Towards Safe Reinforcement Learning via Constraining Conditional Value-at-Risk

ChengYang Ying[1] , **Xinning Zhou**[1] , **Hang Su**[*1,2,3] , **Dong Yan**[1] , **Ning Chen**[1] and **Jun Zhu**[*1,2,3]
[1]Department of Computer Science & Technology, Institute for AI, BNRist Center,
Tsinghua-Bosch Joint ML Center, THBI Lab, Tsinghua University;
[2]Peng Cheng Laboratory
[3] Tsinghua University-China Mobile Communications Group Co., Ltd. Joint Institute
ycy21@mails.tsinghua.edu.cn, {coderlemon, sproblvem17}@gmail.com,
{suhangss, ningchen, dcszj}@mail.tsinghua.edu.cn

Abstract

Though deep reinforcement learning (DRL) has obtained substantial success, it may encounter catastrophic failures due to the intrinsic uncertainty of both transition and observation. Most of the existing methods for safe reinforcement learning can only handle transition disturbance or observation disturbance since these two kinds of disturbance affect different parts of the agent; besides, the popular worst-case return may lead to overly pessimistic policies. To address these issues, we first theoretically prove that the performance degradation under transition disturbance and observation disturbance depends on a novel metric of Value Function Range (VFR), which corresponds to the gap in the value function between the best state and the worst state. Based on the analysis, we adopt conditional value-at-risk (CVaR) as an assessment of risk and propose a novel reinforcement learning algorithm of CVaR-Proximal-Policy-Optimization (CPPO) which formalizes the risk-sensitive constrained optimization problem by keeping its CVaR under a given threshold. Experimental results show that CPPO achieves a higher cumulative reward and is more robust against both observation and transition disturbances on a series of continuous control tasks in MuJoCo.

1 Introduction

Deep reinforcement learning (DRL) has achieved enormous success on a variety of tasks, ranging from playing Atari games [Mnih *et al.*, 2015] and Go [Silver *et al.*, 2016] to manipulating complex robotics in real world [Kendall *et al.*, 2019]. However, due to the intrinsic uncertainty of both transition and observation, these methods may result in catastrophic failures [Heger, 1994; Huang *et al.*, 2017], i.e., the agent may receive significantly negative outcomes. This phenomenon is attributed to several factors. One is that traditional DRL only aims at cumulative return maximization without considering the stochasticity of the environment [Garcıa and Fernández, 2015], which may lead to serious consequences with a certain probability and thereby expose policies to risk. This can be illustrated briefly in the case of self-driving, where the agent might try to achieve the highest reward by acting dangerously, e.g., agents may drive along the edge of a curve in order to reach the destination more quickly without considering the potential danger. Also, a random disturbance or adversarial disturbance may interfere with the agent's observation, yielding a significant performance degeneration [Huang *et al.*, 2017].

Various efforts has been made on safe reinforcement learning (safe RL) to handle transition uncertainty and observation uncertainty [Heger, 1994; Garcıa and Fernández, 2015; Zhang *et al.*, 2020]. Garcıa and Fernández (2015) conduct a comprehensive survey on safe RL and argue that an array of methods focusing on transition uncertainty are based on transforming the optimization criterion. For example, robust approximate dynamic programming [Tamar *et al.*, 2013], based on a projected fixed point equation, considers how to solve robust MDPs [Wiesemann *et al.*, 2013] approximately to improve the robustness of the agent under transition uncertainty. Moreover, there is an array of work paying attention to observation disturbance. For example, some recent work formulates observation disturbance as a state-adversarial Markov decision process (SA-MDP) and proposes robust algorithms against observation disturbance. However, such work for handling transition uncertainty and observation uncertainty has some major drawbacks. First, due to the consideration of the worst-case outcomes [Heger, 1994], those methods may lead to overly pessimistic policies, which will focus too much on the worst case and own poor average performance. Moreover, though the agent may suffer from transition uncertainty and observation uncertainty at the same time, existing work [Nilim and El Ghaoui, 2005; Tamar *et al.*, 2013; Zhang *et al.*, 2020] can only handle observation disturbance or transition disturbance separately. The main reason is that these two kinds of disturbance are structurally different. To the best of our knowledge, there is currently no analysis on the connection of the two typical uncertainties and few methods to deal with them both at the same time.

To build a connection between transition disturbance and observation disturbance, we first prove that the performance degradation resulting from each of them is theoretically dependent on a new notion of *Value Function Range* (VFR), which is the value function gap between the best and worst states. However, directly controlling VFR may also suf-

*Corresponding author.

fer from the problem of excessive pessimism, because VFR only considers the value of extreme states, and moreover, the value function calculated in VFR is difficult to estimate. We first use conditional value-at-risk (CVaR) as a slack of the minimum since CVaR can be used for avoiding overly pessimistic policies [Alexander and Baptista, 2004; Alexander *et al.*, 2006]. Moreover, We theoretically prove that the CVaR of the return of trajectories is a lower bound of the CVaR of the value function and explain that the former one is much easier to estimate.

