Exterior Penalty Policy Optimization with Penalty Metric Network under Constraints

Shiqing Gao , **Jiaxin Ding*** , **Luoyi Fu** , **Xinbing Wang** and **Chenghu Zhou**

Shanghai Jiao Tong University

{shiqinggao, jiaxinding, yiluofu, xwang8}@sjtu.edu.cn, zhouchsjtu@gmail.com

Abstract

In Constrained Reinforcement Learning (CRL), agents explore the environment to learn the optimal policy while satisfying constraints. The penalty function method has recently been studied as an effective approach for handling constraints, which imposes constraints penalties on the objective to transform the constrained problem into an unconstrained one. However, it is challenging to choose appropriate penalties that balance policy performance and constraint satisfaction efficiently. In this paper, we propose a theoretically guaranteed penalty function method, Exterior Penalty Policy Optimization (EPO), with adaptive penalties generated by a Penalty Metric Network (PMN). PMN responds appropriately to varying degrees of constraint violations, enabling efficient constraint satisfaction and safe exploration. We theoretically prove that EPO consistently improves constraint satisfaction with a convergence guarantee. We propose a new surrogate function and provide worst-case constraint violation and approximation error. In practice, we propose an effective smooth penalty function, which can be easily implemented with a first-order optimizer. Extensive experiments are conducted, showing that EPO outperforms the baselines in terms of policy performance and constraint satisfaction with a stable training process, particularly on complex tasks.

1 Introduction

Deep reinforcement learning (RL) [Sutton and Barto, 2018] has demonstrated great potential in various real-world scenarios, including video games [Mnih *et al.*, 2016; Wang *et al.*, 2016; Vinyals *et al.*, 2019], robotics control [Gu *et al.*, 2016; Levine *et al.*, 2016; Haarnoja *et al.*, 2018], and Go [Silver *et al.*, 2016; Silver *et al.*, 2017]. However, many real-world problems inherit constraints when finding the optimal policy. For example, a robot agent should avoid obstacles while completing tasks. Such policy optimization with constraints is the focus of constrained RL (CRL), a critical field of RL research that expands the applicability of RL in real-world scenarios.

CRL is generally modeled as Constrained Markov Decision Processes (CMDP) [Beutler and Ross, 1985; Ross and Varadarajan, 1989; Altman, 2021], in which agents aim to maximize rewards while satisfying constraints. CMDP integrates safety criteria in the form of constraints into RL and provides a fundamental mathematical framework.

Extensive research has been conducted in CMDP, but most existing methods have inherent oscillations or high computational complexity. CPO [Achiam *et al.*, 2017] approximates the objective with a second-order expansion, leading to high computation costs of the high-dimensional inverse Fisher information matrix. Primal-dual methods convert constrained problems into unconstrained ones using the Lagrangian function to solve in the dual space. However, the learning dynamics exhibit high oscillations and constraint violations [Tessler *et al.*, 2018; Yu *et al.*, 2019; Paternain *et al.*, 2019; Ding *et al.*, 2020]. APPO [Dai *et al.*, 2023] employs the augmented Lagrangian method to enhance the stability of the primal-dual algorithm, but it is sensitive to Lagrangian multiplier initialization [Chow *et al.*, 2018; Stooke *et al.*, 2020]. Projection-based methods, such as PCPO [Yang *et al.*, 2020] and FOCOPS [Zhang *et al.*, 2020], project actions back to the feasible region, presenting a costly computation and underperforming cumulative rewards [Zhang *et al.*, 2023].

Penalty function methods impose constraints as penalties on the objective, addressing CRL in the primal space without introducing dual variables or requiring a complex second-order Taylor expansion [Liu *et al.*, 2020]. However, it is challenging to choose an appropriate penalty to trade off superior performance and efficient constraint satisfaction. An excessive penalty leads to a conservative policy that fails to explore the environment, while a minor penalty cannot guarantee the safety of the policy [Zhang *et al.*, 2023]. IPO [Liu *et al.*, 2020] restricts the policy within feasible regions with a logarithmic barrier function, which outputs excessive penalties near boundaries of the feasible region and cannot handle tasks where the initial policy violates constraints. P3O [Zhang *et al.*, 2022] utilizes a linear penalty function with the ReLU operator to optimize policy. When constraints are severely violated, the linear relation cannot provide sufficient penalty to bring the policy back to the feasible region quickly.

In this paper, we divide the policy space based on the de-

gree of constraint violation and introduce a Penalty Metric Network (PMN) comprising linear and quadratic cost value critics to capture penalties in different policy regions. The linear critic provides consistent incremental responses to small violations, while the quadratic critic imposes more severe penalties for larger violations, strongly discouraging substantial breaches. A dedicated weighting layer within PMN is used to aggregate different types of penalties, producing suitable penalties for efficient constraint satisfaction. A filter layer eliminates penalties in the feasible region, ensuring safe exploration for optimal performance. We propose a first-order penalty function algorithm, Exterior Penalty Policy Optimization (EPO), which imposes penalties output from PMN on the objective to transform constrained policy optimization into an unconstrained one. EPO guarantees monotonically non-increasing constraints through convergence analysis by maximizing sequential subproblems. Furthermore, we present a surrogate penalty function using samples generated from the current policy and theoretically provide the worst-case constraint violation and approximation error. In practice, EPO employs a smooth surrogate penalty function and an adaptive penalty factor adjustment strategy to ensure a stable training process. Extensive experiments on challenging tasks demonstrate that EPO achieves faster constraint satisfaction and better policy performance compared to multiple baseline algorithms.

In summary, our contributions are as follows:

- We design the Penalty Metric Network (PMN) to capture suitable constraint penalties in different policy regions.

- We propose the EPO algorithm based on PMN, which guarantees stable convergence and efficient constraint satisfaction. Theoretical guarantees on worst-case constraint violation and approximation errors are provided.

- We conduct extensive experiments to compare EPO with multiple state-of-the-art algorithms. EPO outperforms the baselines with higher performance and better constraint satisfaction.

2 Related Work

In the tabular case, the CMDP problem has been extensively studied [Beutler and Ross, 1986; Ross, 1989; Ross and Varadarajan, 1991]. For high-dimensional problems, RL with constraints is a challenging topic. Many methods have been proposed in recent years, which can be roughly divided into primal methods and primal-dual methods.

