ON THE ROBUSTNESS OF SAFE REINFORCEMENT LEARNING UNDER OBSERVATIONAL PERTURBATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Safe reinforcement learning (RL) trains a policy to maximize the task reward while satisfying safety constraints. While prior works focus on the performance optimality, we find that the optimal solutions of many safe RL problems are not robust and safe against carefully designed observational perturbations. We formally analyze the unique properties of designing effective observational adversarial attackers in the safe RL setting. We show that baseline adversarial attack techniques for standard RL tasks are not always effective for safe RL and proposed two new approaches - one maximizes the cost and the other maximizes the reward. One interesting and counter-intuitive finding is that the maximum reward attack is strong, as it can both induce unsafe behaviors and make the attack stealthy by maintaining the reward. We further propose a more effective adversarial training framework for safe RL and evaluate it via comprehensive experiments [1]. This paper provides a pioneer work to investigate the safety and robustness of RL under observational attacks for future safe RL studies.

1 INTRODUCTION

Despite the great success of deep reinforcement learning in recent years, it is still challenging to ensure safety when deploying them to safety-critical real-world applications. Safe reinforcement learning (RL) tackles the problem by solving a constrained optimization that can maximize the task reward while satisfying certain constraints (Brunke et al., 2021). This is usually done under the Constrained Markov Decision Processes (CMDPs) framework and has shown to be effective in learning a constraint satisfaction policy in many tasks (Liu et al., 2022). The success of recent safe RL approaches leverages the power of neural networks (Srinivasan et al., 2020; Thananjeyan et al., 2021). However, it has been shown that neural networks are vulnerable to adversarial attacks – a small perturbation of the input data may lead to a large variance of the output (Machado et al., 2021; Pitropakis et al., 2019), which raises a concern when deploying a neural network RL policy to safety-critical applications (Akhtar & Mian, 2018). While many recent safe RL methods with deep policies can achieve outstanding constraint satisfaction in noise-free simulation environments, such a concern regarding their vulnerability under adversarial perturbations has not been studied in the safe RL setting. We consider the observational perturbations that commonly exist in the physical world, such as unavoidable sensor errors and upstream perception inaccuracy (Zhang et al., 2020a).

Several recent works of observational robust RL have shown that deep RL agent could be attacked via sophisticated observation perturbations, drastically decreasing their rewards (Huang et al., 2017; Zhang et al., 2020b). **(R4-NTgs)** However, the robustness concept and adversarial training methods in standard RL settings may not be suitable for safe RL because of an additional metric that characterizes the cost of constraint violations (Brunke et al., 2021). The cost should be more important than the measure of reward, since any constraint violations could be fatal and unacceptable in the real world (Berkenkamp et al., 2017). For example, consider the autonomous vehicle navigation task where the reward is to reach the goal as fast as possible and the safety constraint is to not collide with obstacles, then sacrificing some reward is not comparable with violating the constraint because the latter may cause catastrophic consequences. However, we find little research formally studying the robustness in the safe RL setting with adversarial observation perturbations, while we believe this should be an important aspect in the safe RL area, because **a vulnerable policy under adversarial attacks cannot be regarded as truly safe in the physical world.**

[1] video demos are available at: `https://sites.google.com/view/robustsaferl/home`

We aim to address the following questions in this work: 1) How vulnerable would a learned RL agent be under observational adversarial attacks? 2) How to design effective attackers in the safe RL setting? 3) How to obtain a robust policy that can maintain safety even under worst-case perturbations? To answer them, we formally define the observational robust safe RL problem and discuss how to evaluate the adversary and robustness of a safe RL policy. We also propose two strong adversarial attacks that can induce the agent to perform unsafe behaviors and show that adversarial training can help improve the robustness of constraint satisfaction. We summarize the contributions as follows.

1. As far as we are aware, we are the first to formally analyze the unique vulnerability of the optimal policy in safe RL under observational corruptions. We define the observational-adversarial safe RL problem and investigate its fundamental properties. We show that the optimal solutions of safe RL problems are theoretically vulnerable under observational adversarial attacks.
2. We show that existing adversarial attack algorithms focusing on minimizing agent rewards do not always work, and propose two effective attack algorithms with theoretical justifications – one directly maximizes the constraint violation cost, and one maximizes the task reward to induce a tempting but risky policy. Surprisingly, the maximum reward attack is very strong in inducing unsafe behaviors, both in theory and practice. We believe this property is overlooked as maximizing reward is the optimization goal for standard RL, yet it leads to risky and stealthy attacks to safety constraints. Since this attack can maintain the nominal reward, it may not be detected in practice before catastrophic failures.
3. We propose an adversarial training algorithm with the proposed attackers and show contraction properties of their Bellman operators. Extensive experiments in continuous control tasks show that our method is more robust against adversarial perturbations in terms of constraint satisfaction.

2 Related Work

Safe RL. One type of approach utilizes domain knowledge of the target problem to improve the safety of an RL agent, such as designing a safety filter (Dalal et al., 2018), assuming sophisticated system dynamics model (Berkenkamp et al., 2017; Luo & Ma, 2021; Chen et al., 2021), or incorporating expert interventions (Saunders et al., 2017; Alshiekh et al., 2018). Constrained Markov Decision Process (CMDP) is a commonly used framework to model the safe RL problem, which can be solved via constrained optimization techniques (Garcıa & Fernández, 2015; Sootla et al., 2022; Flet-Berliac & Basu, 2022). The Lagrangian-based method is a generic constrained optimization algorithm to solve CMDP, which introduces additional Lagrange multipliers to penalize constraints violations (Bhatnagar & Lakshmanan, 2012; Chow et al., 2017; As et al., 2022). The multiplier can be optimized via gradient descent together with the policy parameters (Liang et al., 2018; Tessler et al., 2018), and can be easily incorporated into many existing unconstrained RL methods (Ray et al., 2019). Another line of work approximates the non-convex constrained optimization problem with low-order Taylor expansions and then obtains the dual variable via convex optimization (Achiam et al., 2017; Yu et al., 2019; Zhang et al., 2020c; Yang et al., 2020). Since the constrained optimization-based methods are more general, we will focus on the discussions of safe RL upon them.