Based on this theoretical analysis, we propose to use the CVaR of the return of trajectories to replace VFR and formulate a CVaR-based constrained optimization problem for safe RL under transition disturbance as well as observation disturbance. Furthermore, we analyze the properties of this optimization problem and present a new algorithm called CVaR-Proximal-Policy-Optimization (CPPO) based on Proximal Policy Optimization (PPO) [Schulman *et al.*, 2017]. Empirically, we compare CPPO to multiple on-policy baselines as well as some previous CVaR-based methods on various continuous control tasks in MuJoCo [Todorov *et al.*, 2012]. Our results show that CPPO achieves competitive cumulative reward in the training stage and exhibits stronger robustness when we apply perturbations to these environments.

In summary, our contributions are:

- We theoretically analyze the performance of trained policies under transition and observation disturbance, and use VFR to build a theoretical connection between these two types of structurally different disturbance;
- Based on the analysis, we present a constrained optimization problem to maximize the cumulative reward and simultaneously control the risk, which is solved by our CPPO algorithm under the regularization of CVaR;
- We empirically demonstrate that CPPO exhibits stronger robustness under transition/observation perturbations compared with the alternative common on-policy RL algorithms and previous CVaR-based RL algorithms on different MuJoCo tasks.

2 Background

In this section, we briefly introduce safe reinforcement learning (safe RL) and conditional value-at-risk (CVaR), which motivate us to adopt CVaR as a metric of risk in safe RL.

2.1 Safe RL

In a standard RL setting, the agent interacts with an unknown environment and learns to achieve the highest long-term return. The task is modeled as a Markov decision process (MDP) of $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ represent the state space and the action space, respectively; $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ denotes the transition probability that captures the dynamics of the environment; $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to [-R_{\max}, R_{\max}]$ represents the reward function; and γ is a discount factor. We use π_θ to represent the policy of the agent with parameter θ, which is a mapping from $\mathcal{S}$ to the set of distributions on $\mathcal{A}$. At each time step t, the agent perceives the current state $s_t \in \mathcal{S}$, chooses its action $a_t \in \mathcal{A}$ sampled from the distribution $\pi_\theta(\cdot|s_t)$ and obtains a reward r_t. However, most of the current algorithms try to maximize cumulative reward without considering the risk of the policy, which may cause catastrophic results [Heger, 1994].

To address this problem, an array of safe RL methods tend to change the objective in order to eliminate the uncertainty and avoid the danger [Garcıa and Fernández, 2015]. In general, two kinds of uncertainty are widely discussed, namely, transition uncertainty and observation uncertainty. The transition uncertainty of RL denotes scenarios where the parameters of the MDP are unknown or there is a gap between the training and testing environments. Studies conducted by Nilim and El Ghaoui and Tamar *et al.* assume that the actual transition belongs to a set $\hat{\mathcal{P}}$ and propose to optimize

$$\max_\theta \min_{\mathcal{P}\in\hat{\mathcal{P}}} J_{\mathrm{tr}}(\pi_\theta, \mathcal{P}) \triangleq \mathbb{E}\left[D(\pi_\theta) \triangleq \sum_{t=1}^{\infty} \gamma^t r_t \Big| \pi_\theta, \mathcal{P}\right]. \quad (1)$$

As for the observation uncertainty of RL, it refers to the gap between the observation and the true state. For example, the observation of the agent may be disturbed by random disturbance as well as adversarial disturbance, which will cause a drop in the performance [Huang *et al.*, 2017]. For evaluating observation uncertainty, some previous work [Zhang *et al.*, 2020] has assumed that the observation will be disturbed as $\nu(s) \in \mathcal{S}$ when the true state is s, and hope to find a robust policy under any $\nu \in \Gamma$, here Γ is the set of all state-observation disturbance. Based on that assumption, such work has built a framework named state-adversarial MDP (SA-MDP) to solve

$$\max_\theta \min_{\nu\in\Gamma} J_{\mathrm{obs}}(\pi_\theta) \triangleq \mathbb{E}\left[D(\pi_\theta, \nu) \triangleq \sum_{t=1}^{\infty} \gamma^t r_t\right]. \quad (2)$$

However, existing safe RL methods are not problemless. First, both (1) and (2) are *max-min problems*, which do not have general effective solvers and usually have high computational complexity. Second, focusing on the worst trajectories may cause overly pessimistic behaviors. Finally, because these two kinds of uncertainty are structurally different, existing work always considers them separately rather than building a connection between them.

2.2 CVaR

Value-at-risk (VaR) and conditional value-at-risk (CVaR) are both well-established metrics for measuring risk in economy [Alexander and Baptista, 2004]. First, we give their definitions [Chow and Ghavamzadeh, 2014] below.

Definition 1 (VaR and CVaR)**.** *For a bounded-mean random variable Z, the value-at-risk (VaR) of Z with confidence level $\alpha \in (0, 1)$ is defined as:*

$$\mathrm{VaR}_\alpha(Z) = \min\{z|F(z) \geq \alpha\}, \quad (3)$$

where $F(z) = P(Z \leq z)$ is the cumulative distribution function (CDF); and the conditional value-at-risk (CVaR) of Z with confidence level α is defined as the expectation of the α-tail distribution of Z as

$$\mathrm{CVaR}_\alpha(Z) = \mathbb{E}_{z\sim Z}\{z|z \geq \mathrm{VaR}_\alpha(Z)\}. \quad (4)$$

It is easy to prove that [Chow *et al.*, 2015]

$$\lim_{\alpha\to 1^-} \mathrm{CVaR}_\alpha(Z) = \max(Z). \quad (5)$$

Previous work [Chow and Ghavamzadeh, 2014; Chow *et al.*, 2015; Chow *et al.*, 2017] has attempted to use CVaR to analyze the risk-MDP, which considers a risk function $\mathcal{C}$ rather than a reward function $\mathcal{R}$. They propose gradient-based methods and value-based methods to optimize loss of MDP as well as keeping the CVaR under a certain value. However, these studies ignore the reward in MDP and thus cannot be directly used in RL settings.