Primal-Dual methods. Primal-dual methods transform constrained problems into unconstrained ones by introducing dual variables [Chow *et al.*, 2017; Ding *et al.*, 2021]. [Tessler *et al.*, 2018] propose multi-timescale Lagrangian methods, which utilize penalty signals to guide the policy update towards constraint satisfaction. The traditional Lagrangian multiplier update behaves as integral control, and [Stooke *et al.*, 2020] introduce proportional and differential control to reduce cost overshoot and oscillations in the learning dynamics. [Ding *et al.*, 2020] establish the global convergence with sublinear rates regarding the optimality gap. The Augmented Lagrangian method is introduced in [Dai *et al.*,

2023] to reduce the oscillations during training. However, primal-dual approaches are still sensitive to initial parameters, limiting their applicability [Zhang *et al.*, 2022].

Primal methods. Primal methods directly optimize the constrained problem in the primal space [Chow *et al.*, 2018; Chow *et al.*, 2019; Dalal *et al.*, 2018; Yu *et al.*, 2022]. CPO [Achiam *et al.*, 2017] provides a lower bound on performance and an upper bound on constraint violation. PCPO [Yang *et al.*, 2020] first improves the reward within the trust region, then projects the policy to the feasible region. However, second-order methods in CPO and PCPO, which are based on local policy search [Peters and Schaal, 2008], encounter computational challenges due to the high-dimensional Hessian matrix. FOCOPS [Zhang *et al.*, 2020] proposes a first-order policy optimization method, which first solves the constraint problem in the nonparametric policy space and then projects the updated policy back into the parametric space. CUP [Yang *et al.*, 2022] provides generalized theoretical guarantees for surrogate functions with a generalized advantage estimator [Schulman *et al.*, 2015], effectively reducing variance while maintaining acceptable bias. However, projection approaches have poor performance since the correction only ensures feasibility but lacks equivalence with optimality [Zhang *et al.*, 2023]. All the above methods involve solving primal-dual subproblems, and the introduced dual variables lead to an unstable training process. Our approach offers a simpler first-order optimization method in the primal space, ensuring the stability of constraint satisfaction.

IPO [Liu *et al.*, 2020] augments the objective with a logarithmic barrier function to restrict the policy to feasible regions. However, the logarithmic function becomes ineffective when constraints are violated, limiting the agent's exploration. P3O [Zhang *et al.*, 2022] penalizes constraints with a ReLU operator to obtain an unconstrained problem. However, the linear penalty function in P3O cannot provide significant penalties for severe constraint violations, and the ReLU operator is non-differentiable at constraint boundaries, resulting in inaccurate gradient information. Compared to P3O and IPO, our PMN provides more appropriate penalties based on policy regions for faster constraint satisfaction. Theoretically, EPO provides convergence guarantees, continuous improvement constraints, and approximation errors, guaranteeing better performance and constraint satisfaction.

3 Preliminaries

RL can be modeled as a Markov decision process (MDP), represented by a tuple $(S, A, R, \mathcal{P}, \rho, \gamma)$, where S denotes the state space, A denotes the action space, $R : S \times A \to \mathbb{R}$ is the reward function, $\mathcal{P} : S \times A \to [0, 1]$ is the transition probability function, ρ is the initial state distribution, and $\gamma \in (0, 1)$ is the discount factor of the reward. The policy $\pi : S \to A$ maps states to probability distributions over actions, where $\pi(a|s)$ is the probability of choosing action a in state s, and Π is the set of all stationary policies. The discounted future state visitation distribution is defined as $d^\pi(s) := (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \mathcal{P}(s_t = s|\pi)$. In RL, we aim to find the optimal policy by maximizing the expected discounted return $J_R(\pi) := \mathbb{E}_{\tau \sim \pi}[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)]$, where

$\tau = (s_0, a_0, s_1, a_1, \cdots)$ denotes the trajectory based on policy π. The value function based on policy π is $V_R^\pi(s) := \mathbb{E}_{\tau \sim \pi}[\sum_{t=0}^\infty \gamma^t R(s_t, a_t)|s_0 = s]$, and action-value function is $Q_R^\pi(s, a) := \mathbb{E}_{\tau \sim \pi}[\sum_{t=0}^\infty \gamma^t R(s_t, a_t)|s_0 = s, a_0 = a]$. The advantage function measures the advantage of action a over the mean value: $A_R^\pi(s, a) := Q_R^\pi(s, a) - V_R^\pi(s)$.

The CMDP $(S, A, R, C, \mathcal{P}, \rho, \gamma)$ introduces constraints to the MDP to restrict allowable policies. $C : S \times A \to \mathbb{R}$ is the cost function, mapping state-action pairs to costs. d denotes the constraints threshold, the expected cumulative discount cost aims to satisfy $J_C(\pi) := \mathbb{E}_{\tau \sim \pi}[\sum_{t=0}^\infty \gamma^t C(s_t, a_t)] \leq d$. The cost value function $V_C^\pi(s)$, cost action-value function $Q_C^\pi(s, a)$ and cost advantage function $A_C^\pi(s, a)$ in CMDP can be obtained as in MDP by replacing the reward R with the cost C. The CRL aims to find an optimal policy by maximizing the expected discount return over the set of feasible policies $\Pi_C := \{\pi \in \Pi : J_C(\pi) \leq d\}$:

$$\arg \max_{\pi \in \Pi} J_R(\pi)$$
$$s.t. \quad J_C(\pi) \leq d \tag{1}$$

The following equation gives the performance difference between two arbitrary policies [Kakade and Langford, 2002]:

$$J_R(\pi') - J_R(\pi) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d^{\pi'}, a \sim \pi'}[A_R^\pi(s, a)] \tag{2}$$

This implies that iterative updates to the policy, $\pi'(s) = \arg \max_a A_R^\pi(s, a)$, lead to performance improvement until convergence to the optimal solution.

4 Methodology

In this section, we present the EPO algorithm and its theoretical foundation. We first show the structure of the PMN. Then, we propose the EPO method based on the PMN. The convergence analysis and error bound are provided to ensure effectiveness. Finally, we propose a smooth penalty function with penalty factor terms, which can be easily optimized in practice using a first-order optimizer.