Robust RL. The robustness definition in the RL context has many interpretations (Moos et al., 2022), including the robustness against action perturbations (Tessler et al., 2019), reward corruptions (Wang et al., 2020; Lin et al., 2020; Eysenbach & Levine, 2021), domain shift (Tobin et al., 2017; Muratore et al., 2018), and dynamics uncertainty (Iyengar, 2005; Wiesemann et al., 2013; Lim et al., 2013; Pinto et al., 2017; Huang et al., 2022). The most related works are investigating the observational robustness of an RL agent under observational adversarial attacks (Zhang et al., 2020a;b). It has been shown that the neural network policies can be easily attacked by adversarial observation noise and thus lead to much lower rewards than the optimal policy (Huang et al., 2017; Kos & Song, 2017; Lin et al., 2017; Pattanaik et al., 2017). However, most of the robust RL approaches model the attack and defense as a zero-sum game regarding the reward, while the robustness regarding safety, i.e., constraint satisfaction for safe RL, has not been formally investigated.

3 Observational Adversarial Attack for Safe RL

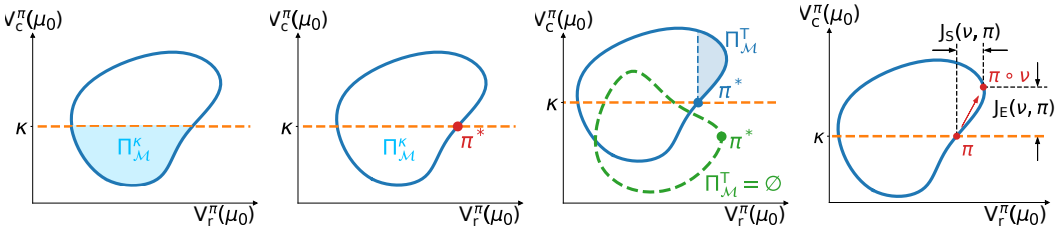
3.1 MDP, CMDP, and the safe RL problem

An infinite horizon Markov Decision Process (MDP) is defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma, \mu_0)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition kernel that specifies the transition probability $p(s_{t+1}|s_t, a_t)$ from state s_t to s_{t+1} under the action a_t, $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the reward function, $\gamma \to [0, 1)$ is the discount factor, and $\mu_0 : \mathcal{S} \to [0, 1]$ is the initial state

distribution. We consider the safe RL modeled under the Constrained Markov Decision Process (CMDP) framework (Altman, 1998), which augments the MDP tuple to $\mathcal{M} := (\mathcal{S}, \mathcal{A}, \mathcal{P}, r, c, \gamma, \mu_0)$ with an additional element $c : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, C_m]$ to characterize the cost for violating the constraint, where C_m is the maximum cost.

We denote a safe RL problem as $\mathcal{M}_\Pi^\kappa := (\mathcal{S}, \mathcal{A}, \mathcal{P}, r, c, \gamma, \mu_0, \Pi, \kappa)$, **(R3-GBM8)** where $\Pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the policy class, and $\kappa \to [0, +\infty)$ is a threshold for constraint violation cost. Let $\pi(a|s) \in \Pi$ denote the policy and $\tau = \{s_0, a_0, ...,\}$ denote the trajectory. We use shorthand $f_t = f(s_t, a_t, s_{t+1}), f \in \{r, c\}$ for simplicity. The value function is $V_f^\pi(\mu_0) = \mathbb{E}_{\tau\sim\pi, s_0\sim\mu_0}[\sum_{t=0}^\infty \gamma^t f_t], f \in \{r, c\}$, which is the expectation of discounted return under the policy π and the initial state distribution μ_0. We overload the notation $V_f^\pi(s) = \mathbb{E}_{\tau\sim\pi, s_0=s}[\sum_{t=0}^\infty \gamma^t f_t], f \in \{r, c\}$ to denote the value function with the initial state $s_0 = s$, and denote $Q_f^\pi(s, a) = \mathbb{E}_{\tau\sim\pi, s_0=s, a_0=a}[\sum_{t=0}^\infty \gamma^t f_t], f \in \{r, c\}$ as the state-action value function under the policy π. The objective of $\mathcal{M}_\Pi^\kappa$ is to find the policy that maximizes the reward while limiting the cost incurred from constraint violations to a threshold κ:

$$\pi^* = \arg\max_\pi V_r^\pi(\mu_0), \quad s.t. \quad V_c^\pi(\mu_0) \le \kappa. \tag{1}$$



(a) Feasibility (b) Optimality (c) Temptation (d) Effectiveness and Stealthiness

Figure 1: Illustration of definitions via a mapping from the policy space to the metric plane $\Pi \to \mathbb{R}^2$, where the x-axis is the reward return and the y-axis is the cost return. A point on the metric plane denotes corresponding policies, i.e., the point (v_r, v_c) represents the policies $\{\pi \in \Pi | V_r^\pi(\mu_0) = v_r, V_c^\pi(\mu_0) = v_c\}$. The blue and green circles denote the policy space of two safe RL problems.

We then define feasibility, optimality and temptation to better describe the properties of a safe RL problem $\mathcal{M}_\Pi^\kappa$. The figure illustration of one example is shown in Fig. 1. Note that although the temptation concept naturally exists in many safe RL settings under the CMDP framework, we did not find formal descriptions or definitions of it in the literature.

Definition 1. Feasibility. The feasible policy class is the set of policies that satisfies the constraint with threshold κ: $\Pi_\mathcal{M}^\kappa := \{\pi(a|s) : V_c^\pi(\mu_0) \le \kappa, \pi \in \Pi\}$. A feasible policy should satisfy $\pi \in \Pi_\mathcal{M}^\kappa$.

Definition 2. Optimality. A policy π^* is optimal in the safe RL context if 1) it is feasible: $\pi^* \in \Pi_\mathcal{M}^\kappa$; 2) no other feasible policy has higher reward return than it: $\forall \pi \in \Pi_\mathcal{M}^\kappa, V_r^{\pi^*}(\mu_0) \ge V_r^\pi(\mu_0)$.

