\
&= \sum_{t=0}^{\infty}(\gamma\lambda)^t \sum_{s_t\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s) \Bigg(\underbrace{\sum_{a_t\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a_t|s_t) \sum_{s_{t+1}\in\mathcal{S}} \mathbb{P}(s_{t+1}|s_t,a_t) r(s_{t+1}|s_t,a_t)}_{=R_{\pi_{\boldsymbol{\theta}}}(s_t)} \\
&\qquad\qquad + \gamma \underbrace{\sum_{a_t\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a_t|s_t) \sum_{s_{t+1}\in\mathcal{S}} \mathbb{P}(s_{t+1}|s_t,a_t)}_{=\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}|s_t)} \varphi(s_{t+1}) - \varphi(s_t) \Bigg) \\
&= \sum_{t=0}^{\infty}(\gamma\lambda)^t \sum_{s_t\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s) \sum_{a_t\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a_t|s_t) \sum_{s_{t+1}\in\mathcal{S}} \mathbb{P}(s_{t+1}|s_t,a_t) \left(r(s_{t+1}|s_t,a_t) + \gamma\varphi(s_{t+1}) - \varphi(s_t) \right) \\
&\qquad (112) \\
&= \sum_{t=0}^{\infty}(\gamma\lambda)^t \mathbb{E}_{s_t\sim\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(\cdot|s), a_t\sim\pi_{\boldsymbol{\theta}}(\cdot|s_t), s_{t+1}\sim\mathbb{P}(\cdot|s_t,a_t)} \left[r(s_{t+1}|s_t,a_t) + \gamma\varphi(s_{t+1}) - \varphi(s_t) \right], \qquad (113)
\end{aligned}$$

the equation from Eq.(109) to Eq.(110) holds since:

$$\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}|s) \overset{(52)}{=} \sum_{s_t\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}|s_t)\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_t|s);$$

the equation from Eq.(110) to Eq.(111) holds since we use the Markov property of the definition of MDP: for each time $t \in \mathbb{N}$,

$$\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1} = s^{'}|s_t = s) = \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s^{'}|s);$$

the equation (112) the following identity:

$$\sum_{a_t\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a_t|s_t) = 1, \qquad \sum_{s_{t+1}\in\mathcal{S}} \mathbb{P}(s_{t+1}|s_t,a_t) = 1,$$

then

$$\varphi(s_t) = \sum_{a_t \in \mathcal{A}} \pi_{\boldsymbol{\theta}}(a_t|s_t) \sum_{s_{t+1} \in \mathcal{S}} \mathbb{P}(s_{t+1}|s_t, a)\varphi(s_t).$$

Step 3: Put all the result together.

Finally, let

$$\begin{aligned}\delta_t^{\varphi} &= r(s_{t+1}|s_t, a_t) + \gamma\varphi(s_{t+1}) - \varphi(s_t),\\ \delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s) &= \mathbb{E}_{s_t \sim \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(\cdot|s), a_t \sim \pi_{\boldsymbol{\theta}}(\cdot|s_t), s_{t+1} \sim \mathbb{P}(\cdot|s_t, a_t)} \left[\delta_t^{\varphi}\right],\end{aligned}$$

combining the results (101) and (113), we have

$$\begin{aligned}J(\pi_{\boldsymbol{\theta}}) =& \mathbb{E}_{s_0 \sim \rho_0(\cdot)}[\varphi(s_0)] + \frac{1}{1-\tilde{\gamma}} \sum_{s \in \mathcal{S}} d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s) \left(\sum_{t=0}^{\infty} \gamma^t \lambda^t \delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s) \right) \\ =& \mathbb{E}_{s_0 \sim \rho_0(\cdot)}[\varphi(s_0)] + \frac{1}{1-\tilde{\gamma}} \mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(\cdot)} \left[\sum_{t=0}^{\infty} \gamma^t \lambda^t \delta_{\pi_{\boldsymbol{\theta}},t}^{\varphi}(s) \right].\end{aligned} \tag{114}$$

This concludes the proof of Proposition 4. □

E.3 Proposition 3

All above bound results appear in (11) and (13) can be extended for a total variational divergence to KL-divergence between policies, which are desirable for policy optimization.

We obtain

$$\mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}'}}^{\lambda}(\cdot)} \left[D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s]\right] \leq \mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}'}}^{\lambda}(\cdot)} \left[\sqrt{\frac{1}{2}\mathrm{KL}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s]}\right] \leq \sqrt{\frac{1}{2}\mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}'}}^{\lambda}(\cdot)} \left[\mathrm{KL}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s]\right]}, \tag{115}$$

where $\mathrm{KL}(\cdot, \cdot)$ is KL-divergence, and

$$\mathrm{KL}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s] = \mathrm{KL}(\pi_{\boldsymbol{\theta}'}(\cdot|s), \pi_{\boldsymbol{\theta}}(\cdot|s));$$

the first inequality follows Pinsker's inequality [Csiszár and Körner, 2011] and the second inequality follows Jensen's inequality. According to (115), we obtain the next Proposition 3.

Proposition 3. *All the bounds in (11) and (13) hold if we make the following substitution:*

$$\mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}'}}^{\lambda}(\cdot)} \left[D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s]\right] \leftarrow \sqrt{\frac{1}{2}\mathbb{E}_{s \sim d_{\pi_{\boldsymbol{\theta}'}}^{\lambda}(\cdot)} \left[\mathrm{KL}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s]\right]}.$$

F Lemma 2

In this section, we show Lemma 2 that presents an upper bound to the difference between two λ-version of normalized discounted distribution. Before we present our main results, we review the norms induced by p-norms for matrix.

F.1 Norms Induced by p-norms for Matrix

If the p-norm for vectors $(1 \leq p \leq \infty)$ is used for both spaces $\mathbb{R}^n$ and $\mathbb{R}^m$, then the corresponding operator norm is:

$$\|\mathbf{A}\|_p = \sup_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{A}\mathbf{x}\|_p}{\|\mathbf{x}\|_p}.$$

These induced norms are different from the "entry-wise" p-norms and the *Schatten* p-norms for matrices treated below, which are also usually denoted by $\|\mathbf{A}\|_p$.