Besides, there are an array of work for optimizing CVaR [Tamar *et al.*, 2015; Tang *et al.*, 2020], optimizing the CVaR-constrained objective [Prashanth, 2014; Yang *et al.*, 2021], and analyzing the connection between optimizing CVaR and the robustness against transition disturbance [Chow *et al.*, 2015; Rigter *et al.*, 2021]. Also, there are some work [Ma *et al.*, 2020] extending methods in distributional RL [Dabney *et al.*, 2018], which mainly considers the randomness of the return, for CVaR optimization.

3 Theoretical Analysis

In this section, we first analyze the robustness of policies against transition perturbations and observation disturbances, and further build a connection between them.

3.1 Value Function Range

For an MDP $\mathcal{M}$ and a given policy π, we denote its expected cumulative reward and value function as $J_{\mathcal{M}}(\pi)$ and $V_{\mathcal{M},\pi}$ [Sutton and Barto, 2018], respectively. We define the *Value Function Range* (VFR) to capture the gap of the value function between the best state and the worst state as follows.

Definition 2 (Value Function Range)**.** *For MDP $\mathcal{M}$, the Value Function Range (VFR) of the policy π is*

$$\hat{V}_{\mathcal{M},\pi} \triangleq \max_s V_{\mathcal{M},\pi}(s) - \min_s V_{\mathcal{M},\pi}(s). \tag{6}$$

Moreover, for every state $s \in \mathcal{M}$, we define its discounted future state distribution as

$$d^{\pi}_{\mathcal{M}}(s) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathcal{P}(s_t = s|\pi, \mathcal{M}).$$

3.2 Performance against Transition Disturbance

First, we consider transition disturbance. Assume that the transition $\mathcal{P}$ is disturbed by $\hat{\mathcal{P}}$, we attempt to evaluate the reduction of cumulative reward against the disturbance. We can calculate and bound the difference of performance of π under $\mathcal{M}$ and $\hat{\mathcal{M}}$ in Theorem 1 as below:

Theorem 1. *For any policy π in MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$ and any disturbed environment $\hat{\mathcal{M}} = (\mathcal{S}, \mathcal{A}, \hat{\mathcal{P}}, \mathcal{R}, \gamma)$, the reduction of the cumulative reward against the transition disturbance is*

$$\begin{aligned} &J_{\mathcal{M}}(\pi) - J_{\hat{\mathcal{M}}}(\pi) \\ =&\frac{\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\pi}_{\hat{\mathcal{M}}}}\mathbb{E}_{a\sim\pi}\mathbb{E}_{s'\sim\hat{\mathcal{P}}}\left(1 - \frac{\mathcal{P}(s'|s,a)}{\hat{\mathcal{P}}(s'|s,a)}\right)V_{\mathcal{M},\pi}(s'). \end{aligned}$$

Furthermore, an upper bound of the reduction is

$$\begin{aligned} &\left|J_{\mathcal{M}}(\pi) - J_{\hat{\mathcal{M}}}(\pi)\right| \\ \le&\frac{2\gamma}{1-\gamma}\max_{s,a} D_{\mathrm{TV}}(\mathcal{P}(\cdot|s,a), \hat{\mathcal{P}}(\cdot|s,a))\hat{V}_{\mathcal{M},\pi}. \end{aligned} \tag{7}$$

The key of the proof is to analyze the relationship of $V_{\mathcal{M},\pi} - V_{\hat{\mathcal{M}},\pi}$ with different states. We defer the complete proof to Appendix B.1, which resembles the proof by [Kakade and Langford, 2002]. Compared with the policy π for a given MDP $\mathcal{M}$, the factors that mainly affect the performance of π in disturbed environment $\hat{\mathcal{M}}$ are Total Variation (TV) distance $\max_{s,a} D_{\mathrm{TV}}(\mathcal{P}(\cdot|s,a), \hat{\mathcal{P}}(\cdot|s,a))$ and VFR $\hat{V}_{\mathcal{M},\pi}$. The TV distance, depends on the range of transition disturbance, which is independent with the agent and cannot be controlled by safe RL. By contrast, VFR depends only on the value functions of π in $\mathcal{M}$ and is an intrinsic property of the policy π. Therefore, we can improve the robustness of the policy π under a transition disturbance policy by controlling $\hat{V}_{\mathcal{M},\pi}$.