We denote π^* as the optimal policy. Note that the optimality is defined w.r.t. the reward return within the feasible policy class $\Pi_\mathcal{M}^\kappa$ rather than the full policy class space Π, which means that policies that have a higher reward return than π^* may exist in a safe RL problem due to the constraint, and we formally define them as tempting policies because they are rewarding but unsafe:

Definition 3. Temptation. We define the tempting policy class as the set of policies that have a higher reward return than the optimal policy: $\Pi_\mathcal{M}^T := \{\pi(a|s) : V_r^\pi(\mu_0) > V_r^{\pi^*}(\mu_0), \pi \in \Pi\}$. **A tempting safe RL problem** has a non-empty tempting policy class: $\Pi_\mathcal{M}^T \ne \emptyset$.

We show that all the tempting policies are not feasible (proved by contradiction in Appendix A.1):

Lemma 1. *The tempting policy class and the feasible policy class are disjoint:* $\Pi_\mathcal{M}^T \cap \Pi_\mathcal{M}^\kappa = \emptyset$. *Namely, all the tempting policies violate the constraint:* $\forall \pi \in \Pi_\mathcal{M}^T, V_c^\pi(\mu_0) > \kappa$.

The existence of tempting policies is a unique feature, and one of the major challenges of safe RL since the agent needs to maximize the reward carefully to avoid being tempted. One can always tune the threshold κ to change the temptation status of a safe RL problem with the same CMDP. In this paper, we only consider the solvable **tempting** safe RL problems (i.e., the problems with a non-empty feasible class and a non-empty tempting class) because **(R1-yCc1)** otherwise, the non-tempting safe RL problem $\mathcal{M}_\Pi^\kappa$ can be reduced to a standard RL problem – an optimal policy could be obtained by maximizing the reward without considering the constraint.

3.2 SAFE RL UNDER OBSERVATIONAL PERTURBATIONS

We introduce a deterministic **observational** adversary $\nu(s) : \mathcal{S} \to \mathcal{S}$ which corrupts the state observation of the agent. We denote the corrupted observation as $\tilde{s} := \nu(s)$ and the corrupted policy as $\pi \circ \nu := \pi(a|\tilde{s}) = \pi(a|\nu(s))$, as the state is first contaminated by ν and then used by the operator π. Note that the adversary does **not** modify the original CMDP and true states in the environment, but only the input of the agent. This setting mimics realistic scenarios, for instance, the adversary could be the noise from the sensing system or the errors from the upstream perception system.

Different from standard RL, safe RL has to ensure constraint satisfaction, since the cost of violating constraints in many safety-critical applications can be unaffordable. In addition, the reward metric is usually used to measure the agent's performance in finishing a task, so significantly reducing the task reward may warn the agent of the existence of attacks. As a result, **(R4-NTgs)** a strong adversary in the safe RL setting aims to generate more constraint violations while maintaining high rewards to make the attack stealthy (i.e., safety violation). In contrast, existing adversaries on standard RL aim to reduce the overall reward. This is one of our main contributions, which connects existing observational robustness with safety in RL. Concretely, we define two metrics to evaluate the adversary performance for safe RL:

Definition 4. (Attack) Effectiveness $J_E(\nu, \pi)$ is defined as the increased cost value under the adversary: $J_E(\nu, \pi) = V_c^{\pi \circ \nu}(\mu_0) - V_c^{\pi}(\mu_0)$. An adversary ν is effective if $J_E(\nu, \pi) > 0$.

(R4-NTgs) The effectiveness metric measures an adversary's capability of attacking the safe RL agent to violate constraints. We additionally introduce another metric to characterize the adversary's stealthiness w.r.t. the task reward in the safe RL setting.

Definition 5. (Reward) Stealthiness $J_S(\nu, \pi)$ is defined as the increased reward value under the adversary: $J_S(\nu, \pi) = V_r^{\pi \circ \nu}(\mu_0) - V_r^{\pi}(\mu_0)$. An adversary ν is stealthy if $J_S(\nu, \pi) \geq 0$.

Note that the stealthiness concept is widely used in supervised learning (Sharif et al., 2016; Pitropakis et al., 2019). It usually means that the adversarial attack should be covert to human eyes regarding the input data so that it can hardly be identified (Machado et al., 2021). While the stealthiness regarding the perturbation range is naturally satisfied based on the perturbation set definition, we introduce another level of stealthiness in terms of the task reward in the safe RL task. In some situations, the agent might easily detect a dramatic reward drop. A more stealthy attack is maintaining the agent's task reward while increasing constraint violations; see Appendix B.1 for more discussions.

In practice, the power of the adversary is usually restricted (Madry et al., 2017; Zhang et al., 2020a), such that the perturbed observation will be limited within a pre-defined perturbation set $B(s)$: $\forall s \in \mathcal{S}, \nu(s) \in B(s)$. Following convention, we define the perturbation set $B_p^{\epsilon}(s)$ as the ℓ_p-ball around the original observation: $\forall s' \in B_p^{\epsilon}(s), \|s' - s\|_p \leq \epsilon$, where ϵ is the ball size.

3.3 VULNERABILITY OF AN OPTIMAL POLICY UNDER ADVERSARIAL ATTACKS

Given an optimal policy π^* of a tempting safe RL problem, we aim to design strong adversaries such that they are effective in making the agent unsafe and keep reward stealthiness in some applications. Motivated by Lemma 1, we propose the following **Maximum Reward (MR) attacker** that corrupts the observation of a policy π by maximizing the reward value: $\nu_{\text{MR}} = \arg\max_{\nu} V_r^{\pi \circ \nu}(\mu_0)$

Proposition 1. *For an optimal policy π^*, the MR attacker is guaranteed to be reward stealthy and effective, given enough large perturbation set $B_p^{\epsilon}(s)$ such that $V_r^{\pi^* \circ \nu_{MR}} > V_r^{\pi^*}$.*

The MR attacker is counter-intuitive because it is exactly the goal for standard RL.This is an interesting phenomenon worthy of highlighting since we observe that the MR attacker effectively makes the optimal policy unsafe and retains stealthy regarding the reward in the safe RL setting. The proof is given in Appendix A.1. **(R3-GBM8)** If we enlarge the policy space from $\Pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ to an augmented space $\bar{\Pi} : \mathcal{S} \times \mathcal{A} \times \mathcal{O} \to [0, 1]$, where $\mathcal{O} = \{0, 1\}$ is the space of indicator, we can further observe the following important property for the optimal policy:

Lemma 2. ***(R3-GBM8)*** *In augmented policy space $\bar{\Pi}$, the optimal policy π^* of a tempting safe RL problem satisfies: $V_c^{\pi^*}(\mu_0) = \kappa$.*

The proof is given in Appendix A.2. **(R3-GBM8)** $\bar{\Pi}$ is a superset of original space. The definition of the augmented policy space is commonly used in hierarchical RL and can be viewed as a subset of option-based RL (Sutton et al., 1999; Riemer et al., 2018; Zhang & Whiteson, 2019). Note that

Lemma 2 holds in expectation rather than for a single trajectory. It suggests that the optimal policy in a tempting safe RL problem will be vulnerable as it is on the safety boundary, which motivates us to propose the **Maximum Cost (MC) attacker** that corrupts the observation of a policy π by maximizing the cost value: $\nu_{\text{MC}} = \arg\max_\nu V_c^{\pi\circ\nu}(\mu_0)$

It is apparent to see that the MC attacker is effective w.r.t. the optimal policy with a large enough perturbation range, since we directly solve the adversarial observation such that it can maximize the constraint violations. Therefore, as long as ν_{MC} can lead to a policy that has a higher cost return than π^*, it is guaranteed to be effective in making the agent violate the constraint based on Lemma 2.

(R2-7p6w) Practically, given a fixed policy π and its critics $Q_f^\pi(s,a), f \in \{r,c\}$, we obtain the corrupted observation $\tilde{s}$ of s from the MR and MC attackers by solving:

$$\nu_{\text{MR}}(s) = \arg\max_{\tilde{s}\in B_p^\epsilon(s)} \mathbb{E}_{\tilde{a}\sim\pi(a|\tilde{s})}\left[Q_r^\pi(s,\tilde{a}))\right], \quad \nu_{\text{MC}}(s) = \arg\max_{\tilde{s}\in B_p^\epsilon(s)} \mathbb{E}_{\tilde{a}\sim\pi(a|\tilde{s})}\left[Q_c^\pi(s,\tilde{a}))\right] \quad (2)$$

Suppose the policy π and the critics Q are all parametrized by differentiable models such as neural networks, then we can back-propagate the gradient through Q and π to solve the adversarial observation $\tilde{s}$. This is similar to the policy optimization procedure in TD3 (Fujimoto et al., 2018) and DDPG (Lillicrap et al., 2015), whereas we replace the optimization domain from the policy parameter space to the observation space $B_p^\epsilon(s)$. The attacker implementation details can be found in Appendix C.1.

3.4 Theoretical analysis of adversarial attacks

Theorem 1 (Existence of optimal and deterministic MC/MR attackers)**.** *A deterministic MC attacker ν_{MC} and a deterministic MR attacker ν_{MR} always exist, and there is no stochastic adversary ν' such that $V_c^{\pi\circ\nu'}(\mu_0) > V_c^{\pi\circ\nu_{MC}}(\mu_0)$ or $V_r^{\pi\circ\nu'}(\mu_0) > V_r^{\pi\circ\nu_{MR}}(\mu_0)$.*

Theorem 1 provides the theoretical foundation of Bellman operators that require optimal and deterministic adversaries in the next section. The proof is given in Appendix A.3. We can also obtain the upper-bound of constraint violations of the adversary attack at state s. Denote $\mathcal{S}_c$ as the set of unsafe states that have non-zero cost: $\mathcal{S}_c := \{s' \in \mathcal{S} : c(s,a,s') > 0\}$ and p_s as the maximum probability of entering unsafe states from state s: $p_s = \max_a \sum_{s'\in\mathcal{S}_c} p(s'|s,a)$.

Theorem 2 (One-step perturbation cost value bound)**.** *Suppose the optimal policy is locally L-Lipschitz continuous at state s: $D_{\text{TV}}[\pi(\cdot|s')\|\pi(\cdot|s)] \le L\left\|s'-s\right\|_p$, and the perturbation set of the adversary $\nu(s)$ is an ℓ_p-ball $B_p^\epsilon(s)$. Let $\tilde{V}_c^{\pi,\nu}(s) = \mathbb{E}_{a\sim\pi(\cdot|\nu(s)),s'\sim p(\cdot|s,a)}[c(s,a,s') + \gamma V_c^\pi(s')]$ denote the cost value for only perturbing state s. The upper bound of $\tilde{V}_c^{\pi,\nu}(s)$ is given by:*

$$\tilde{V}_c^{\pi,\nu}(s) - V_c^\pi(s) \le 2L\epsilon\left(p_s C_m + \frac{\gamma C_m}{1-\gamma}\right). \quad (3)$$

Note that $\tilde{V}_c^{\pi,\nu}(s) \neq V_c^\pi(\nu(s))$ because the next state s' is still transited from the original state s, i.e., $s' \sim p(\cdot|s,a)$ instead of $s' \sim p(\cdot|\nu(s),a)$. Theorem 2 indicates that the power of an adversary is controlled by the policy smoothness L and perturbation range ϵ. In addition, the p_s term indicates that a safe policy should keep a safe distance from the unsafe state to prevent it from being attacked. We further derive the upper bound of constraint violation for attacking the entire episodes.

Theorem 3 (Episodic perturbation cost value bound)**.** *Given a feasible policy $\pi \in \Pi_{\mathcal{M}}^\kappa$, suppose the L-Lipschitz continuity holds globally for π, and the perturbation set of ν is within an ℓ_p-ball, then the following bound holds:*

$$V_c^{\pi\circ\nu}(\mu_0) \le \kappa + 2L\epsilon C_m\left(\frac{1}{1-\gamma} + \frac{4\gamma L\epsilon}{(1-\gamma)^2}\right)\left(\max_s p_s + \frac{\gamma}{1-\gamma}\right). \quad (4)$$

See Theorem 2, 3 proofs in Appendix A.4, A.5. We can still observe that the maximum cost value under perturbations is bounded by the Lipschitzness of the policy and the maximum perturbation range ϵ. The bound is tight since when $\epsilon \to 0$ (no attack) or $L \to 0$ (constant policy $\pi(\cdot|s)$ for all states), the RHS is 0 for Eq. (3) and κ for Eq. (4), which means that the attack is ineffective.