4.1 Exterior Penalty Function with Penalty Metric Network

According to the performance difference equation (2), our problem is defined as a constrained optimization problem:

$$\pi_{k+1} = \arg \max_{\pi \in \Pi_\theta} \mathbb{E}_{s \sim d^\pi, a \sim \pi}[A_R^{\pi_k}(s, a)]$$
$$s.t. \quad J_C(\pi_k) + \frac{1}{1-\gamma} \mathbb{E}_{s \sim d^\pi, a \sim \pi}[A_C^{\pi_k}(s, a)] \leq d \tag{3}$$

where policy $\pi \in \Pi_\theta$ is parameterized with parameters θ, and π_k represents the current policy. For simplification, we introduce the notations to represent the above objective function $F_R(\pi) := \mathbb{E}_{s \sim d^\pi, a \sim \pi}[A_R^{\pi_k}(s, a)]$ and constraint function $F_C(\pi) := J_C(\pi_k) + \frac{1}{1-\gamma} \mathbb{E}_{s \sim d^\pi, a \sim \pi}[A_C^{\pi_k}(s, a)] - d$.

$F_C(\pi)$ can be considered as a measure of the constraint distance between policy π and the constraint boundaries in the policy space, indicating the degree of constraint violation. We
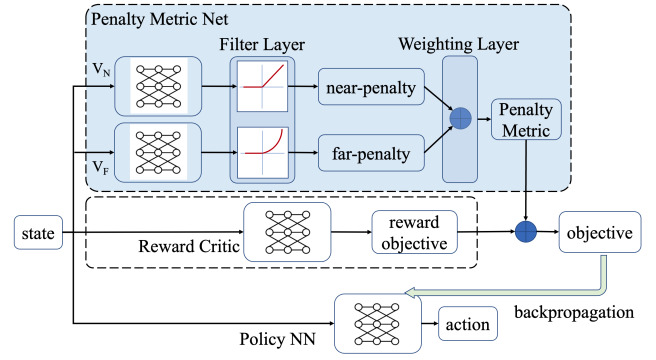

Figure 1: Structure of the Penalty Metric Network in EPO, comprising two streams to capture the near-penalty and far-penalty. V_N and V_F denote the linear and quadratic cost value critics, respectively. The Weighting Layer integrates the different types of penalties and outputs the Penalty Metric, which is imposed on the objective to guide the policy update.

normalize $F_C(\pi)$ across samples in the current replay buffer for uniform scaling. Then we divide the policy space into near-boundary region and far-boundary region according to $F_C(\pi)$, defining the penalties within these regions as near-penalty and far-penalty, respectively.

We propose a Penalty Metric Network to capture penalties across various regions. The PMN consists of two streams, representing the near-penalty and the far-penalty, as shown in Figure 1. We construct the linear cost value critic $V_N(s)$ and the quadratic cost value critic $V_F(s)$ as the backbone of near-penalty and far-penalty streams, respectively. Both critics are designed to provide sufficient penalties in their respective regions, ensuring efficient constraint satisfaction. For sufficient safety exploration, a filter layer eliminates negative values in both critics to ensure the penalty in the feasible region is 0. A weighting layer then integrates the two streams to generate a penalty metric for the current policy. We denote the PMN as a function $\Phi(F_C^+(\pi))$:

$$\Phi(F_C^+(\pi)) := \alpha F_C^+(\pi) + (1-\alpha)(F_C^+(\pi))^2 \tag{4}$$

where $\alpha \in [0, 1]$ is the weight factor in weighting layer, and $F_C^+(\pi) = \max\{(F_C(\pi)), 0\}$, $\max\{\cdot\}$ represents the filter operation. Imposing the penalty metric $\Phi(F_C^+(\pi))$ to the objective function $F_R(\pi)$, we obtain the exterior penalty function $P(\pi, \mu)$, which transforms the original constrained problem into an unconstrained one:

$$P(\pi, \mu) = F_R(\pi) - \frac{1}{\mu} \Phi(F_C^+(\pi)) \tag{5}$$

where $\mu > 0$ denotes the penalty factor. Now we get the new objective for (3) by maximizing the penalty function:

$$\arg \max_{\pi \in \Pi_\theta} F_R(\pi) - \frac{1}{\mu}\left(\alpha F_C^+(\pi) + (1-\alpha)(F_C^+(\pi))^2\right) \tag{6}$$

When constraints are satisfied, the objective focuses solely on maximizing the expected return. Conversely, when constraints are violated, the objective shifts to maximizing the expected return while simultaneously minimizing the cost.

Our Penalty Metric Network with linear and quadratic value estimators can provide appropriate penalties and bring the policy back to the feasible region Π_C quickly.

Properties 1. *Normalize the $F_C(\pi)$ and define the policy space for $F_C(\pi) \leq \frac{1}{2}$ as the near-boundary region, for $F_C(\pi) > 1$ as the far-boundary region, and for $\frac{1}{2} < F_C(\pi) \leq 1$ as the mid-boundary region. Increasing α provides increased penalty $\Phi(F_C^+(\pi))$ and gradient $|\nabla_\pi \Phi(F_C^+(\pi))|$ in the near-boundary region. Decreasing α provides increased penalty $\Phi(F_C^+(\pi))$ and gradient $|\nabla_\pi \Phi(F_C^+(\pi))|$ in the far-boundary region. Increasing α provides increased penalty $\Phi(F_C^+(\pi))$ and reduced gradient $|\nabla_\pi \Phi(F_C^+(\pi))|$ in the mid-boundary region.*

The weight α in the weighting layer is chosen based on $F_C(\pi)$. This ensures that PMN consistently generates large penalties and gradients when constraints are violated, enabling the policy to quickly return to the feasible region. In this paper, we set $\alpha = 0$ in the far-boundary region, 1 in the near-boundary region, and 0.5 in the mid-boundary region.

We use two neural networks $V_N(v)$ and $V_F(\psi)$, parameterized by v and ψ respectively, to capture the near-boundary cost value and far-boundary cost value. The Penalty Metric Network is updated by:

$$\arg\min_{v,\psi} \mathbb{E}_{s \sim d^\pi, a \sim \pi} \left[(V_N^{target} - V_N(s_t; v))^2 \right]$$

$$+ \mathbb{E}_{s \sim d^\pi, a \sim \pi} \left[(V_F^{target} + V_F(s_t; \psi) - 2\sqrt{V_F^{target} V_F(s_t; \psi)}) \right]$$

$$(7)$$

where $V_N^{target} = \max\{C_t + \gamma V(s_{t+1}) - d, 0\}$ and $V_F^{target} = (\max\{C_t + \gamma V(s_{t+1}) - d, 0\})^2$ are targets for V_N and V_F.