3.3 Performance against Observation Disturbance

Now, we consider the situation of observation disturbance. Similarly to the setting of SA-MDP [Zhang *et al.*, 2020], we introduce adversary $\nu : \mathcal{S} \to \mathcal{S}$ to describe the disturbance of the state and denote the policy disturbed by adversary ν as $\hat{\pi}_\nu$, which means $\hat{\pi}_\nu(\cdot|s) = \pi(\cdot|\nu(s))$. Similarly to Theorem 1, we can also show a similar result as below:

Theorem 2. *For any policy π and any adversary ν, the reduction of the expected cumulative reward of π against the observation disturbance of ν is*

$$\begin{aligned} &J_{\mathcal{M}}(\pi) - J_{\mathcal{M}}(\hat{\pi}_\nu) \\ =&\frac{\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\hat{\pi}_\nu}_{\mathcal{M}}}\mathbb{E}_{a\sim\pi(\cdot|\nu(s))}\left(1 - \frac{\pi(a|s)}{\pi(a|\nu(s))}\right) \\ &\mathbb{E}_{s'\sim\mathcal{P}}V_{\mathcal{M},\pi}(s') \\ +&\frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\hat{\pi}_\nu}_{\mathcal{M}}}\mathbb{E}_{a\sim\pi(\cdot|\nu(s))}\left(1 - \frac{\pi(a|s)}{\pi(a|\nu(s))}\right)\mathcal{R}(s,a). \end{aligned}$$

Furthermore, an upper bound of the reduction is

$$\begin{aligned} &|J_{\mathcal{M}}(\pi) - J_{\mathcal{M}}(\hat{\pi}_\nu)| \\ \le&\frac{\gamma}{1-\gamma}\max_s D_{\mathrm{TV}}(\pi(\cdot|s), \pi(\cdot|\nu(s)))\hat{V}_{\mathcal{M},\pi} \\ +&\frac{2}{1-\gamma}\max_s D_{\mathrm{TV}}(\pi(\cdot|s), \pi(\cdot|\nu(s))\max_{s,a}|\mathcal{R}(s,a)|. \end{aligned} \tag{8}$$

The proof is similar to that of Theorem 1 and is also included in Appendix B.1. Moreover, for the upper bound, Theorem 2 provides a bound that is structurally homologous to, but tighter than, the bound provided in [Zhang *et al.*, 2020]. This is because our VFR can be bounded by $\max_{s,a}|R(s,a)|$, which is also proven in Appendix B.1. Similarly, compared with the policy π for the given MDP $\mathcal{M}$, the factors mainly affecting the performance of the disturbed policy π_ν are TV distance $\max_s D_{\mathrm{TV}}(\pi(\cdot|s), \pi(\cdot|\nu(s))$ and the VFR $\hat{V}_{\mathcal{M},\pi}$. The TV distance, depends on the policy π as well as the disturbance ν, reflecting both the robustness of the policy π and the adversarial ability. However, independent of the adversary, the latter factor (the VFR of the policy), only depends on the value functions of π in $\mathcal{M}$, reflecting the robustness of the policy π. Thus, we can also improve the robustness under observation disturbance of the policy by controlling the VFR of the policy.

3.4 Connection between the Transition and Observation Disturbance

Transition disturbance and observation disturbance are structurally different, as they affect MDP and the observation of the policy respectively. Although existing literature usually considers them separately, by Theorem 1 and Theorem 2, we can find out that their effects on cumulative reward are similar; the similarity depends on the VFR $\hat{V}_{\mathcal{M},\pi}$, which is an inherent property of π and independent of the adversary. Theoretically, if we set $\epsilon_{\mathcal{P}} = \max_{s,a} D_{\mathrm{TV}}(\mathcal{P}(\cdot|s,a), \hat{\mathcal{P}}(\cdot|s,a))$, $\epsilon_{\pi} = \max_s D_{\mathrm{TV}}(\pi(\cdot|s), \pi(\cdot|\nu(s))$ and assume that $\max_{s,a} |\mathcal{R}(s,a)| = 1$, we can naturally deduce

$$\begin{aligned} \left|J_{\mathcal{M}}(\pi) - J_{\hat{\mathcal{M}}}(\pi)\right| &\leq \frac{2\gamma}{1-\gamma}\epsilon_{\mathcal{P}}\hat{V}_{\mathcal{M},\pi} \\ |J_{\mathcal{M}}(\pi) - J_{\mathcal{M}}(\hat{\pi}_\nu)| &\leq \frac{\gamma}{1-\gamma}\epsilon_{\pi}\hat{V}_{\mathcal{M},\pi} + \frac{2}{1-\gamma}\epsilon_{\pi}. \end{aligned} \tag{9}$$

Thus we can improve the robustness of the policy under observation and transition disturbances by controlling $\hat{V}_{\mathcal{M},\pi}$.

4 Methodology

In this section, we first formulate our problem for improving the robustness of the agent and then propose a novel on-policy algorithm of CPPO to solve it.

4.1 Problem Formulation

We first discuss the connection between controlling the VFR $\hat{V}_{\mathcal{M},\pi}$ and CVaR-based RL. For controlling $\hat{V}_{\mathcal{M},\pi}$, it is more reasonable to maximize $\min_s V_{\mathcal{M},\pi}(s)$ rather than minimize $\max_s V_{\mathcal{M},\pi}(s)$ since the latter one contradicts our goal of maximizing cumulative expected return. However, as mentioned in Sec. 2.1, directly maximizing the value function of the worst state may cause our policy to be overly conservative. By the property (5) of CVaR, it is more reasonable to loosen $\min_s V_{\mathcal{M},\pi}(s)$ to $-\mathrm{CVaR}_\alpha(-V(s))$ where $s \sim \mu(\cdot)$ obeys the initial distribution of the environment. Unfortunately, we cannot precisely approximate the value function of every state in practice. Since the return of every trajectory can be calculated exactly, we consider loosening $-\mathrm{CVaR}_\alpha(-V(s))$ by $-\mathrm{CVaR}_\alpha(-D(\pi))$ via Theorem 3.