In the special cases of $p = 1$ and $p = \infty$, the induced matrix norms can be computed or estimated by

$$\|\mathbf{A}\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^{m} |a_{ij}|,$$

which is simply the maximum absolute column sum of the matrix;

$$\|\mathbf{A}\|_\infty = \max_{1 \leq i \leq m} \sum_{j=1}^{n} |a_{ij}|,$$

which is simply the maximum absolute row sum of the matrix. Thus, the following equation holds

$$\|\mathbf{A}^\top\|_\infty = \|\mathbf{A}\|_1. \tag{116}$$

F.2 Lemma 2

Lemma 2. *The divergence between discounted future state visitation distributions,* $\|\mathbf{d}^\lambda_{\pi_{\boldsymbol{\theta}'}} - \mathbf{d}^\lambda_{\pi_{\boldsymbol{\theta}}}\|_1$ *is bounded as follows,*

$$\|\mathbf{d}^\lambda_{\pi_{\boldsymbol{\theta}'}} - \mathbf{d}^\lambda_{\pi_{\boldsymbol{\theta}}}\|_1 \leq \frac{1}{1-\tilde{\gamma}} \cdot \frac{\tilde{\gamma}\left(\gamma\lambda(|\mathcal{S}|-1)+1\right)}{1-\gamma\lambda} \mathbb{E}_{s \sim d^\lambda_{\pi_{\boldsymbol{\theta}}}(\cdot)} \left[2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s]\right],$$

where $D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s]$ *is the total variational divergence between action distributions at state* s*, i.e.,*

$$2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s] = \sum_{a \in \mathcal{A}} \left|\pi_{\boldsymbol{\theta}'}(a|s) - \pi_{\boldsymbol{\theta}}(a|s)\right|.$$

Proof. (of Lemma 2). Recall Eq.(69), we know,

$$\mathbf{d}^\lambda_{\pi_{\boldsymbol{\theta}}} = (1-\tilde{\gamma}) \sum_{t=0}^{\infty} \left(\gamma \mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right)^t \boldsymbol{\rho}_0 = (1-\tilde{\gamma}) \left(\mathbf{I} - \tilde{\gamma}\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right)^{-1} \boldsymbol{\rho}_0.$$

To short the expression, we introduce some additional notations as follows.

$$\mathbf{G}_{\pi_{\boldsymbol{\theta}}} = \left(\mathbf{I} - \tilde{\gamma}\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right)^{-1},\ \mathbf{G}_{\pi_{\boldsymbol{\theta}'}} = \left(\mathbf{I} - \tilde{\gamma}\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}'}}\right)^{-1},\ \mathbf{D} = \mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}. \tag{117}$$

Then, after some simple algebra, the following holds

$$\mathbf{G}^{-1}_{\pi_{\boldsymbol{\theta}}} - \mathbf{G}^{-1}_{\pi_{\boldsymbol{\theta}'}} = \left(\mathbf{I} - \tilde{\gamma}\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}}}\right) - \left(\mathbf{I} - \tilde{\gamma}\mathbf{P}^{(\lambda)}_{\pi_{\boldsymbol{\theta}'}}\right) = \tilde{\gamma}\mathbf{D}. \tag{118}$$

Furthermore, by left-multiplying by $\mathbf{G}_{\pi_{\boldsymbol{\theta}}}$ and right-multiplying by $\mathbf{G}_{\pi_{\boldsymbol{\theta}'}}$, we achieve

$$\mathbf{G}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{G}_{\pi_{\boldsymbol{\theta}}} = \tilde{\gamma}\mathbf{G}_{\pi_{\boldsymbol{\theta}'}}\mathbf{D}\mathbf{G}_{\pi_{\boldsymbol{\theta}}}. \tag{119}$$

Grouping all the results from (117)-(119), recall (69),

$$\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} = (1-\tilde{\gamma})\sum_{t=0}^{\infty}\left(\gamma \mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}\right)^{t}\boldsymbol{\rho}_0 = (1-\tilde{\gamma})\left(\mathbf{I}-\tilde{\gamma}\mathbf{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}\right)^{-1}\boldsymbol{\rho}_0 = (1-\tilde{\gamma})\mathbf{G}_{\pi_{\boldsymbol{\theta}}}\boldsymbol{\rho}_0, \tag{120}$$

then we have

$$\begin{aligned}\mathbf{d}_{\pi_{\boldsymbol{\theta}'}}^{\lambda} - \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda} =& (1-\tilde{\gamma})\left(\mathbf{G}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{G}_{\pi_{\boldsymbol{\theta}}}\right)\boldsymbol{\rho}_0 \\ \overset{(119)}{=}& (1-\tilde{\gamma})\tilde{\gamma}\mathbf{G}_{\pi_{\boldsymbol{\theta}'}}\mathbf{D}\mathbf{G}_{\pi_{\boldsymbol{\theta}}}\boldsymbol{\rho}_0 \\ \overset{(120)}{=}& \tilde{\gamma}\mathbf{G}_{\pi_{\boldsymbol{\theta}'}}\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}.\end{aligned} \tag{121}$$

Applying (121), we have

$$\|\mathbf{d}_{\pi_{\boldsymbol{\theta}'}}^{\lambda} - \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1 \overset{(121)}{\leq} \tilde{\gamma}\|\mathbf{G}_{\pi_{\boldsymbol{\theta}'}}\|_1\|\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1. \tag{122}$$

Firstly, we bound the term $\|\mathbf{G}_{\pi_{\boldsymbol{\theta}'}}\|_1$ as follows,

$$\|\mathbf{G}_{\pi_{\boldsymbol{\theta}'}}\|_1 = \left\|\left(\mathbf{I}-\tilde{\gamma}\mathbf{P}_{\pi_{\boldsymbol{\theta}'}}^{(\lambda)}\right)^{-1}\right\|_1 \leq \sum_{t=0}^{\infty}\tilde{\gamma}^t\left\|\mathbf{P}_{\pi_{\boldsymbol{\theta}'}}^{(\lambda)}\right\|_1 = \frac{1}{1-\tilde{\gamma}}. \tag{123}$$

Thus, recall $\tilde{\gamma} = \dfrac{\gamma(1-\lambda)}{1-\gamma\lambda}$, we obtain

$$\|\mathbf{d}_{\pi_{\boldsymbol{\theta}'}}^{\lambda} - \mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1 \leq \tilde{\gamma}\|\mathbf{G}_{\pi_{\boldsymbol{\theta}'}}\|_1\|\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1 \leq \frac{\tilde{\gamma}}{1-\tilde{\gamma}}\|\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1 \tag{124}$$

$$\leq \frac{1}{1-\tilde{\gamma}}\cdot\frac{\tilde{\gamma}\left(\gamma\lambda(|\mathcal{S}|-1)+1\right)}{1-\gamma\lambda}\mathbb{E}_{s\sim d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(\cdot)}\left[2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\right], \tag{125}$$

where the last equation holds due to Lemma 3, this concludes the proof of Lemma 2 . □

Lemma 3. *The term $\|\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1$ is bounded as follows,*

$$\|\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1 \leq \frac{\gamma\lambda(|\mathcal{S}|-1)+1}{1-\gamma\lambda}\mathbb{E}_{s\sim d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(\cdot)}\left[2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\right].$$

Proof. Now, we analyze $\|\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1$ as follows,

$$\begin{aligned}\|\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1 &= \sum_{s\in\mathcal{S}}\left|\sum_{s'\in\mathcal{S}}\mathbf{D}(s'|s)d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s)\right| \overset{(63)}{=} \sum_{s\in\mathcal{S}}\left|\sum_{s'\in\mathcal{S}}\left(\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}^{(\lambda)}(s'|s) - \mathbb{P}_{\pi_{\boldsymbol{\theta}}}^{(\lambda)}(s'|s)\right)\right| d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s) \\ &\overset{(63)}{=} \sum_{s\in\mathcal{S}}\left|(1-\gamma\lambda)\sum_{t=0}^{\infty}(\gamma\lambda)^t\sum_{s'\in\mathcal{S}}\left(\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_{t+1}=s'|s) - \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\right)\right| d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s),\end{aligned}$$

which implies that to bound $\|\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1$, we need to bound the following difference

$$\sum_{t=0}^{\infty}(\gamma\lambda)^t\sum_{s'\in\mathcal{S}}\left(\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_{t+1}=s'|s) - \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\right).$$

Step 1: Rewrite $\sum_{t=0}^{\infty}(\gamma\lambda)^t\sum_{s'\in\mathcal{S}}\left(\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_{t+1}=s'|s) - \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s)\right)$.