4 Observational Robust Safe RL

4.1 Adversarial training against observational perturbations

To defend against observational attacks, we propose an adversarial training method for safe RL. We directly optimize the policy upon the corrupted sampling trajectories $\tilde{\tau} = \{s_0, \tilde{a}_0, s_1, \tilde{a}_1, ...\}$, where

$\tilde{a}_t \sim \pi(a|\nu(s_t))$. **(R4-NTgs)** We can compactly represent the adversarial safe RL objective under observational perturbation as:

$$\pi^* = \arg\max_{\pi} V_r^{\pi\circ\nu}(\mu_0), \quad s.t. \quad V_c^{\pi\circ\nu}(\mu_0) \leq \kappa, \quad \forall \nu. \tag{5}$$

The adversarial training objective (5) can be solved by many policy-based safe RL methods, such as the primal-dual approach (Ray et al., 2019), and we show that the Bellman operator for evaluating the policy performance under a deterministic adversary is a contraction (see Appendix A.6 for proof).

Theorem 4 (Bellman contraction)**.** *Define the Bellman policy operator as* $\mathcal{T}_\pi : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$*:*

$$(\mathcal{T}_\pi V_f^{\pi\circ\nu})(s) = \sum_{a\in\mathcal{A}} \pi(a|\nu(s)) \sum_{s'\in\mathcal{S}} p(s'|s,a) \left[f(s,a,s') + \gamma V_f^{\pi\circ\nu}(s') \right], \quad f \in \{r, c\}. \tag{6}$$

The Bellman equation can be written as $V_f^{\pi\circ\nu}(s) = (\mathcal{T}_\pi V_f^{\pi\circ\nu})(s)$*. In addition, the operator* $\mathcal{T}_\pi$ *is a contraction under the sup-norm* $\|\cdot\|_\infty$ *and has a fixed point.*

Theorem 4 shows that we can accurately evaluate the task performance (reward return) and the safety performance (cost return) of a policy under one fixed deterministic adversary, which is similar to solving a standard CMDP. The Bellman contraction property provides the theoretical justification of adversarial training, i.e., training a safe RL agent under observational perturbed sampling trajectories. Then the key part is selecting proper adversaries during learning, such that the trained policy is robust and safe against any other attackers. We show that performing adversarial training with the MC or the MR attacker will enable the agent to be robust against the most effective or the most reward stealthy perturbations, respectively. We have the following proposition (see Appendix A.6 for proof):

Proposition 2. *Suppose a trained policy* π' *under the MC attacker satisfies:* $V_c^{\pi'\circ\nu_{MC}}(\mu_0) \leq \kappa$*, then* $\pi' \circ \nu$ *is guaranteed to be feasible with any* B_p^ϵ *bounded adversarial perturbations.*

Similarly, suppose a trained policy π' *under the MR attacker satisfies:* $V_c^{\pi'\circ\nu_{MR}}(\mu_0) \leq \kappa$*, then* $\pi' \circ \nu$ *is guaranteed to be non-tempting with any* B_p^ϵ *bounded adversarial perturbations.*

Proposition 2 indicates that by solving the adversarial constrained optimization problem under the MC attacker, all the feasible solutions will be safe under any bounded adversarial perturbations. It also shows a nice property for training a robust policy, since the `max` operation over the reward in the safe RL objective may lead the policy to the tempting policy class, while the adversarial training with MR attacker can naturally keep the trained policy at a safe distance from the tempting policy class. Practically, we observe that both MC and MR attackers can increase the robustness and safety via adversarial training, and could be easily plugged into any on-policy safe RL algorithms, in principle. We leave the robust training framework for off-policy safe RL methods as future work.

4.2 Practical Implementation

The meta adversarial training algorithm is shown in Algo. 1. We particularly adopt the primal-dual methods (Ray et al., 2019; Stooke et al., 2020) that are widely used in the safe RL literature as the `learner`, then the adversarial training objective in Eq. (5) can be converted to a min-max form by using the Lagrange multiplier λ:

Algorithm 1 Adversarial safe RL training meta algorithm

Input: Safe RL `learner`, Adversary `scheduler`
Output: Observational robust policy π

1: Initialize policy $\pi \in \Pi$ and adversary $\nu : \mathcal{S} \to \mathcal{S}$
2: **for** each training epoch $n = 1, ..., N$ **do**
3: Rollout trajectories: $\tilde{\tau} = \{s_0, \tilde{a}_0, ...\}_T$, $\tilde{a}_t \sim \pi(a|\nu(s_t))$
4: Run safe RL learner: $\pi \leftarrow$ `learner`$(\tilde{\tau}, \Pi)$
5: Update adversary: $\nu \leftarrow$ `scheduler`$(\tilde{\tau}, \pi, n)$
6: **end for**

$$(\pi^*, \lambda^*) = \min_{\lambda\geq 0} \max_{\pi\in\Pi} V_r^{\pi\circ\nu}(\mu_0) - \lambda(V_c^{\pi\circ\nu}(\mu_0) - \kappa) \tag{7}$$

Solving the inner maximization (primal update) via any policy optimization methods and the outer minimization (dual update) via gradient descent iteratively yields the Lagrangian algorithm. Under proper learning rates and bounded noise assumptions, the iterates (π_n, λ_n) converge to a fixed point (a local minimum) almost surely (Tessler et al., 2018; Paternain et al., 2019).

Based on previous theoretical analysis, we adopt MC or MR as the adversary when sampling trajectories. **(R2-7p6w)** The `scheduler` aims to train the reward and cost Q-value functions for

the MR and the MC attackers, because many on-policy algorithms such as PPO do not provide them. In addition, the scheduler can update the power of the adversary based on the learning progress accordingly, since a strong adversary at the beginning may prohibit the `learner` from exploring the environment and thus corrupt the training. We gradually increase the perturbation range ϵ along with the training epochs to adjust the adversary perturbation set B_p^ϵ, such that the agent will not be too conservative in the early stage of training. A similar idea is also used in adversarial training (Salimans et al., 2016; Arjovsky & Bottou, 2017; Gowal et al., 2018) and curriculum learning literature (Dennis et al., 2020; Portelas et al., 2020). See more implementation details in Appendix C.3.