Theorem 3 (Proof in Appendix B.2)**.** *For any $\alpha \in [0,1]$, we have*

$$-\mathrm{CVaR}_\alpha(-D(\pi)) \leq -\mathrm{CVaR}_\alpha(-V(s)). \tag{10}$$

Therefore, we consider constraining $-\mathrm{CVaR}_\alpha(-D(\pi))$ to improve the VFR of the policy and further improve the robustness of the policy against observation disturbance as well as transition disturbance. Based on this analysis, we define our constrained optimization problem as

$$\max_\theta J(\pi_\theta) \quad s.t. -\mathrm{CVaR}_\alpha(-D(\pi_\theta)) \geq \beta, \tag{11}$$

where α, β are hyper-parameters.

We denote the best policy of problem (11) as $\pi_c(\alpha,\beta)$. Compared with the best policy π_s of the standard RL problem, we obviously have $J(\pi_c(\alpha,\beta)) \leq J(\pi_s)$ since $\pi_c(\alpha,\beta)$ is in a restricted region related to hyper-parameters α, β. We can further give a lower bound of $J(\pi_c(\alpha,\beta))$ as follows:

Algorithm 1 CVaR Proximal Policy Optimization (CPPO)

Require: confidence level α, learning rate $lr_\eta, lr_\theta, lr_\lambda, lr_\phi$
Ensure: parameterized policy π_θ and parameterized value function V_ϕ.
1: **for** $k = 1, 2, ..., N_{iter}$ **do**
2: Generate N trajectories with the current policy π_θ.
3: Compute advantage estimates $\hat{A}_i^t$ of each state $s_{i,t}$ in each trajectory ξ_i and the cumulative reward $D(\xi_i)$.
4: Update parameters $\eta, \theta, \lambda, \phi$ respectively with the calculated gradients.
5: Modify β as a function of current trajectories' return.
6: **end for**

Theorem 4 (Proof in Appendix B.3)**.** *Assume that the discounted return of every trajectory $\tau = (S_0, A_0, R_0, S_1, ...)$ can be bounded by a constant M, i.e., $\sum_{t=0}^{\infty} \gamma^t R_t \leq M$, then we have*

$$J(\pi_c(\alpha,\beta)) \geq \frac{J(\pi_s) - \alpha M}{1-\alpha}.$$

By Theorem 4, we can see that the expected cumulative return of $\pi_c(\alpha,\beta)$ will be no worse than the lower bound although it is optimized in a restricted region.

4.2 Optimization and Algorithm

We now simplify the constrained problem (11) to an unconstrained one. First, with the properties of CVaR, we can equivalently reformulate problem (11) as

$$\min_{\theta,\eta} -J(\pi_\theta) \quad s.t.\ \frac{1}{1-\alpha}\mathbb{E}[(\eta - D(\pi_\theta))^+] - \eta \leq -\beta.$$

The deviation is provided in Appendix B.4. Moreover, by using a Lagrangian relaxation method [Bertsekas, 1997], we need to solve the saddle point of the function $L(\theta,\eta,\lambda)$ as

$$\begin{aligned} &\max_{\lambda\geq 0}\min_{\theta,\eta} L(\theta,\eta,\lambda) \\ &\triangleq -J(\pi_\theta) + \lambda\left(\frac{1}{1-\alpha}\mathbb{E}[(\eta - D(\pi_\theta))^+] - \eta + \beta\right). \end{aligned} \tag{12}$$

To solve problem (12), we extend Proximal Policy Optimization (PPO) [Schulman *et al.*, 2017] with CVaR and name our algorithm CVaR Proximal Policy Optimization (CPPO). In particular, the key point is to calculate gradients [Sutton *et al.*, 2000]. Here, we use methods in [Chow and Ghavamzadeh, 2014] to compute the gradient of our objective function (12) with respect to η, θ, λ as

$$\nabla_\eta L(\theta,\eta,\lambda) = \frac{\lambda}{1-\alpha}\mathbb{E}_{\xi\sim\pi_\theta}\mathbf{1}\{\eta \geq D(\xi)\}) - \lambda$$

$$\begin{aligned} &\nabla_\theta L(\theta,\eta,\lambda) \\ &= -\mathbb{E}_{\xi\sim\pi_\theta}(\nabla_\theta \log P_\theta(\xi))\left(D(\xi) - \frac{\lambda}{1-\alpha}(-D(\xi)+\eta)^+\right) \end{aligned}$$

$$\nabla_\lambda L(\theta,\eta,\lambda) = \frac{1}{1-\alpha}\mathbb{E}_{\xi\sim\pi_\theta}(-D(\xi)+\eta)^+ + \beta - \eta.$$

The detailed calculation is in Appendix B.5. Moreover, with the improvement of policies' performance during training, it is unreasonable to fix β to constrain the risk of the policy.