Let $s_0 = s$, then

$$\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s) \overset{(52)}{=} \sum_{s_1\in\mathcal{S}}\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s_1)\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_1|s_0) \tag{126}$$

$$= \sum_{s_1\in\mathcal{S}}\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s'|s_1)\left(\sum_{a\in\mathcal{A}}\pi_{\boldsymbol{\theta}}(a|s_0)\mathbb{P}(s_1|s_0,a)\right). \tag{127}$$

Similarly,

$$\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_{t+1}=s^{'}|s) \overset{(52)}{=} \sum_{s_1\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_{t+1}=s^{'}|s_1)\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_1|s_0) \tag{128}$$

$$= \sum_{s_1\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_{t+1}=s^{'}|s_1)\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}'}(a|s_0)\mathbb{P}(s_1|s_0,a)\right). \tag{129}$$

Firstly, we consider the following term

$$\begin{aligned}
\sum_{t=0}^{\infty}(\gamma\lambda)^t \sum_{s'\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s^{'}|s) &= \sum_{t=0}^{\infty}(\gamma\lambda)^t \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s^{'}|s_1)\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right)\\
&= \sum_{t=0}^{\infty}(\gamma\lambda)^t \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbf{P}^t_{\pi_{\boldsymbol{\theta}}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right)\\
&= \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \left(\sum_{t=0}^{\infty}(\gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}})^t\right)[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right)\\
&= \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \left(\mathbf{I}-\gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)^{-1}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right).
\end{aligned} \tag{130}$$

To short expression, we introduce a new notation as follows,

$$\mathbf{F}_{\pi_{\boldsymbol{\theta}}} = \left(\mathbf{I}-\gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)^{-1}. \tag{131}$$

Then, we rewrite (130) as follows,

$$\sum_{t=0}^{\infty}(\gamma\lambda)^t \sum_{s'\in\mathcal{S}} \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s^{'}|s) = \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbf{F}_{\pi_{\boldsymbol{\theta}}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right). \tag{132}$$

Furthermore, according to (132), we obtain

$$\begin{aligned}
&\sum_{t=0}^{\infty}(\gamma\lambda)^t \sum_{s'\in\mathcal{S}} \left(\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_{t+1}=s^{'}|s) - \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s^{'}|s)\right)\\
&= \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}'}(a|s)\mathbb{P}(s_1|s,a)\right) - \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbf{F}_{\pi_{\boldsymbol{\theta}}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right)\\
&= \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}'}(a|s)\mathbb{P}(s_1|s,a)\right) - \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right)\\
&\quad + \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right) - \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbf{F}_{\pi_{\boldsymbol{\theta}}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right)\\
&= \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}} \left(\pi_{\boldsymbol{\theta}'}(a|s) - \pi_{\boldsymbol{\theta}}(a|s)\right)\mathbb{P}(s_1|s,a)\right) \qquad (133)\\
&\quad + \sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}} \left(\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s_1,s^{'}] - \mathbf{F}_{\pi_{\boldsymbol{\theta}}}[s_1,s^{'}]\right)\left(\sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right), \qquad (134)
\end{aligned}$$

which implies that to bound the following difference

$$\sum_{t=0}^{\infty}(\gamma\lambda)^t \sum_{s'\in\mathcal{S}} \left(\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_{t+1}=s^{'}|s) - \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s^{'}|s)\right),$$

we need to bound (133) and (134).

Step 2: Bound the difference (134).

Due to the simple fact: for any inverse matrices $\mathbf{A}$ and $\mathbf{B}$, then the following identity holds

$$\mathbf{A}^{-1} - \mathbf{B}^{-1} = \mathbf{A}^{-1}(\mathbf{B} - \mathbf{A})\mathbf{B}^{-1},$$

we rewrite the difference $\mathbf{F}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{F}_{\pi_{\boldsymbol{\theta}}}$ as follows,

$$\begin{aligned}\mathbf{F}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{F}_{\pi_{\boldsymbol{\theta}}} &= \left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}'}}\right)^{-1} - \left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)^{-1} \\ &= \gamma\lambda\left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}'}}\right)^{-1}\left(\mathbf{P}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)\left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)^{-1}.\end{aligned}$$

Then, we rewrite (134) as the following matrix version

$$\begin{aligned}&\sum_{s'\in\mathcal{S}}\sum_{s_1\in\mathcal{S}}\left(\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s_1, s'] - \mathbf{F}_{\pi_{\boldsymbol{\theta}}}[s_1, s']\right)\left(\sum_{a\in\mathcal{A}}\pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right) \\ =&\sum_{s''\in\mathcal{S}}\sum_{s'\in\mathcal{S}}\left(\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s', s''] - \mathbf{F}_{\pi_{\boldsymbol{\theta}}}[s', s'']\right)\left(\sum_{a\in\mathcal{A}}\pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s'|s,a)\right) = \left\|\left(\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}^{\top} - \mathbf{F}_{\pi_{\boldsymbol{\theta}}}^{\top}\right)\mathbf{p}_{\pi_{\boldsymbol{\theta}}}(s)\right\|_1,\end{aligned}$$

where $\mathbf{p}_{\pi_{\boldsymbol{\theta}}}(s) \in \mathbb{R}^{|\mathcal{S}|}$, and

$$\mathbf{p}_{\pi_{\boldsymbol{\theta}}}(s) = \left(\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_1|s), \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_2|s), \cdots, \mathbb{P}_{\pi_{\boldsymbol{\theta}}}\left(s_{|\mathcal{S}|}|s\right)\right)^{\top}.$$