4.2 Exterior Penalty Policy Optimization and Convergence Analysis

Consider a decreasing sequence of penalty factors $\{\mu_t\}$ with $\mu_t \to 0$ as $t \to \infty$, where t denotes the iteration number of penalty factor. For each penalty factor μ_t, we solve a maximization problem of the penalty function $P(\pi, \mu_t)$ to obtain the corresponding global maximum policy π_t. By decreasing the penalty factor μ_t, the updated policy in EPO is guided towards the feasible region and eventually converges to the optimal policy under certain conditions. Below, we present the theoretical guarantees.

Lemma 1. *Suppose π_t is the global maximum policy of the penalty function $P(\pi, \mu_t)$ with factor μ_t, $\bar{\pi}$ is the optimal solution of the constrained problem (3), and $F_R(\pi)$ is the objective function, then the inequality holds*

$$F_R(\bar{\pi}) \leq P(\pi_t, \mu_t) \leq F_R(\pi_t) \quad (8)$$

Decreasing the penalty factor $\mu_{t+1} < \mu_t$, we get

$$P(\pi_{t+1}, \mu_{t+1}) \leq P(\pi_t, \mu_t)$$
$$F_R(\pi_{t+1}) \leq F_R(\pi_t) \quad (9)$$

The constraint $F_C(\pi_t)$ is monotonically non-increasing with the decreasing μ_t and same α if constraint is violated

$$F_C(\pi_{t+1}) \leq F_C(\pi_t) \quad (10)$$

Proof. See the supplemental material.

Lemma 1 demonstrates that when the policy π_t violates constraints, the maximum value of the penalty function $P(\pi_t, \mu_t)$ is greater than or equal to the optimal value $F_R(\bar{\pi})$ of problem (3). By decreasing the penalty factor from μ_t to μ_{t+1}, both $P(\pi_{t+1}, \mu_{t+1})$ and $F_R(\pi_{t+1})$ approach the solution $F_R(\bar{\pi})$, and the policy approaches the feasible region with a monotonically non-increasing constraint. Furthermore, we establish the conditions ensuring that the updated policy in the penalty function satisfies the constraints.

Theorem 1. *Define a decreasing sequence of penalty factors $\{\mu_t\}$, suppose that the objective function $F_R(\pi)$ has an upper bound and π_t is the global maximum policy of penalty function $P(\pi, \mu_t)$ with factor μ_t. When $\mu_t \to 0$, the corresponding limit π^* of the sequence $\{\pi_t\}$ is a solution of the constrained problem (3).*

Corollary 1. *Consider the weight factor $\alpha = 1$, π_t is the global maximum policy of penalty function $P(\pi, \mu_t)$ with factor μ_t, λ^* is the Lagrange multiplier corresponding to the optimum $\bar{\pi}$ of the primal problem. When $\mu_t < \frac{1}{\|\lambda^*\|_\infty}$, π_t is the solution of the constrained problem (3).*

Proof. See the supplemental material.

According to Theorem 1, we need to solve the sequential subproblems with decreasing μ_t to obtain the optimal solution of the problem (3). As μ_t decreases, policy π_t converges to the optimal policy. Corollary 1 guides us to raise α to 1 within the near-boundary region, guaranteeing the policy returns to the feasible region with a finite penalty factor.

Algorithm 1 EPO: Exterior Penalty Policy Optimization

Input: Initialize policy network π_θ, value networks V_R^ω, V_N^v and V_F^ψ. Set the hyperparameter for PPO clip rate, μ, α for penalty function and learning rate η.
Output: The optimal policy parameter θ.
1: **for** epoch $k = 0, 1, 2, ...$ **do**
2:      Sample N trajectories $\tau_1, ..., \tau_N$ under the current policy π_{θ_k}.
3:      Process the trajectories to C-returns, and calculate advantage functions with V_N^v and V_F^ψ.
4:      **for** K iterations **do**
5:          Update value networks V_R^ω, V_N^v and V_F^ψ.
6:          Update policy network π_θ with first-order optimizer $\theta_{k+1} = \theta_k + \eta \nabla_\theta \left(L_R^{CLIP}(\theta) - \Psi(\mu)\Phi(log(1 + e^{L_C^{CLIP}(\theta)})) \right)$.
7:          **if** $\frac{1}{N}\sum_{j=1}^N D_{KL}(\pi_\theta || \pi_{\theta_k})[s_j] > \delta$ **then**
8:            Break.
9:          **end if**
10:     **end for**
11:     Decrease $\mu_{t+1} \in (0, \mu_t)$ and adjust $\alpha \in [0, 1]$ based on $F_C(\pi)$.
12: **end for**
13: **return** policy parameters $\theta = \theta_{k+1}$.

4.3 Surrogate Penalty Function within Trust Region and Theoretical Bounds

The complex dependency of state visitation distribution $d^\pi(s)$ on the unknown policy π makes (6) difficult to optimize directly. In this paper, we use samples generated by the current policy π_k to locally approximate the original problem (3) within the trust region:

$$\pi_{k+1} = \arg\max_{\pi \in \Pi_\theta} \mathbb{E}_{s \sim d^{\pi_k}, a \sim \pi}[A_R^{\pi_k}(s, a)]$$

$$s.t. \quad J_C(\pi_k) + \frac{1}{1-\gamma}\mathbb{E}_{s \sim d^{\pi_k}, a \sim \pi}[A_C^{\pi_k}(s, a)] \leq d \quad (11)$$

$$D(\pi\|\pi_k) \leq \delta$$

where Π_θ represents the policy set parameterized by parameter θ, $D(\pi\|\pi_k) = \mathbb{E}_{s \sim \pi_k}[D_{KL(\pi\|\pi_k)}[s]]$, D_{KL} is the KL divergence, and $\delta > 0$ is the step size. The set $\{\pi \in \Pi_\theta : D(\pi\|\pi_k) \leq \delta\}$ is the trust region. The surrogate objective and constraint are defined in a simplified formulation:

$$L_R^{\pi_k}(\pi) := \mathbb{E}_{s \sim d^{\pi_k}, a \sim \pi}[A_R^{\pi_k}(s, a)]$$

$$L_C^{\pi_k}(\pi) := \frac{1}{1-\gamma}\mathbb{E}_{s \sim d^{\pi_k}, a \sim \pi}[A_C^{\pi_k}(s, a)] + J_C(\pi_k) - d$$

$$(12)$$

Then the surrogate penalty objective function is:

$$\arg\max_{\pi \in \Pi_\theta} L_R^{\pi_k}(\pi) - \frac{1}{\mu}\left(\alpha L_C^{\pi_k +}(\pi) + (1-\alpha)(L_C^{\pi_k +}(\pi))^2\right)$$

$$s.t. \quad D(\pi\|\pi_k) \leq \delta$$

$$(13)$$

EPO is a trust region method, so it satisfies the worst-case bound for the updated policy in CPO.

Proposition 1 ([Achiam *et al.*, 2017]). *Suppose surrogate constraint $L_C^{\pi_k}(\pi_{k+1})$ is satisfied, the upper bound on constraint violation for each update is*

$$J_C(\pi_{k+1}) \leq d + \frac{\sqrt{2\delta}\gamma\epsilon_C^{\pi_{k+1}}}{(1-\gamma)^2} \quad (14)$$

where $\epsilon_C^{\pi_{k+1}} := \max_s |\mathbb{E}_{a \sim \pi_{k+1}}[A_C^{\pi_k}(s, a)]|$.

The local approximation approach introduces bias. We further give the worst-case error bound for the surrogate penalty function compared to the original penalty function. We denote the surrogate penalty function for problem (11) as $P^{\pi_k}(\pi, \mu_t) = L_R^{\pi_k}(\pi) - \frac{1}{\mu_t}\Phi(L_C^{\pi_k +}(\pi))$.

Theorem 2. *Suppose surrogate constraint $L_C^{\pi_k}(\pi)$ is satisfied, the upper bound on the approximation error between the surrogate and the original penalty functions is*

$$|P(\pi, \mu_t) - P^{\pi_k}(\pi, \mu_t)| \leq \frac{\sqrt{2\delta}\gamma\epsilon_R^\pi}{1-\gamma} +$$

$$\frac{1}{\mu_t}\left(\frac{\sqrt{2\delta}\alpha\gamma\epsilon_C^\pi}{(1-\gamma)^2} + \left(\frac{\sqrt{2\delta(1-\alpha)}\gamma\epsilon_C^\pi}{(1-\gamma)^2}\right)^2\right) \quad (15)$$

where $\epsilon_R^\pi := \max_s |\mathbb{E}_{a \sim \pi}[A_R^{\pi_k}(s, a)]|$, *and* $\epsilon_C^\pi := \max_s |\mathbb{E}_{a \sim \pi}[A_C^{\pi_k}(s, a)]|$.

Proof. See the supplemental material.

4.4 Smooth Penalty Function in Practical Implementation

The penalty function in (13) is mostly smooth except at the constraint boundary. However, the constraint boundary is crucial in CRL, particularly when the optimal policy lies close to it. The first-order optimizer needs precise gradient information during policy updates. To address these issues, we propose a novel smooth penalty function containing smoothing and factor terms in the implementation:

$$P_S(\pi, \mu) := F_R(\pi) - \Psi(\mu)\Phi(log(1 + e^{F_C(\pi)}))$$

$$where \quad \Psi(\mu) = \frac{1}{\mu}e^{F_C(\pi)} \quad (16)$$

where log denotes the natural logarithm, $log(1 + e^{F_C(\pi)})$ is the smoothing term. $\Phi(x) = \alpha x + (1-\alpha)x^2$ represents the penalty term, $\Psi(\mu)$ is the penalty factor term.

The smooth term ensures that the penalty function remains differentiable throughout the parameter space. The adaptive factor function $\Psi(\mu)$ serves as a strategy for determining the penalty factor based on the empirical value of F_C, and it remains detached from the gradient calculation. This effectively reduces the frequency of factor adjustments, thus decreasing the number of sub-problems. For this implementation version in (16), Lemma 1 still guarantees monotonically non-increasing constraints, and Theorem 1 ensures that the practical policy converges to the feasible region. The proof is shown in the supplemental material.

To address the divergence constraint in the surrogate objective function (11), we employ the clipped surrogate function of the PPO [Schulman *et al.*, 2017] for approximation. Then, we get the final penalty objective in practice:

$$\arg\max_{\pi \in \Pi_\theta} L_R^{CLIP}(\pi) - \Psi(\mu)\Phi(log(1 + e^{L_C^{CLIP}(\pi)})) \quad (17)$$

$$L_R^{CLIP}(\pi) = \mathbb{E}_{s \sim d^{\pi_k}, a \sim \pi_k}[min\{r(\pi)A_R^{\pi_k}(s, a),$$
$$clip(r(\pi), 1 - \epsilon, 1 + \epsilon)A_R^{\pi_k}(s, a)\}]$$

$$L_C^{CLIP}(\pi) = \frac{1}{1-\gamma}\mathbb{E}_{s \sim d^{\pi_k}, a \sim \pi_k}[min\{r(\pi)A_C^{\pi_k}(s, a),$$
$$clip(r(\pi), 1 - \epsilon, 1 + \epsilon)A_C^{\pi_k}(s, a)\}] + J_C(\pi_k) - d$$

$$(18)$$

where $r(\pi) = \frac{\pi}{\pi_k}$ is the importance sampling ratio, ϵ is the clip ratio. Algorithm 1 presents a summary of EPO.