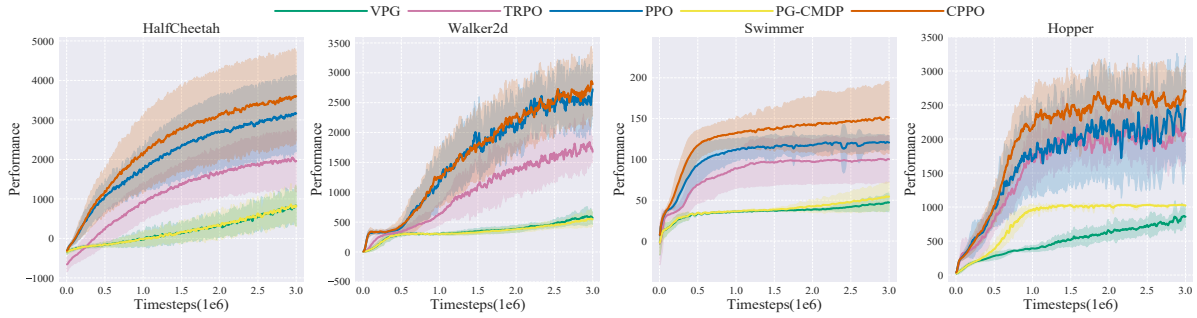

Figure 1: Cumulative reward curves for VPG, TRPO, PPO, PG-CMDP and our CPPO. The x-axes indicate the number of steps interacting with the environment, and the y-axes indicate the performance of the agent, including average rewards with standard deviations.

Method	Ant-v3	HalfCheetah-v3	Walker2d-v3	Swimmer-v3	Hopper-v3
VPG	12.8± 0.0	896.9± 531.1	628.6± 229.4	48.3± 11.3	888.4± 209.5
TRPO	1625.4± 356.4	2073.8± 741.3	2005.6± 398.7	101.2± 29.3	2391.4± 455.3
PPO	3372.2± 301.4	3245.4± 947.3	2946.3± 944.3	122.0± 7.9	2726.0± 886.0
PG-CMDP	7.4 ± 3.6	928.7± 562.9	596.7± 219.9	55.4± 18.8	1039.2± 21.1
CPPO(ours)	**3514.7± 247.2**	**3680.5± 1121.3**	**3194.0± 648.2**	**182.5± 46.0**	**3144.6± 158.4**

Table 1: Cumulative reward (mean ± one std) of best policy trained by VPG, TRPO, PPO, PG-CMDP and CPPO in different MuJoCo games. In each column, we **bold** the best performance over all algorithms.

Thus, we consider modifying β as a function of the risk of trajectories in the current epoch. For example, in CPPO, we set β as the mean value of the expected cumulative return of the worst K trajectories of all N trajectories in the previous epoch, and we will set the ratio K/N to be larger than the ratio in the constraint for reducing the risk. Algorithm 1 outlines the CPPO algorithm, and a more detailed version is in Appendix A.

5 Experiments

In this section, we empirically evaluate the performance and robustness of CPPO under both transition disturbance and observation disturbance in a series of continuous control tasks in MuJoCo [Todorov *et al.*, 2012] against other common on-policy RL algorithms.

5.1 Experiment Setup

Environments. We choose MuJoCo [Todorov *et al.*, 2012] as our experimental environment. As a robotic locomotion simulator, MuJoCo has lots of different continuous control tasks like Ant, HalfCheetah, Walker2d, Swimmer and Hopper, which are widely used for the evaluation of RL algorithms.

Baselines and Codes. We compare our algorithm with common on-policy algorithms and previous CVaR-based algorithms. For the former, we choose Vanilla Policy Gradient (VPG) [Sutton *et al.*, 2000], Trust Region Policy Optimization (TRPO) [Schulman *et al.*, 2015] and PPO [Schulman *et al.*, 2017]. For the latter, we implement PG-CMDP [Chow and Ghavamzadeh, 2014] with a deep neural network. We use Adam [Kingma and Ba, 2015] to optimize all algorithms. The implementation of all code, including CPPO and baselines, are based on the codebase SpinningUp.

Evaluation. First, we compare the cumulative reward of each algorithm in the training process and its performance after convergence. In order to measure the robustness and safety, we compare the performance under transition disturbance and observation disturbance, respectively. For transition perturbation, since MuJoCo is a physical simulation engine and its transition depends on its physical parameters, we choose to modify the mass of the agent to change the transition dynamics, and study the relationship between the agent's performance and the mass of the agent. For observation disturbance, we apply Gaussian disturbance to the agent's observation to study the relationship between the agent's performance and the magnitude of the disturbance.

5.2 Performance in the Training Stage

In this part, we compare the performance in the training stage of our CPPO against common on-policy algorithms as well as the previous CVaR-based algorithm in MuJoCo environments. For each algorithm in each task, we train 10 policies with different random seeds, since the environments and policies are stochastic. For each algorithm in each task, we also plot the mean and variance of the 10 policies as a function of timestep in the training stage, as shown in Figure 1. The solid line represents the average reward of 10 strategies, and the part with a lighter color represents their variance. The final mean and variance of the cumulative return of 10 policies trained by each algorithm in each environment are reported in Table 1. As shown in the figures and the table, for all five tasks, CPPO learns a better policy compared with all baselines even when there is no transition or observation disturbance, especially in HalfCheetah, Swimmer and Hopper. Compared with VPG, TRPO and PG-CMDP, CPPO performs better since we use better policy optimization techniques. Moreover, the