According to (116), we obtain

$$\begin{aligned}&\left\|\left(\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}^{\top} - \mathbf{F}_{\pi_{\boldsymbol{\theta}}}^{\top}\right)\mathbf{p}_{\pi_{\boldsymbol{\theta}}}(s)\right\|_1 = \left\|\mathbf{p}_{\pi_{\boldsymbol{\theta}}}^{\top}(s)\left(\mathbf{F}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{F}_{\pi_{\boldsymbol{\theta}}}\right)\right\|_\infty \\ =&\gamma\lambda\left\|\mathbf{p}_{\pi_{\boldsymbol{\theta}}}^{\top}(s)\left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}'}}\right)^{-1}\left(\mathbf{P}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)\left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)^{-1}\right\|_\infty \\ =&\frac{\gamma\lambda}{1-\gamma\lambda}\left\|\underbrace{\mathbf{p}_{\pi_{\boldsymbol{\theta}}}^{\top}(s)(1-\gamma\lambda)\left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}'}}\right)^{-1}}_{\mathbf{f}_s^{\top}}\left(\mathbf{P}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)\left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)^{-1}\right\|_\infty \qquad (135) \\ \leq&\frac{\gamma\lambda}{1-\gamma\lambda}\left\|\mathbf{f}_s^{\top}\left(\mathbf{P}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)\right\|_\infty\left\|\left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)^{-1}\right\|_\infty \\ =&\frac{\gamma\lambda}{(1-\gamma\lambda)^2}\left\|\mathbf{f}_s^{\top}\left(\mathbf{P}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)\right\|_\infty \qquad (136) \\ =&\frac{2\gamma\lambda}{(1-\gamma\lambda)^2}\sum_{s\in\mathcal{S}}D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'}, \pi_{\boldsymbol{\theta}})[s], \qquad (137)\end{aligned}$$

where in Eq.(135), we introduce a notation $\mathbf{f}_s^{\top} \in \mathbb{R}^{|\mathcal{S}|}$ as follows,

$$\mathbf{f}_s^{\top} =: \mathbf{p}_{\pi_{\boldsymbol{\theta}}}^{\top}(s)(1-\gamma\lambda)\left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}'}}\right)^{-1};$$

Eq.(136) holds since:

$$\left\|\left(\mathbf{I} - \gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)^{-1}\right\|_\infty = \frac{1}{1-\gamma\lambda};$$

Eq.(137) holds since:

$$\begin{aligned}\left\|\mathbf{f}^{\top}\left(\mathbf{P}_{\pi_{\boldsymbol{\theta}'}} - \mathbf{P}_{\pi_{\boldsymbol{\theta}}}\right)\right\|_\infty &= \sum_{s\in\mathcal{S}}\sum_{s'\in\mathcal{S}}\mathbf{f}_s[s']\left|\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s'|s) - \mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s'|s)\right| \\ &= \sum_{s\in\mathcal{S}}\sum_{s'\in\mathcal{S}}\mathbf{f}_s[s']\left|\sum_{a\in\mathcal{A}}\mathbb{P}(s'|s,a)\left(\pi_{\boldsymbol{\theta}'}(a|s) - \pi_{\boldsymbol{\theta}}(a|s)\right)\right| \\ &\leq \sum_{s\in\mathcal{S}}\sum_{s'\in\mathcal{S}}\mathbf{f}_s[s']\sum_{a\in\mathcal{A}}\left|\pi_{\boldsymbol{\theta}'}(a|s) - \pi_{\boldsymbol{\theta}}(a|s)\right| \\ &= \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}}\left|\pi_{\boldsymbol{\theta}'}(a|s) - \pi_{\boldsymbol{\theta}}(a|s)\right|,\end{aligned}$$

where the last equation holds due to the following fact

$$\sum_{s^{'}\in\mathcal{S}} \mathbf{f}_s[s^{'}] = \sum_{s^{'}\in\mathcal{S}} \mathbf{p}_{\pi_{\boldsymbol{\theta}}}^{\top}(s)(1-\gamma\lambda)\left(\mathbf{I}-\gamma\lambda\mathbf{P}_{\pi_{\boldsymbol{\theta}'}}\right)^{-1}[s^{'}] = 1.$$

Thus, the difference (134) is bounded as follows,

$$\sum_{s^{'}\in\mathcal{S}}\sum_{s_1\in\mathcal{S}}\left(\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s_1,s^{'}]-\mathbf{F}_{\pi_{\boldsymbol{\theta}}}[s_1,s^{'}]\right)\left(\sum_{a\in\mathcal{A}}\pi_{\boldsymbol{\theta}}(a|s)\mathbb{P}(s_1|s,a)\right) \leq \frac{2\gamma\lambda}{(1-\gamma\lambda)^2}\sum_{s\in\mathcal{S}} D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s].$$

Step 3: Bound the difference (133).

We turn to consider (133):

$$\begin{aligned}
&\sum_{s^{'}\in\mathcal{S}}\sum_{s_1\in\mathcal{S}}\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s_1,s^{'}]\left(\sum_{a\in\mathcal{A}}\left(\pi_{\boldsymbol{\theta}'}(a|s)-\pi_{\boldsymbol{\theta}}(a|s)\right)\mathbb{P}(s_1|s,a)\right)\\
=&\sum_{s^{''}\in\mathcal{S}}\sum_{s^{'}\in\mathcal{S}}\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}[s^{'},s^{''}]\left(\sum_{a\in\mathcal{A}}\left(\pi_{\boldsymbol{\theta}'}(a|s)-\pi_{\boldsymbol{\theta}}(a|s)\right)\mathbb{P}(s^{'}|s,a)\right)\\
=&\left\|\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}^{\top}\left(\mathbf{p}_{\pi_{\boldsymbol{\theta}'}}(s)-\mathbf{p}_{\pi_{\boldsymbol{\theta}}}(s)\right)\right\|_1\\
\leq&\left\|\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}^{\top}\right\|_1\left\|\mathbf{p}_{\pi_{\boldsymbol{\theta}'}}(s)-\mathbf{p}_{\pi_{\boldsymbol{\theta}}}(s)\right\|_1\\
=&\left\|\mathbf{p}_{\pi_{\boldsymbol{\theta}'}}(s)-\mathbf{p}_{\pi_{\boldsymbol{\theta}}}(s)\right\|_1 \leq \frac{2}{1-\gamma\lambda}D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s],
\end{aligned} \tag{138}$$

where the last Eq.(138) holds since:

$$\left\|\mathbf{F}_{\pi_{\boldsymbol{\theta}'}}^{\top}\right\|_1 = \frac{1}{1-\gamma\lambda},$$

and

$$\begin{aligned}
\left\|\mathbf{p}_{\pi_{\boldsymbol{\theta}'}}(s)-\mathbf{p}_{\pi_{\boldsymbol{\theta}}}(s)\right\|_1 &= \sum_{s^{'}\in\mathcal{S}}\left|\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s^{'}|s)-\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s^{'}|s)\right|\\
&= \sum_{s^{'}\in\mathcal{S}}\left|\sum_{a\in\mathcal{A}}\mathbb{P}(s^{'}|s,a)\left(\pi_{\boldsymbol{\theta}'}(a|s)-\pi_{\boldsymbol{\theta}}(a|s)\right)\right|\\
&\leq \sum_{a\in\mathcal{A}}\sum_{s^{'}\in\mathcal{S}}\mathbb{P}(s^{'}|s,a)\left|\pi_{\boldsymbol{\theta}'}(a|s)-\pi_{\boldsymbol{\theta}}(a|s)\right|\\
&= \sum_{a\in\mathcal{A}}\left|\pi_{\boldsymbol{\theta}'}(a|s)-\pi_{\boldsymbol{\theta}}(a|s)\right| = 2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s].
\end{aligned}$$

Step 4: Put all the result together.