5 EXPERIMENT

In this section, we aim to answer the questions raised in Sec. 1. To this end, we adopt the robot locomotion continuous control tasks that are easy to interpret, motivated by safety, and used in many previous works (Achiam et al., 2017; Chow et al., 2019; Zhang et al., 2020c). The simulation environments are from a public available benchmark (Gronauer, 2022). We consider two tasks, and train multiple different robots (Car, Drone, Ant) for each task:

Run task. Agents are rewarded for running fast between two safety boundaries and are given costs for violation constraints if they run across the boundaries or exceed an agent-specific velocity threshold. The tempting policies can violate the velocity constraint to obtain more rewards.

Circle task. The agents are rewarded for running in a circle in a clockwise direction but are constrained to stay within a safe region that is smaller than the radius of the target circle. The tempting policies in this task will leave the safe region to gain more rewards.

We name each task via the `Robot-Task` format, for instance, `Car-Run`. More detailed descriptions and video demos are available on our anonymous project website [2]. In addition, we will use the PID PPO-Lagrangian (abbreviated as PPOL) method (Stooke et al., 2020) as the base safe RL algorithm to fairly compare different robust training approaches, while the proposed adversarial training can be easily used in other on-policy safe RL methods as well. The detailed hyperparameters of the adversaries and safe RL algorithms can be found in Appendix C.

5.1 ADVERSARIAL ATTACKER COMPARISON

We first demonstrate the vulnerability of the optimal safe RL policies without adversarial training and compare the performance of different adversaries. All the adversaries have the same ℓ_∞ norm perturbation set B_∞^ϵ restriction. We adopt three adversary baselines, including one improved version:

Random attacker baseline. This is a simple baseline by sampling the corrupted observations randomly within the perturbation set via a uniform distribution.

Maximum Action Difference (MAD) attacker baseline. The MAD attacker (Zhang et al., 2020a) is designed for standard RL tasks, which is shown to be effective in decreasing a trained RL agent's reward return. The optimal adversarial observation is obtained by maximizing the KL-divergence between the corrupted policy: $\nu_{\text{MAD}}(s) = \arg\max_{\tilde{s}\in B_p^\epsilon(s)} D_{\text{KL}}\left[\pi(a|\tilde{s})\|\pi(a|s)\right]$

Adaptive MAD (AMAD) attacker baseline. Since the vanilla MAD attacker is not designed for safe RL, we further improve it to an adaptive version as a stronger baseline. The motivation comes from Lemma 2 – the optimal policy will be close to the constraint boundary that with high risks. To better understand this property, we introduce the discounted future state distribution $d^\pi(s)$ (Kakade, 2003), which allows us to rewrite the result in Lemma 2 as (see Appendix C.6 for derivation and implementation details): $\frac{1}{1-\gamma}\int_{s\in\mathcal{S}} d^{\pi^*}(s)\int_{a\in\mathcal{A}}\pi^*(a|s)\int_{s'\in\mathcal{S}} p(s'|s,a)c(s,a,s')ds'dads = \kappa$. We can see that performing MAD attack for the optimal policy π^* in low-risk regions that with small $p(s'|s,a)c(s,a,s')$ values may not be effective. Therefore, AMAD only perturbs the observation when the agent is within high-risk regions that are determined by the cost value function and a threshold ξ to achieve more effective attacks: $\nu_{\text{AMAD}}(s) := \begin{cases}\nu_{\text{MAD}}(s), & \text{if } V_c^\pi(s)\geq\xi,\\ s, & \text{otherwise .}\end{cases}$

Experiment setting. We evaluate the performance of all three baselines above and our MC, MR adversaries by attacking well-trained PPO-Lagrangian policies in different tasks. The trained policies can achieve nearly zero constraint violation costs without observational perturbations. We keep the trained model weights and environment seeds fixed for all the attackers to ensure fair comparisons.

[2] `https://sites.google.com/view/robustsaferl/home`

Experiment result. Fig. 2 shows the attack results of the 5 adversaries on PPOL-vanilla. Each column corresponds to an environment. The first row is the episode reward and the second row is the episode cost of constraint violations. We can see that the vanilla safe RL policies are vulnerable, since the safety performance deteriorates (cost increases) significantly even with a small adversarial perturbation range ϵ. Generally, we can see an increasing cost trend as the ϵ increases, except for the MAD attacker. Although MAD can reduce the agent's reward quite well, it fails to perform an effective attack in increasing the cost because the reward decrease may keep the agent away from high-risk regions. It is even worse than the random attacker in the `Car-Circle` task. The improved AMAD attacker is a stronger baseline than MAD, as it only attacks in high-risk regions and thus has a higher chance of entering unsafe regions to induce more constraint violations. More comparisons between MAD and AMAD can be found in Appendix C.9. Our proposed MC and MR attackers outperform all baselines attackers (Random, MAD and AMAD) in terms of effectiveness by increasing the cost by a large margin in most tasks. Surprisingly, the MR attacker can achieve even higher costs than MC and is more stealthy as it can maintain or increase the reward well, which validates our theoretical analysis and the existence of tempting policies.

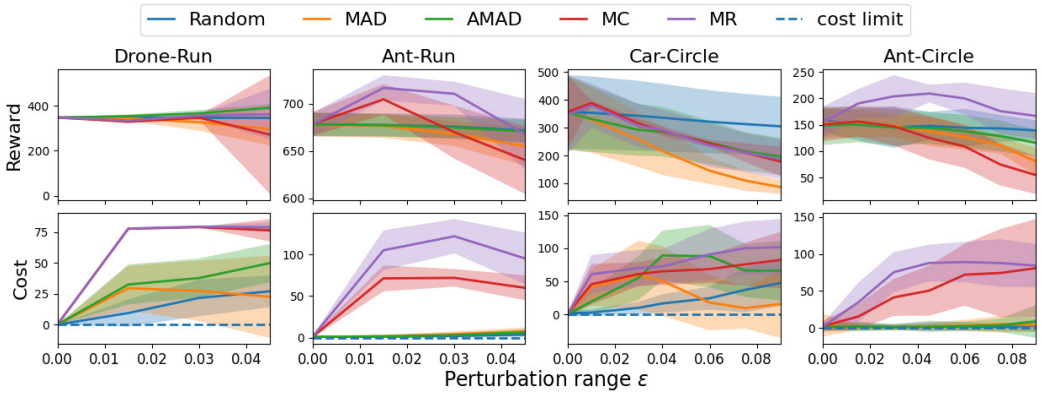


Figure 2: Reward and cost curves of all 5 attackers evaluated on well-trained vanilla PPO-Lagrangian models w.r.t. the perturbation range ϵ. The curves are averaged over 50 episodes and 5 seeds, where the solid lines are the mean and the shadowed areas are the standard deviation. The dashed line is the cost without perturbations.