5 Experiment

We train different agents and design comparison experiments in four navigation tasks based on Safety Gymnasium [Brockman *et al.*, 2016] and four MuJoCo physical simulator tasks [Todorov *et al.*, 2012]. Baselines include the primal-dual method PPO Lagrangian [Chow *et al.*, 2017; Schulman *et al.*, 2017], the second-order optimization method CPO [Achiam *et al.*, 2017], the augmented Lagrangian method APPO [Dai *et al.*, 2023], and the penalty function method IPO [Liu *et al.*, 2020] and P3O [Zhang *et al.*, 2022]. For fairness of the results, we conduct multiple comparison experiments in
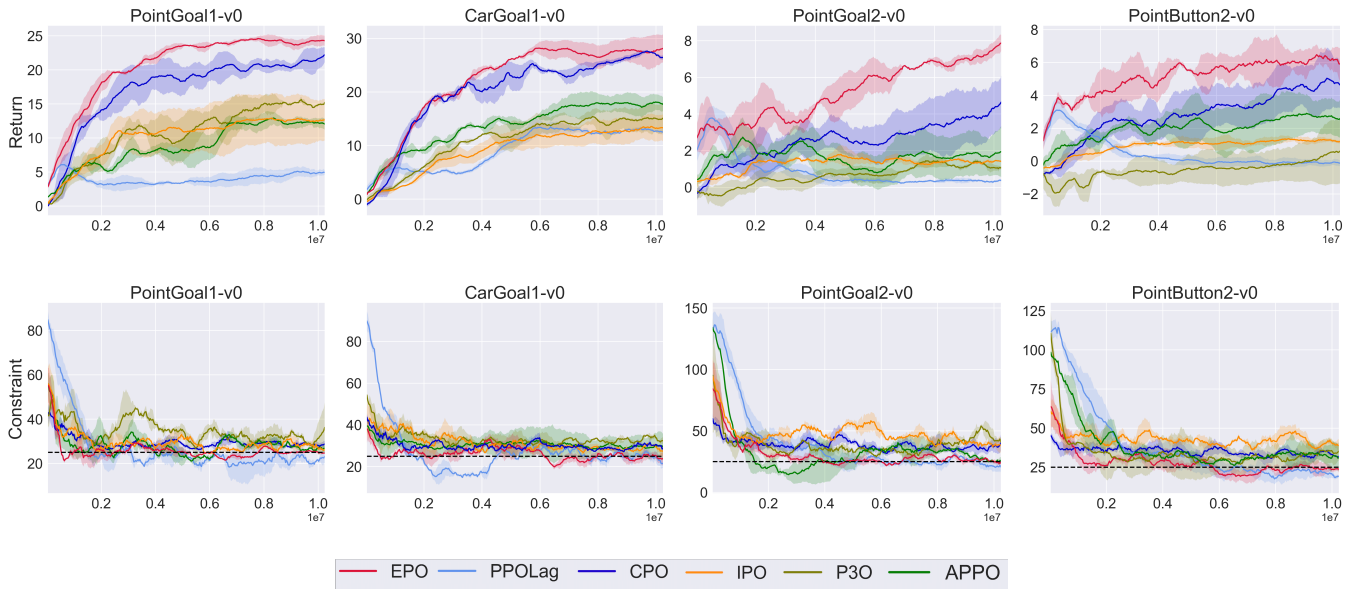
Figure 2: Comparison of EPO to baselines over 3 seeds on Safety Gymnasium. The x-axis is the training steps, the y-axis is the average return or constraint. The solid line is the mean and the shaded area is the standard deviation. The dashed line marks the constraint limit of 25.

the same environment and with the same settings. Extensive experiments demonstrate that (1) EPO outperforms current state-of-the-art algorithms on policy performance and constraint satisfaction, and (2) provides stable convergence and fewer constraint violations during training.

Scenario Description. All tasks aim to maximize the expected reward (the higher, the better) while satisfying the constraint (the lower, the better). In Safety Gymnasium, we train different agents, including Point and Car, for various navigation tasks: (1) the Goal task, where agents navigate to the goal location while circumventing hazards, and (2) the Button task, where agents navigate to the goal button and touch the correct goal button while avoiding gremlins and hazards. Additionally, we test the more difficult Goal2 and Button2 tasks, requiring avoiding more hazards and vases, which have not been proven to be effectively solved by baselines. In Safety MuJoCo, different agents are rewarded for running along a straight line with a velocity limit for safety and stability.

Safety Gymnasium. Navigation tasks in Safety Gymnasium with infeasible initial policies are challenging. Figure 2 presents the learning curves of EPO and baselines. Table 1 in the supplemental material shows the mean performance of different algorithms. EPO demonstrates impressive policy performance and faster constraint satisfaction compared to baselines in challenging tasks. In tasks with infeasible initial policies, EPO converges faster and generates fewer constraint violations, as illustrated in Table 2 in the supplemental material, demonstrating the effectiveness of PMN in providing appropriate penalties when constraints are violated. In contrast, IPO tends to generate conservative policies restricted by excessive penalties and struggles with infeasible initial policies for constraint satisfaction. P3O fails to provide enough penalties when the initial constraint is large, resulting in a slow de-

crease in constraints. PPO Lagrangian requires waiting for an increase in the dual variable to mitigate constraints, leading to a slower convergence rate of constraints. APPO outperforms PPO Lagrangian in performance and stability but is sensitive to the initial Lagrangian multiplier. In PointGoal2-v0 and PointButton2-v0, most baselines update the policy conservatively to ensure constraint satisfaction, while EPO shows better performance under the cost limit.