Finally, according to (133), (134), (137), and (138), we obtain

$$\begin{aligned}
\|\mathbf{D}\mathbf{d}_{\pi_{\boldsymbol{\theta}}}^{\lambda}\|_1 =& \sum_{s\in\mathcal{S}}\left|(1-\gamma\lambda)\sum_{t=0}^{\infty}(\gamma\lambda)^t\sum_{s^{'}\in\mathcal{S}}\left(\mathbb{P}_{\pi_{\boldsymbol{\theta}'}}(s_{t+1}=s^{'}|s)-\mathbb{P}_{\pi_{\boldsymbol{\theta}}}(s_{t+1}=s^{'}|s)\right)\right| d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s)\\
\leq& \sum_{s\in\mathcal{S}} d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s)\left[\frac{2\gamma\lambda}{1-\gamma\lambda}\sum_{s\in\mathcal{S}}D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]+2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\right]\\
=& \sum_{s\in\mathcal{S}} d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(s)\left[\frac{2\gamma\lambda|\mathcal{S}|}{1-\gamma\lambda}D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]+2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\right]\\
=& \frac{\gamma\lambda(|\mathcal{S}|-1)+1}{1-\gamma\lambda}\mathbb{E}_{s\sim d_{\pi_{\boldsymbol{\theta}}}^{\lambda}(\cdot)}\left[2D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}'},\pi_{\boldsymbol{\theta}})[s]\right].
\end{aligned}$$

This concludes the result of Lemma 3. □

G Proof of Theorem 2

Before we present the main result, we define some notations.

$$\chi_k = \mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}\left(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)[s]\right], \tag{139}$$

$$\iota = \frac{\tilde{\gamma}\left(\gamma\lambda(|\mathcal{S}|-1)+1\right)}{(1-\tilde{\gamma})(1-\gamma\lambda)}. \tag{140}$$

Proof. (of Theorem 2)

According to Bregman divergence, if policy $\pi_{\boldsymbol{\theta}_k}$ is feasible, policy $\pi_{\boldsymbol{\theta}_{k+1}}$ is generated according to (15), then the following

$$\mathrm{KL}\left(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right) \geq \mathrm{KL}\left(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+1}}\right) + \mathrm{KL}\left(\pi_{\boldsymbol{\theta}_{k+1}}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)$$

implies

$$\chi_k = \mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}\left(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}\right)[s]\right] \geq \mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}\left(\pi_{\boldsymbol{\theta}_{k+1}}, \pi_{\boldsymbol{\theta}_k}\right)[s]\right].$$

According to the asymptotically symmetry of KL divergence if we update the policy within a local region, then, we have

$$\chi_k \geq \mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}\left(\pi_{\boldsymbol{\theta}_{k+\frac{1}{2}}}, \pi_{\boldsymbol{\theta}_k}\right)[s]\right] \geq \mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}\left(\pi_{\boldsymbol{\theta}_{k+1}}, \pi_{\boldsymbol{\theta}_k}\right)[s]\right].$$

Furthermore, according to Proposition 1 and Proposition 3, we have

$$\begin{aligned}
&J(\pi_{\boldsymbol{\theta}_{k+1}}) - J(\pi_{\boldsymbol{\theta}_k}) \\
\geq& \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}_{k+1}}(\cdot|s)}\left[A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k}}(s,a) - \iota\epsilon^{V}_{\pi_{\boldsymbol{\theta}_{k+1}}}(\pi_{\boldsymbol{\theta}_k}) D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+1}})[s]\right] \\
\geq& \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}_{k+1}}(\cdot|s)}\left[-\iota\alpha_k\epsilon^{V}_{\pi_{\boldsymbol{\theta}_{k+1}}}(\pi_{\boldsymbol{\theta}_k})\sqrt{\frac{1}{2}\mathrm{KL}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+1}})[s]}\right] \\
\geq& -\frac{\iota}{1-\tilde{\gamma}}\alpha_k\sqrt{2\chi_k}\epsilon^{V}_{\pi_{\boldsymbol{\theta}_{k+1}}}(\pi_{\boldsymbol{\theta}_k}).
\end{aligned}$$

Similarly, according to Proposition 1 and Proposition 2, and since policy $\pi_{\boldsymbol{\theta}_{k+1}}$ satisfies

$$J^c(\pi_{\boldsymbol{\theta}_k}) + \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}_{k+1}}(\cdot|s)}\left[A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k},C}(s,a)\right] + \beta_k\sqrt{\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+1}})[s]\right]} \leq b, \tag{141}$$

and

$$\begin{aligned}
&J^c(\pi_{\boldsymbol{\theta}_{k+1}}) - J^c(\pi_{\boldsymbol{\theta}_k}) \qquad (142)\\
\leq& \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}_{k+1}}(\cdot|s)}\left[A^{\mathtt{GAE}(\gamma,\lambda)}_{\pi_{\boldsymbol{\theta}_k},C}(s,a) + \iota\beta_k\epsilon^{C}_{\pi_{\boldsymbol{\theta}_{k+1}}} D_{\mathrm{TV}}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+1}})[s]\right].
\end{aligned}$$

Combining (141)- (143), we have

$$\begin{aligned}
&J^c(\pi_{\boldsymbol{\theta}_{k+1}}) - J^c(\pi_{\boldsymbol{\theta}_k}) \qquad (143)\\
\leq& b + \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}_{k+1}}(\cdot|s)}\left[\iota\beta_k\epsilon^{C}_{\pi_{\boldsymbol{\theta}_{k+1}}}\sqrt{\frac{1}{2}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[\mathrm{KL}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}_{k+1}})[s]\right]}\right] \\
\leq& b + \frac{1}{1-\tilde{\gamma}}\mathbb{E}_{s\sim d^{\lambda}_{\pi_{\boldsymbol{\theta}_k}}(\cdot), a\sim\pi_{\boldsymbol{\theta}_{k+1}}(\cdot|s)}\left[\iota\beta_k\sqrt{2\chi_k}\epsilon^{C}_{\pi_{\boldsymbol{\theta}_{k+1}}}\right]. \qquad (144)
\end{aligned}$$

□

H Experiments

The Python code for our implementation of CUP is provided along with this submission in the supplementary material.

All experiments were implemented in Pytorch 1.7.0 with CUDA 11.0 and conducted on an Ubuntu 20.04.2 LTS (GNU/Linux 5.8.0-59-generic x86 64) with 40 CPU cores (Intel(R) Xeon(R) Silver 4210R CPU @ 2.40GHz), 251G memory and 4 GPU cards (GeForce RTX 3080). The baseline algorithm FOCOPS based on the open-source `https://github.com/ymzhang01/focops`, which were offical code library. The other baseline algorithms include CPO, TRPO-L, PPO-L based on `https://github.com/openai/safety-starter-agents`, which published by openai.