5.2 Performance of Safe RL with Adversarial Training

We adopt 5 baselines, including the **PPOL-vanilla** method without robust training, the naive adversarial training under random noise **PPOL-random**, the state-adversarial algorithm **SA-PPOL** proposed in (Zhang et al., 2020a), but we extend their PPO in standard RL setting to PPOL in the safe RL setting. The original SA-PPOL algorithm utilizes the MAD attacker to compute the adversarial observations, and then adds a KL regularizer to penalize the divergence between them and the original observations. We add two additional baselines **SA-PPOL(MC)** and **SA-PPOL(MR)** for the ablation study, where we change the MAD attacker to our proposed MC and MR adversaries. Our adversarial training methods are named as **ADV-PPOL(MC)** and **ADV-PPOL(MR)**, which are trained under the MC and MR attackers respectively. We use the same PPOL implementation and hyperparameters for all methods for fair comparisons. More details can be found in Appendix C.5-C.8.

Results. The evaluation results of different trained policies under adversarial attacks are shown in Table 1, where **Natural** represents the performance without noise. We train each algorithm with 5 random seeds and evaluate each trained policy with 50 episodes under each attacker to obtain the values. The training and testing perturbation range ϵ is the same. We use gray shadows to highlight the top two safest agents with the smallest cost values, but we ignore the failure agents whose rewards are less than 50% of the PPOL-vanilla method. We mark the failure agents with $\star$. Due to the page limit, we leave the evaluation results under random and MAD attackers to Appendix C.9.

Analysis. We can observe that although most baselines can achieve near zero natural cost, their safety performances are vulnerable under the strong MC and MR attackers, which are more effective than AMAD in inducing unsafe behaviors. The proposed adversarial training methods (ADV-PPOL) consistently outperform baselines in safety with the lowest costs while maintaining high rewards in most tasks. The comparison with PPOL-random indicates that the MC and MR attackers are essential ingredients of adversarial training. Although SA-PPOL agents can maintain reward very well, they are not safe as to constraint satisfaction under adversarial perturbations in most environments. The

Table 1: Evaluation results of natural performance (no attack) and under 3 attackers. Our methods are ADV-PPOL(MC/MR). Each value is reported as: mean ± standard deviation for 50 episodes and 5 seeds. We shadow two lowest-costs agents under each attacker column and break ties based on rewards, excluding the failing agents (whose natural rewards are less than 50% of PPOL-vanilla's). We mark the failing agents with ⋆.