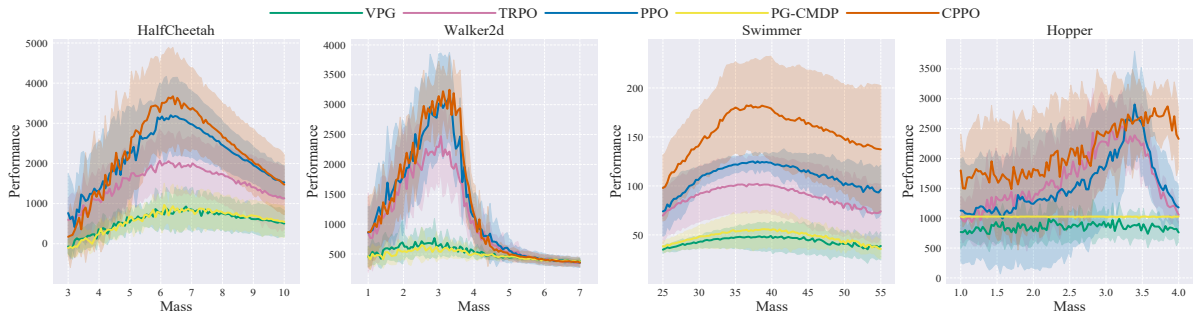

Figure 2: Cumulative reward curves for VPG, TRPO, PPO, PG-CMDP and our CPPO under transition disturbance. The x-axes indicate the mass of the agent, and the y-axes indicate the average performance of the algorithm when the mass changes.

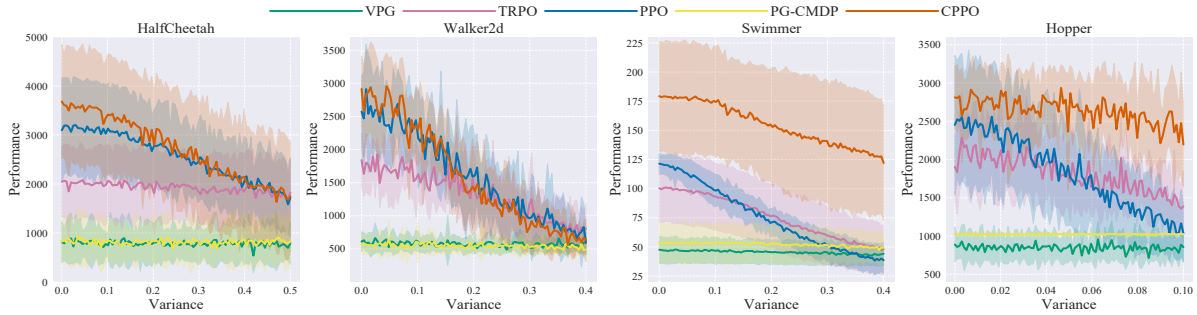

Figure 3: Cumulative reward curves for VPG, TRPO, PPO, PG-CMDP and our CPPO under observation disturbance. The x-axes indicate the range of the disturbance, and the y-axes indicate the average performance of the algorithm under the state disturbance.

performance of CPPO exceeds PPO since penalizing trajectories with relatively low return can also benefit the cumulative return.

5.3 Robustness against Transition Disturbance

An agent may fail in the testing stage because of the transition gap between the simulator and the true environment. In this section, we choose to modify the mass of the robot and test the performance of agents with different transitions, i.e., we change the default mass of HalfCheetah (6.36), Walker2d (3.53), Swimmer (34.6) and Hopper (3.53). Then, we draw Figure 2 to describe the results of agents that are trained under standard mass conditions and tested under different mass conditions. The solid line represents the average reward of 10 strategies, and the part with a lighter color represents their variance. As seen in this figure, the performance of all algorithms decreases to a certain extent with the change of agent quality (whether it becomes larger or smaller). The degree of the decrease is positively correlated with the quality change, which is consistent with our theoretical analysis in Theorem 1 — that is, the upper bound of the performance difference of the algorithm is related to the size of the transition disturbance. Moreover, since the value functions of all states in these policies are relatively low and the VFR of these policies is low, we discover that VPG and PG-CMDP stay robust under transition disturbance since their VFR is low, which is also shown in Theorem 1. At the same time, we can see that CPPO achieves a higher outcome in different tasks, especially in Swimmer and Hopper. This indicates that our method can improve the robustness of policies under transition disturbance since CPPO controls the risk theoretically related to the robustness under transition disturbance.

5.4 Robustness against Observation Disturbance

The agent may also fail in the testing stage because of the gap between its observation and the true state. Consequently, for evaluating the robustness of each algorithm under observation uncertainty, we add a standard Gaussian disturbance to the observation in the testing stage. For this purpose, we plot the performance of the trained policies under observation disturbance in Figure 3. The figure shows that the performance degradation is positively related to the size of the disturbance, which is shown in Theorem 2. Similar to the result under transition disturbance, we can also discover that VPG and PG-CMDP stay robust under observation disturbance since their VFR is low, which is shown in Theorem 2. As shown in the figure, CPPO has made significant progress in Swimmer and Hopper compared to baselines. Therefore, CPPO enables us to maintain robustness under observation disturbance, which is also because CPPO controls the risk theoretically related to the robustness under observation disturbance. We also evaluate their robustness under adversarial attacks of state observations and CPPO shows better robustness than other baselines under adversarial attacks. The detailed results are reported in Appendix C.

6 Conclusions

In this paper, we first provide a theoretical connection between policies' robustness against transition disturbance and observation disturbance, although they are structurally different. Moreover, we analyze the advantages of CVaR for evaluating the uncertainty of policy compared with the worst-case outcome. Based on these analyses, we consider a risk-sensitive optimization objective and propose CPPO to solve it. Extensive experiments on various MuJoCo tasks show that CPPO obtains better performance as well as stronger robustness than various strong competitors.