H.1 Algorithm Parameters

Hyperparameter	CUP	PPO-L	TRPO-L	CPO	FOCOPS
No. of hidden layers	2	2	2	2	2
No. of hidden nodes	64	64	64	64	64
Activation	$\tanh$	$\tanh$	$\tanh$	$\tanh$	$\tanh$
Initial log std	-0.5	-0.5	-1	-0.5	-0.5
Discount for reward γ	0.99	0.99	0.99	0.99	0.99
Discount for cost γ_C	0.99	0.99	0.99	0.99	0.99
Batch size	5000	5000	5000	5000	5000
Minibatch size	64	64	N/A	N/A	64
No. of optimization epochs	10	10	N/A	N/A	10
Maximum episode length	1000	1000	1000	1000	1000
GAE parameter (reward)	0.95	0.95	0.95	0.95	0.95
GAE parameter (cost)	0.95	0.95	0.95	0.95	0.95
Learning rate for policy	3×10^{-4}	3×10^{-4}	N/A	N/A	3×10^{-4}
Learning rate for reward value net	3×10^{-4}	3×10^{-4}	3×10^{-4}	3×10^{-4}	3×10^{-4}
Learning rate for cost value net	3×10^{-4}	3×10^{-4}	3×10^{-4}	3×10^{-4}	3×10^{-4}
Learning rate for ν	0.01	0.01	0.01	N/A	0.01
$L2$-regularization coeff. for value net	10^{-3}	3×10^{-3}	3×10^{-3}	3×10^{-3}	10^{-3}
Clipping coefficient	N/A	0.2	N/A	N/A	N/A
Damping coeff.	N/A	N/A	0.01	0.01	N/A
Backtracking coeff.	N/A	N/A	0.8	0.8	N/A
Max backtracking iterations	N/A	N/A	10	10	N/A
Max conjugate gradient iterations	N/A	N/A	10	10	N/A
Iterations for training value net	1	1	80	80	1
Temperature λ	1.5	N/A	N/A	N/A	1.5
Trust region bound δ	0.02	N/A	0.01	0.01	0.02
Initial ν, $\nu_{\max}$	0, 2	0, 1	0, 2	N/A	0, 2

Table 3: Hyper-parameters for robots.

H.2 Environment

H.2.1 Environment 1: Robots with Speed Limit.

We consider two tasks from MuJoCo [Brockman *et al.*, 2016]: Walker2d-v3 and Hopper-v3, where the setting of cost follows [Zhang *et al.*, 2020]. For agents move on a two-dimensional plane, the cost is calculated as follows,

$$C(s, a) = \sqrt{v_x^2 + v_y^2};$$

for agents move along a straight line, the cost is calculated as

$$C(s, a) = |v_x|,$$

where v_x, v_y are the velocities of the agent in the x and y directions respectively.

H.2.2 Environment 2: Circle.

The Circle Environment follows [Achiam *et al.*, 2017], and we use open-source implementation of the circle environments from `https://github.com/ymzhang01/mujoco-circle`. According to Zhang *et al.* [2020], those experiments were implemented in OpenAI Gym [Brockman *et al.*, 2016] while the circle tasks in Achiam *et al.* [2017] were implemented in rllab [Duan *et al.*, 2016]. We also excluded the Point agent from the original experiments since it is not a valid agent in OpenAI Gym. The first two dimensions in the state space are the (x, y) coordinates of the center mass of the agent, hence the state space for both agents has two extra dimensions compared to the standard Ant-v0 and Humanoid-v0 environments from OpenAI Gym.

Now, we present some necessary details of this environment taken from [Zhang *et al.*, 2020].

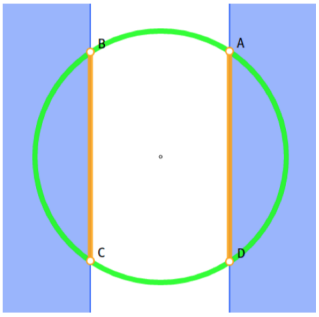


Figure 4: In the Circle task, reward is maximized by moving along the green circle. The agent is not allowed to enter the blue regions, so its optimal constrained path follows the line segments AD and BC (figure and caption taken from [Achiam *et al.*, 2017; Zhang *et al.*, 2020]).

In the circle tasks, the goal is for an agent to move along the circumference of a circle while remaining within a safety region smaller than the radius of the circle. The exact geometry of the task is shown in Figure 4. The reward and cost functions are defined as:

$$R(s) = \frac{-yv_x + xv_y}{1 + |\sqrt{x^2 + y^2} - r|}, \quad C(s) = \mathbb{I}(|x| > x_{\text{lim}}),$$

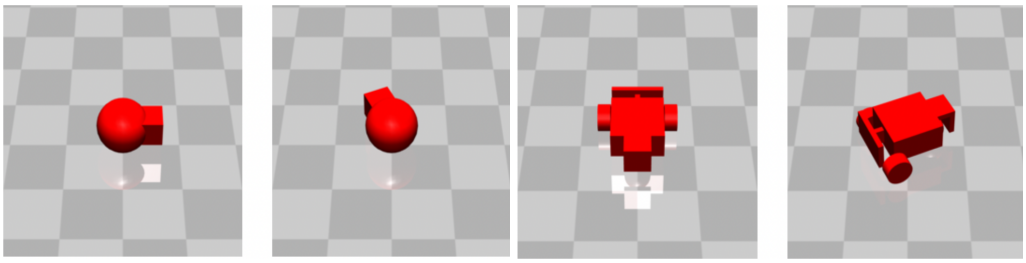
where x, y are the positions of the agent on the plane, v_x, v_y are the velocities of the agent along the x and y directions, r is the radius of the circle, and x_{lim} specifies the range of the safety region. The radius is set to $r = 10$ for both Ant and Humanoid while x_{lim} is set to 3 and 2.5 for Ant and Humanoid respectively. Note that these settings are identical to those of the circle task in Achiam *et al.* [2017]; Zhang *et al.* [2020].

H.3 Safety Gym

In Safety Gym environments, the agent perceives the world through a robot's sensors and interacts with the world through its actuators [Ray *et al.*, 2019]. In this section, we consider two robots: Point and Car, where the presentation of those safety environments are taken from [Ray *et al.*, 2019], for more details, please refer to [Ray *et al.*, 2019, Page 8–10]. In this section, we experiment with the Safety Gym environment-builder two tasks: Goal, Button.