Safety MuJoCo. To increase the challenge, we set a cost limit of 25 for all tasks in Safety MuJoCo. Figure 3 illustrates the learning curves of EPO and baseline algorithms. The results indicate that EPO maintains excellent performance with stable training processes, keeping the policy in the feasible region for tasks with feasible initial policies. P3O struggles to provide suitable penalties to keep the policy within the feasible domain, showing poor performance in Swimmer-v4. In Safety MuJoCo tasks, the optimal policy often resides near constraint boundaries, while the ReLU operator used in P3O eliminates gradient information around these boundaries, making it challenging to identify the optimal policy. Experimental results demonstrate the effectiveness of our smooth penalty function, leading to enhanced policy performance. CPO shows faster performance convergence than EPO in Ant-v4. However, the computational burden imposed by the inverse of the high-dimensional Fisher information matrix leads to training times more than twice as long as EPO. PPO Lagrangian outperforms EPO in HalfCheetah-v4 and Humanoid-v4 due to the slower updates of Lagrange multipliers compared to policy iterations. The policy is not penalized promptly when it reaches the infeasible region, which leads to higher rewards but large constraint oscillations. APPO mitigates these oscillations through the augmented Lagrangian method but decreases policy performance.
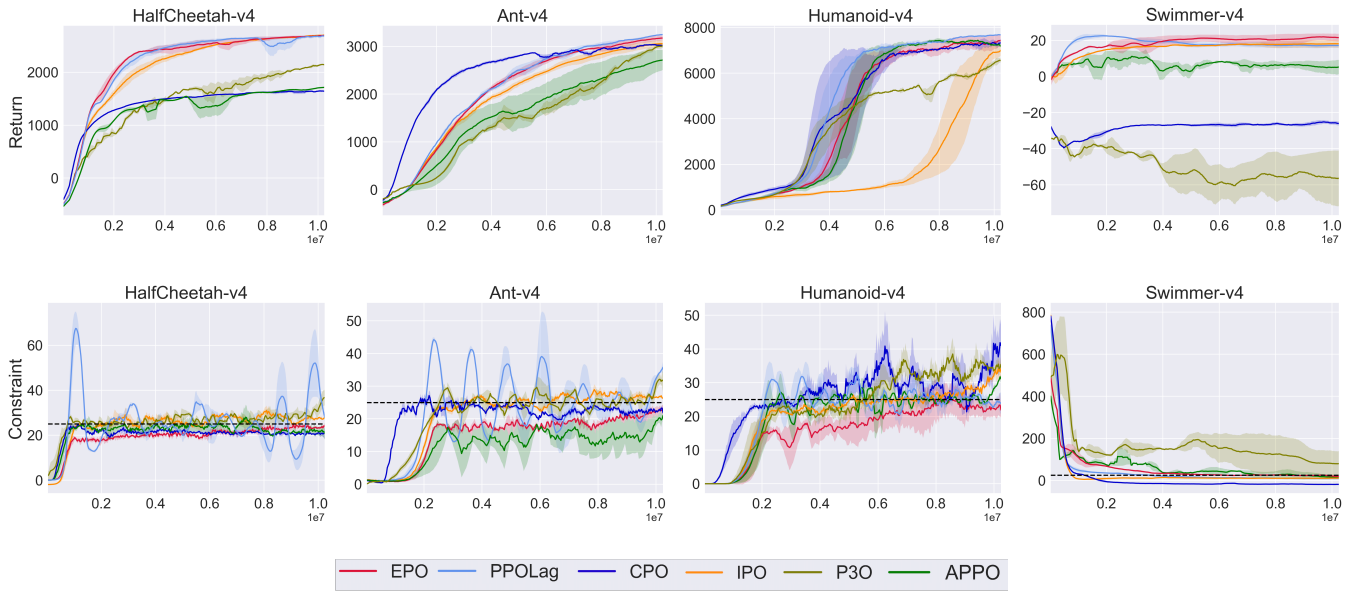
Figure 3: Comparison of EPO to the baselines over 3 seeds on Safety MuJoCo. The dashed line marks the constraint limit of 25.
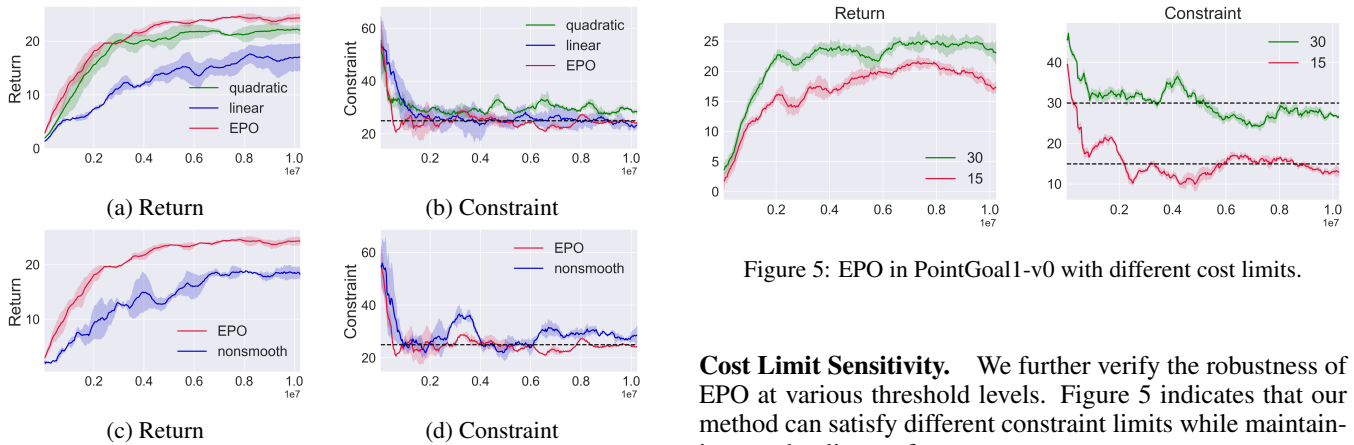

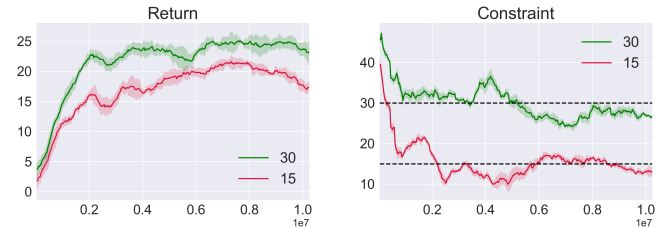Figure 4: Ablation experiments in PointGoal1-v0.

Figure 5: EPO in PointGoal1-v0 with different cost limits.

Cost Limit Sensitivity. We further verify the robustness of EPO at various threshold levels. Figure 5 indicates that our method can satisfy different constraint limits while maintaining good policy performance.

Ablation Experiments. We conduct ablation experiments to verify the effectiveness of components in EPO. First, we compare the PMN with linear and quadratic critics. The results in Figure 4(a)(b) show the EPO with PMN outperforms the other two. The linear critic imposes larger penalties near the constraint boundary, guiding the policy to the feasible region, while the quadratic one imposes larger penalties far from the boundary, accelerating the policy to approach the feasible region. PMN combines the benefits from the linear and quadratic critics, making a trade-off between constraint satisfaction and reward maximization based on the degree of constraint violation. Second, we compare the smooth EPO function with the non-smooth ReLU function, the results in Figure 4(c)(d) show that the smooth function enhances training stability, especially near constraint boundaries, by providing gradient information across the policy space.