Ethical Statement

Deep reinforcement learning may encounter catastrophic failures due to uncertainty and it is imperative to develop safe reinforcement learning algorithms. This paper studies the robustness of reinforcement learning algorithms against transition and observation disturbance, which is beneficial for safe and reliable reinforcement learning. There are no serious ethics concerns as this is a basic research.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No.s 2020AAA0106000, 2020AAA0104304, 2020AAA0106302), NSFC Projects (Nos. 62061136001, 61621136008, 62076147, U19B2034, U19A2081, U1811461), the major key project of PCL (No. PCL2021A12), Tsinghua-Huawei Joint Research Program, Tsinghua-Alibaba Joint Research Program, a grant from Tsinghua Institute for Guo Qiang, Tsinghua-OPPO Joint Research Center, Beijing Academy of Artificial Intelligence (BAAI), and the NVIDIA NVAIL Program with GPU/DGX Acceleration.

References

[Alexander and Baptista, 2004] Gordon J Alexander and Alexandre M Baptista. A comparison of var and cvar constraints on portfolio selection with the mean-variance model. *Management Science*, 50(9):1261–1273, 2004.

[Alexander *et al.*, 2006] Siddharth Alexander, Thomas F Coleman, and Yuying Li. Minimizing cvar and var for a portfolio of derivatives. *Journal of Banking & Finance (JBF)*, 30(2):583–605, 2006.

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

[Chow and Ghavamzadeh, 2014] Yinlam Chow and Mohammad Ghavamzadeh. Algorithms for cvar optimization in mdps. In *Proceedings of the 27th International Conference on Neural Information Processing Systems (NeurIPS)*, pages 3509–3517, 2014.

[Chow *et al.*, 2015] Yinlam Chow, Aviv Tamar, Shie Mannor, and Marco Pavone. Risk-sensitive and robust decision-making: a cvar optimization approach. *Advances in Neural Information Processing Systems (NeurIPS)*, 28:1522–1530, 2015.

[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 (JMLR)*, 18(1):6070–6120, 2017.

[Dabney *et al.*, 2018] Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International conference on machine learning*, pages 1096–1105. PMLR, 2018.

[Garcıa and Fernández, 2015] Javier Garcıa and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research (JMLR)*, 16(1):1437–1480, 2015.

[Heger, 1994] Matthias Heger. Consideration of risk in reinforcement learning. In *Machine Learning Proceedings 1994*, pages 105–111. Elsevier, 1994.

[Huang *et al.*, 2017] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. *arXiv preprint arXiv:1702.02284*, 2017.

[Kakade and Langford, 2002] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *In Proc. 19th International Conference on Machine Learning (ICML)*. Citeseer, 2002.

[Kendall *et al.*, 2019] Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley, and Amar Shah. Learning to drive in a day. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8248–8254. IEEE, 2019.

[Kingma and Ba, 2015] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[Ma *et al.*, 2020] Xiaoteng Ma, Li Xia, Zhengyuan Zhou, Jun Yang, and Qianchuan Zhao. Dsac: Distributional soft actor critic for risk-sensitive reinforcement learning. *arXiv preprint arXiv:2004.14547*, 2020.

[Mnih *et al.*, 2015] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[Nilim and El Ghaoui, 2005] Arnab Nilim and Laurent El Ghaoui. Robust control of markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

[Prashanth, 2014] LA Prashanth. Policy gradients for cvar-constrained mdps. In *International Conference on Algorithmic Learning Theory*, pages 155–169. Springer, 2014.

[Rigter *et al.*, 2021] Marc Rigter, Bruno Lacerda, and Nick Hawes. Risk-averse bayes-adaptive reinforcement learning. *Advances in Neural Information Processing Systems*, 34, 2021.

[Schulman *et al.*, 2015] 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. PMLR, 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, Vedavyas Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P. Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

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

[Sutton *et al.*, 2000] 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.

[Tamar *et al.*, 2013] Aviv Tamar, Huan Xu, and Shie Mannor. Scaling up robust mdps by reinforcement learning. *arXiv preprint arXiv:1306.6189*, 2013.

[Tamar *et al.*, 2015] Aviv Tamar, Yonatan Glassner, and Shie Mannor. Optimizing the cvar via sampling. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[Tang *et al.*, 2020] Yichuan Charlie Tang, Jian Zhang, and Ruslan Salakhutdinov. Worst cases policy gradients. In *Conference on Robot Learning*, pages 1078–1093. PMLR, 2020.

[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. IEEE, 2012.

[Wiesemann *et al.*, 2013] Wolfram Wiesemann, Daniel Kuhn, and Berç Rustem. Robust markov decision processes. *Mathematics of Operations Research*, 38(1):153–183, 2013.

[Yang *et al.*, 2021] Qisong Yang, Thiago D Simão, Simon H Tindemans, and Matthijs TJ Spaan. Wcsac: Worst-case soft actor critic for safety-constrained reinforcement learning. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence. AAAI Press, online*, 2021.

[Zhang *et al.*, 2020] Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Mingyan Liu, Duane Boning, and Cho-Jui Hsieh. Robust deep reinforcement learning against adversarial perturbations on state observations. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:21024–21037, 2020.