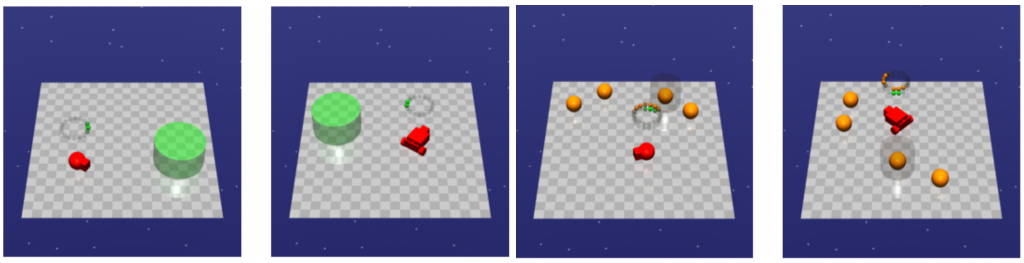
H.3.1 Safety Gym Robots

We consider two robots: Point and Car. All actions for all robots are continuous and linearly scaled to $[-1, +1]$, which is typical for 3D robot-based RL environments and (anecdotally) improves learning with neural nets. Modulo scaling, the action parameterization is based on a mix of hand-tuning and MuJoCo actuator defaults, and we caution that it is not clear if these choices are optimal. Some safe exploration techniques are action-layer interventions, like projecting to the closest predicted safe action [**?**], and these methods can be sensitive to action parameterization. As a result, action parameterization may merit more careful consideration than is usually given. Future work on action



(a) Point (b) Car

Figure 5: Fig (a): a 2D robot that can turn and move; Fig (b): a wheeled robot with a differential drive control, in "Button", the objective is to press the highlighted button (visually indicated with a faint gray cylinder), where figures and caption taken from Safety Gym [Ray *et al.*, 2019].



(a) Goal (b) Button

Figure 6: Fig (a): In "Goal," the objective is to move the robot inside the green goal area; Fig (b): In "Button", the objective is to press the highlighted button (visually indicated with a faint gray cylinder), where figures and caption are taken from Safety Gym [Ray *et al.*, 2019].

space design might be to find action parameterizations that respect physical measures we care about—for example, an action space where a fixed distance corresponds to a fixed amount of energy.

Point: A robot constrained to the 2D plane, with one actuator for turning and another for moving forward/backward. This factored control scheme makes the robot particularly easy to control for navigation. Point has a small square in front that makes it easier to visually determine the robot's direction and helps the point push a box element that appears in one of our tasks.

Car: The car is a slightly more complex robot that has two independently-driven parallel wheels and a free-rolling rear wheel. The car is not fixed to the 2D plane but mostly resides in it. For this robot, both are turning and moving forward/backward require coordinating both of the actuators. It is similar in design to simple robots used in education.

H.3.2 Tasks

Tasks in Safety Gym are mutually exclusive, and an individual environment can only use a single task. Reward functions are configurable, allowing rewards to be either sparse (rewards only obtained on task completion) or dense (rewards have helpful, hand-crafted shaping terms). Task details are shown as follows.

Goal: Move the robot to a series of goal positions. When a goal is achieved, the goal location is randomly reset to someplace new, while keeping the rest of the layout the same. The sparse reward component is attained on achieving a goal position (robot enters the goal circle). The dense reward component gives a bonus for moving towards the goal (shown in Figure 5).

Button: Press a series of goal buttons. Several immobile "buttons" are scattered throughout the environment, and the agent should navigate to and press (contact) the currently-highlighted button, which is the goal button. After the agent presses the correct button, the environment will select and highlight a new goal button, keeping everything else fixed. The sparse reward component is attained

Environment		CPO	TRPO-L	PPO-L	FOCOPS	CUP
Safexp-PointGoal1-v0	Return	21.29 ± 3.49	19.23 ± 1.45	16.17 ± 5.89	12.46 ± 1.49	$\mathbf{23.74 \pm 0.12}$
Cost limit (25.0)	Constraint	39.00 ± 5.19	28.20 ± 5.21	21.82 ± 6.31	34.67 ± 2.62	24.74 ± 0.91
Safexp-PointButton1-v0	Return	17.69 ± 1.22	5.39 ± 1.02	4.74 ± 2.73	8.36 ± 0.34	$\mathbf{19.52 \pm 1.38}$
Cost limit (25.0)	Constraint	69.61 ± 8.29	25.15 ± 4.88	30.37 ± 7.58	18.56 ± 1.31	26.67 ± 1.84
Safexp-CarGoal1-v0	Return	$\mathbf{33.00 \pm 0.00}$	17.78 ± 2.34	19.93 ± 1.13	17.73 ± 3.50	27.41 ± 1.80
Cost limit 25.0)	Constraint	30.50 ± 1.44	23.00 ± 4.11	29.64 ± 4.79	25.50 ± 1.43	30.81 ± 1.60
Safexp-CarButton1-v0	Return	5.80 ± 1.06	0.48 ± 0.15	0.41 ± 0.13	9.47 ± 1.67	$\mathbf{12.12 \pm 1.91}$
Cost limit (25.0)	Constraint	93.88 ± 13.90	23.17 ± 9.76	16.23 ± 15.55	19.60 ± 1.52	29.41 ± 0.40

Table 4: Average results for CPO, PPO-L, TRPO-L, FOCOPS, CUP over 10 seeds after 500 iterations on Safety-Gym. The agent interacts with the environment 5000 times per iteration. Constraint limits are in brackets under the environment names.

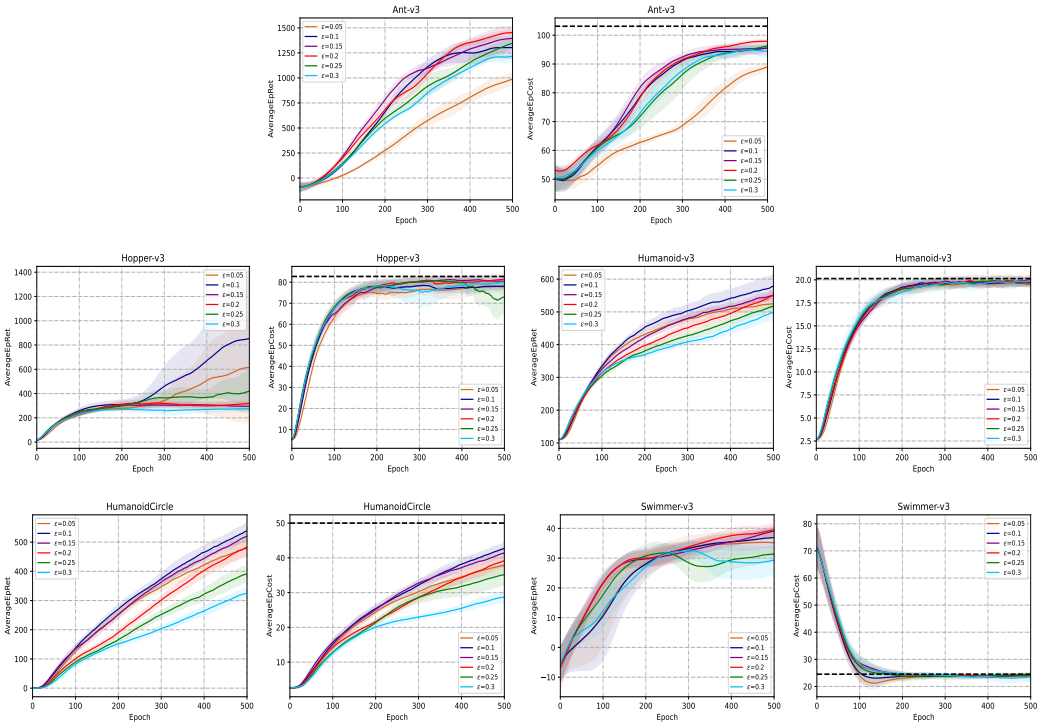


Figure 7: Performance with respect to penalty factor ϵ appears in Algorithm 1.