6 Conclusion

In this work, we propose Exterior Penalty Policy Optimization (EPO) for training CRL agents with superior performance and efficient constraint satisfaction. The Penalty Metric Network (PMN) generates appropriate penalties for fast constraint satisfaction in the infeasible region and safe exploration in the feasible region. Convergence analysis guarantees consistent constraint satisfaction. We further provide the worst-case constraint violation and approximation error analysis for the surrogate function, along with a practical implementation of EPO. EPO is compared with multiple baselines across a wide range of safety tasks, demonstrating its effectiveness in ensuring constraint satisfaction while maintaining policy performance. Future work includes adopting off-policy methods and addressing different types of constraints.

Acknowledgments

References

[Achiam *et al.*, 2017] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pages 22–31, 2017.

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

[Beutler and Ross, 1985] Frederick J Beutler and Keith W Ross. Optimal policies for controlled markov chains with a constraint. *Journal of mathematical analysis and applications*, 112(1):236–252, 1985.

[Beutler and Ross, 1986] Frederick J Beutler and Keith W Ross. Time-average optimal constrained semi-markov decision processes. *Advances in Applied Probability*, 18(2):341–359, 1986.

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

[Chow *et al.*, 2017] Yinlam Chow, Mohammad Ghavamzadeh, Lucas Janson, and Marco Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *The Journal of Machine Learning Research*, 18(1):6070–6120, 2017.

[Chow *et al.*, 2018] Yinlam Chow, Ofir Nachum, Edgar Duenez-Guzman, and Mohammad Ghavamzadeh. A lyapunov-based approach to safe reinforcement learning. *Advances in neural information processing systems*, 31, 2018.

[Chow *et al.*, 2019] Yinlam Chow, Ofir Nachum, Aleksandra Faust, Edgar Duenez-Guzman, and Mohammad Ghavamzadeh. Lyapunov-based safe policy optimization for continuous control. *arXiv preprint arXiv:1901.10031*, 2019.

[Dai *et al.*, 2023] Juntao Dai, Jiaming Ji, Long Yang, Qian Zheng, and Gang Pan. Augmented proximal policy optimization for safe reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 7288–7295, 2023.

[Dalal *et al.*, 2018] Gal Dalal, Balázs Szörényi, Gugan Thoppe, and Shie Mannor. Finite sample analyses for td (0) with function approximation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

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

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

[Gu *et al.*, 2016] Shixiang Gu, Timothy Lillicrap, Ilya Sutskever, and Sergey Levine. Continuous deep q-learning with model-based acceleration. In *International conference on machine learning*, pages 2829–2838, 2016.

[Haarnoja *et al.*, 2018] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870, 2018.

[Kakade and Langford, 2002] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pages 267–274, 2002.

[Levine *et al.*, 2016] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

[Liu *et al.*, 2020] Yongshuai Liu, Jiaxin Ding, and Xin Liu. Ipo: Interior-point policy optimization under constraints. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 4940–4947, 2020.

[Mnih *et al.*, 2016] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937, 2016.

[Paternain *et al.*, 2019] Santiago Paternain, Luiz Chamon, Miguel Calvo-Fullana, and Alejandro Ribeiro. Constrained reinforcement learning has zero duality gap. *Advances in Neural Information Processing Systems*, 32, 2019.

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

[Ross and Varadarajan, 1989] Keith W Ross and Ravi Varadarajan. Markov decision processes with sample path constraints: the communicating case. *Operations Research*, 37(5):780–790, 1989.

[Ross and Varadarajan, 1991] Keith W Ross and Ravi Varadarajan. Multichain markov decision processes with a sample path constraint: A decomposition approach. *Mathematics of Operations Research*, 16(1):195–207, 1991.

[Ross, 1989] Keith W Ross. Randomized and past-dependent policies for markov decision processes with multiple constraints. *Operations Research*, 37(3):474–477, 1989.

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

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

[Silver *et al.*, 2016] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[Silver *et al.*, 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):354–359, 2017.

[Stooke *et al.*, 2020] Adam Stooke, Joshua Achiam, and Pieter Abbeel. Responsive safety in reinforcement learning by pid lagrangian methods. In *International Conference on Machine Learning*, pages 9133–9143, 2020.

[Sutton and Barto, 2018] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[Tessler *et al.*, 2018] Chen Tessler, Daniel J Mankowitz, and Shie Mannor. Reward constrained policy optimization. *arXiv preprint arXiv:1805.11074*, 2018.

[Todorov *et al.*, 2012] 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, 2012.

[Vinyals *et al.*, 2019] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019.

[Wang *et al.*, 2016] Ziyu Wang, Victor Bapst, Nicolas Heess, Volodymyr Mnih, Remi Munos, Koray Kavukcuoglu, and Nando de Freitas. Sample efficient actor-critic with experience replay. *arXiv preprint arXiv:1611.01224*, 2016.

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

[Yang *et al.*, 2022] Long Yang, Jiaming Ji, Juntao Dai, Linrui Zhang, Binbin Zhou, Pengfei Li, Yaodong Yang, and Gang Pan. Constrained update projection approach to safe policy optimization. *arXiv preprint arXiv:2209.07089*, 2022.

[Yu *et al.*, 2019] Ming Yu, Zhuoran Yang, Mladen Kolar, and Zhaoran Wang. Convergent policy optimization for safe reinforcement learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[Yu *et al.*, 2022] Haonan Yu, Wei Xu, and Haichao Zhang. Towards safe reinforcement learning with a safety editor policy. *arXiv preprint arXiv:2201.12427*, 2022.

[Zhang *et al.*, 2020] Yiming Zhang, Quan Vuong, and Keith Ross. First order constrained optimization in policy space. *Advances in Neural Information Processing Systems*, 33:15338–15349, 2020.

[Zhang *et al.*, 2022] Linrui Zhang, Li Shen, Long Yang, Shixiang Chen, Bo Yuan, Xueqian Wang, and Dacheng Tao. Penalized proximal policy optimization for safe reinforcement learning. *arXiv preprint arXiv:2205.11814*, 2022.

[Zhang *et al.*, 2023] Linrui Zhang, Qin Zhang, Li Shen, Bo Yuan, Xueqian Wang, and Dacheng Tao. Evaluating model-free reinforcement learning toward safety-critical tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 15313–15321, 2023.