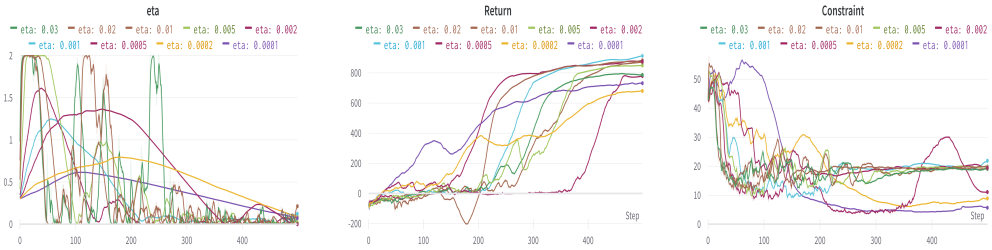
on pressing the current goal button. The dense reward component gives a bonus for moving towards the current goal button. We show a visualization in Figure 6).

H.4 Discussions

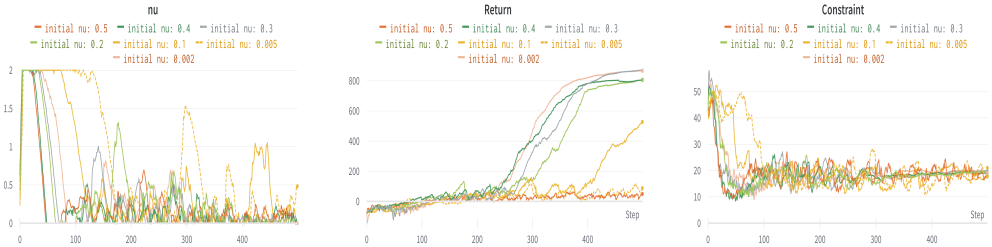
Results of Figure 7 show that the performance of CUP is still very stable for different settings of ϵ. Additionally, the constraint value of CUP also still fluctuates around the target value. The different value achieved by CUP in different setting ϵ is affected by the simulated environment and constraint thresholds, which are easy to control

The results of Table 4 show that the proposed CUP significantly outperforms all the baseline algorithms except on the Safexp-CarGoal1-v0 task. Notably, on the Safexp-PointButton1-v0 task, CUP achieve 21.27 ± 1.42 within the safety region, while the best baseline algorithm is CPO that only obtains a reward of 17.69 ± 1.22 but it violates the cost limit 25 more than a value of 44. This result is consistent with the result of Figure 2. Besides, from Table 4, we know although CPO achieves a reward of 33 ± 00 significantly outperforms the proposed CUP in Safexp-CarGoal1-v0, CPO needs a cost 30.50 ± 1.44 higher than CUP.

H.5 Additional Experiments for Performance with respect to initial η and ν.

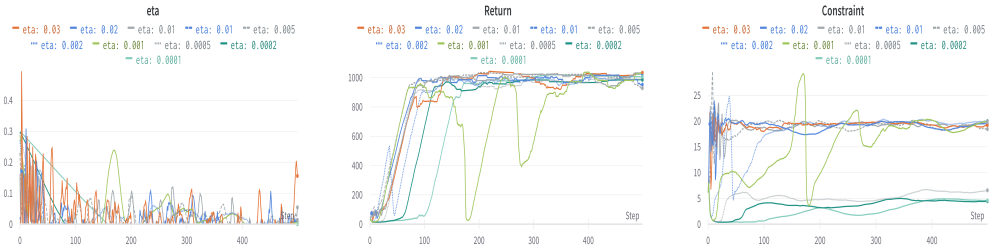


(a) η (b) Return with η (c) Constraint with η

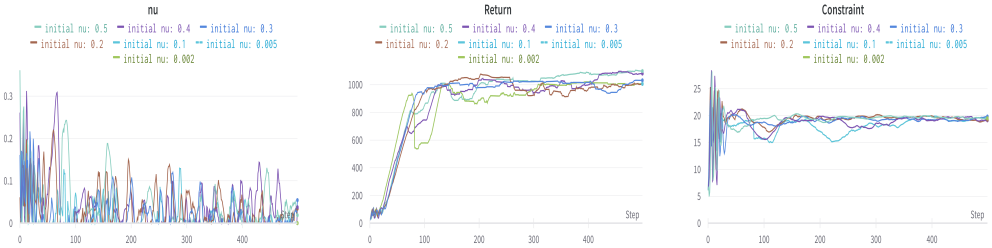


(d) ν (e) Return with ν (f) Ant:Constraint with ν

Figure 8: Ant: Performance with respect to initial η and ν.

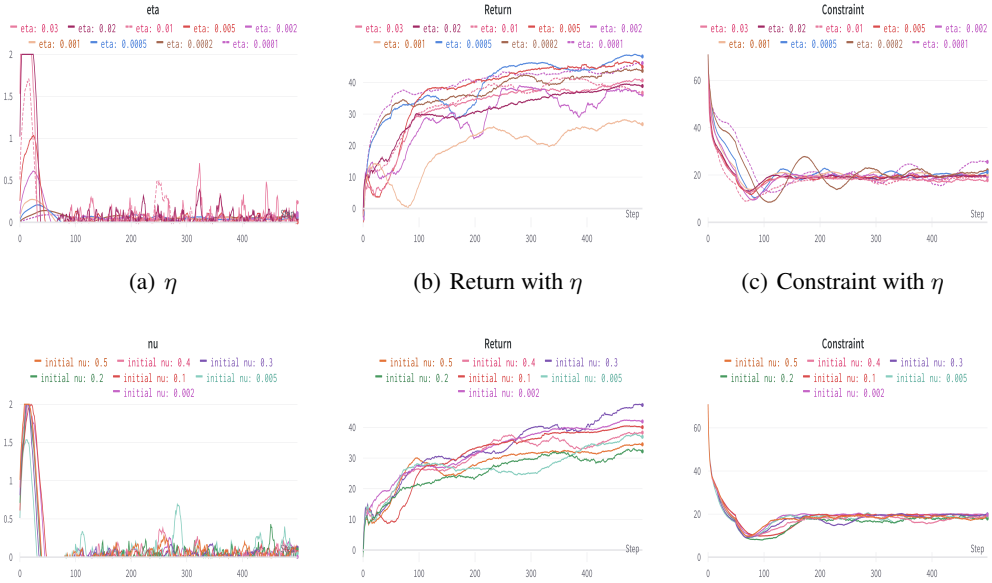


(a) η (b) Return with η (c) Constraint with η



(d) ν (e) Return with ν (f) Ant:Constraint with ν

Figure 9: Hopper: Performance with respect to initial η and ν.



(a) η (b) Return with η (c) Constraint with η

(d) ν (e) Return with ν (f) Ant:Constraint with ν

Figure 10: Swimmer: Performance with respect to initial η and ν.