Env	Method	Natural		AMAD		MC		MR	
		Reward	Cost	Reward	Cost	Reward	Cost	Reward	Cost
Car-Run $\epsilon = 0.05$	PPOL-vanilla	561.33±1.97	0.15±0.36	548.04±17.72	13.49±30.93	624.49±8.87	184.09±0.52	624.46±8.86	184.06±0.47
	PPOL-random	556.84±1.87	0.01±0.08	550.66±1.71	1.73±2.05	584.18±2.69	183.79±0.73	585.79±2.01	183.91±0.59
	SA-PPOL	545.87±2.13	0.0±0.0	546.95±2.11	0.0±0.0	571.14±1.5	6.44±7.28	571.07±1.33	1.77±3.31
	SA-PPOL(MC)	552.58±3.84	0.0±0.0	541.58±3.6	0.0±0.0	568.96±1.92	1.17±2.07	569.6±1.51	0.67±1.24
	SA-PPOL(MR)	543.0±1.14	0.0±0.0	537.19±1.47	0.0±0.0	568.32±1.95	16.27±23.2	568.29±2.12	12.23±17.26
	ADV-PPOL(MC)	525.76±2.99	0.0±0.0	516.22±3.52	0.0±0.0	555.64±3.44	0.05±0.21	554.48±2.78	0.01±0.08
	ADV-PPOL(MR)	525.93±2.28	0.0±0.0	514.97±2.68	0.0±0.0	557.38±2.83	0.06±0.24	556.87±3.03	0.05±0.22
Drone-Run $\epsilon = 0.025$	PPOL-vanilla	347.17±1.53	0.0±0.0	362.18±11.24	35.69±17.74	336.05±9.43	79.0±0.0	345.64±5.22	79.0±0.0
	PPOL-random	343.71±1.55	0.0±0.0	361.85±18.03	65.58±22.58	268.28±4.26	0.9±2.28	317.25±40.34	33.66±29.86
	SA-PPOL	284.47±32.13	0.0±0.0	306.55±26.44	11.85±18.04	156.97±384.11	61.88±18.49	403.11±163.75	75.73±7.71
	⋆ SA-PPOL(MC)	174.61±34.37	0.06±0.73	86.35±56.22	0.0±0.0	205.34±29.57	10.13±10.67	217.51±24.77	8.31±6.05
	⋆ SA-PPOL(MR)	0.13±0.22	0.0±0.0	0.11±0.21	0.0±0.0	0.25±0.37	0.0±0.0	0.28±0.43	0.0±0.0
	ADV-PPOL(MC)	273.4±16.98	0.0±0.0	268.0±12.0	0.05±0.57	275.0±28.26	1.1±3.17	294.8±23.67	18.11±25.87
	ADV-PPOL(MR)	233.31±20.68	0.0±0.0	238.0±22.15	0.0±0.0	229.8±68.0	6.81±8.38	238.11±46.17	0.95±1.92
Ant-Run $\epsilon = 0.025$	PPOL-vanilla	678.4±12.64	1.23±1.4	676.23±12.27	2.68±2.16	661.3±58.17	66.41±14.07	706.32±18.83	112.33±25.57
	PPOL-random	673.42±14.47	1.01±1.06	670.6±13.59	1.9±1.47	661.47±10.02	45.94±10.2	670.85±18.73	46.97±11.63
	SA-PPOL	658.83±14.14	0.46±0.82	658.42±13.96	0.66±0.87	668.14±25.7	67.68±20.17	694.86±11.05	87.1±20.78
	SA-PPOL(MC)	574.36±25.69	3.03±3.15	574.85±26.37	3.16±3.48	604.77±30.51	21.39±10.83	619.4±31.35	32.87±12.69
	⋆ SA-PPOL(MR)	90.49±60.14	5.33±4.27	77.64±84.93	5.17±4.24	77.99±72.79	6.33±4.87	69.93±96.51	6.17±4.87
	ADV-PPOL(MC)	601.25±18.6	0.0±0.0	599.31±18.34	0.0±0.0	666.73±15.21	1.1±1.02	665.47±18.29	1.75±1.54
	ADV-PPOL(MR)	620.17±27.28	0.17±0.41	618.04±24.66	0.31±0.55	634.96±14.94	4.07±2.35	648.95±17.67	4.69±2.81
Car Circle $\epsilon = 0.05$	PPOL-vanilla	337.69±152.34	1.8±3.91	274.61±78.92	92.53±39.32	265.61±12.43	69.33±18.91	238.06±101.01	74.47±38.44
	PPOL-random	398.71±48.96	0.17±0.9	293.77±105.83	69.97±46.75	307.77±32.95	59.3±26.29	295.2±45.56	49.73±22.98
	SA-PPOL	403.92±22.63	0.4±2.15	382.8±20.57	0.37±1.97	361.0±12.96	109.1±6.0	452.98±26.72	89.03±9.13
	SA-PPOL(MC)	417.78±17.79	0.33±1.8	314.13±27.73	0.0±0.0	355.12±13.42	98.43±14.52	468.1±14.41	87.5±9.17
	SA-PPOL(MR)	389.03±47.53	0.2±1.0	351.49±34.16	0.14±0.69	342.67±39.23	77.9±21.57	414.87±66.09	75.68±20.73
	ADV-PPOL(MC)	302.3±12.24	0.1±0.7	296.23±19.02	1.86±5.49	310.37±25.68	1.12±3.98	261.52±24.51	0.28±1.59
	ADV-PPOL(MR)	309.42±35.45	0.0±0.0	321.44±20.52	6.66±10.94	258.52±31.53	0.08±0.56	308.6±54.7	0.16±1.12
Drone Circle $\epsilon = 0.025$	PPOL-vanilla	627.49±55.24	0.27±1.12	527.6±171.54	34.5±36.73	228.79±181.92	95.17±59.23	85.79±159.67	174.4±81.58
	PPOL-random	604.31±46.83	0.37±1.97	559.2±173.25	27.67±32.66	159.16±184.15	91.5±98.26	130.08±146.67	103.1±92.03
	SA-PPOL	503.13±19.89	0.0±0.0	496.34±20.54	0.0±0.0	430.64±89.64	97.57±27.47	346.99±320.08	109.5±78.1
	SA-PPOL(MC)	347.43±97.49	8.5±35.32	346.25±41.68	0.0±0.0	329.05±143.47	58.77±34.94	380.53±176.19	78.07±60.05
	⋆ SA-PPOL(MR)	184.7±128.7	11.94±43.67	189.76±118.14	15.38±47.38	189.18±142.46	44.62±35.83	219.87±138.35	49.14±52.87
	ADV-PPOL(MC)	359.02±33.01	0.0±0.0	351.57±52.5	1.48±6.44	399.78±69.47	4.16±12.57	356.09±90.42	9.66±28.48
	ADV-PPOL(MR)	356.6±46.91	0.0±0.0	339.04±72.43	5.36±23.08	275.43±95.08	5.66±22.41	379.52±87.22	1.2±6.47
Ant Circle $\epsilon = 0.025$	PPOL-vanilla	157.44±26.21	2.7±6.02	143.37±36.86	3.23±9.87	153.98±34.52	38.93±29.78	208.81±20.1	70.53±22.6
	PPOL-random	155.81±16.84	2.67±6.6	150.65±17.63	2.17±5.02	114.24±35.22	1.83±6.08	183.07±24.63	58.53±22.3
	SA-PPOL	143.34±32.08	0.13±0.56	142.66±35.01	4.53±10.67	159.02±43.95	37.47±26.5	203.85±27.56	51.47±27.79
	⋆ SA-PPOL(MC)	-0.62±1.72	0.0±0.0	0.09±1.27	0.0±0.0	-0.17±1.46	0.0±0.0	-0.34±1.61	0.0±0.0
	⋆ SA-PPOL(MR)	-0.8±2.28	0.0±0.0	-0.57±2.2	0.0±0.0	-0.89±2.12	0.0±0.0	-0.86±2.09	0.0±0.0
	ADV-PPOL(MC)	135.98±15.99	0.3±1.62	130.76±18.87	0.77±4.13	137.13±29.4	6.33±13.96	134.68±22.01	5.3±9.39
	ADV-PPOL(MR)	133.27±19.53	0.87±3.25	127.19±32.64	1.2±4.49	118.57±26.37	0.83±2.02	141.74±23.63	1.07±3.08

ablation studies with SA-PPOL(MC) and SA-PPOL(MR) suggest that the KL-regularized robust training technique, which is successful in standard robust RL setting, does not work well for safe RL even with the same adversarial attacks during training, and they may also fail to obtain a high-rewarding policy in some tasks (see discussions of the training failure in Appendix B.2). As a result, we can conclude that the proposed adversarial training methods with the MC and MR attackers are better than baselines regarding both training stability and testing robustness and safety.

Generalization to other safe RL methods. We also conduct the experiments for other types of base safe RL algorithms, including another on-policy method FOCOPS (Zhang et al., 2020c), one off-policy method SAC-Lagrangian Yang et al. (2021), and one policy-gradient-free off-policy method CVPO (Liu et al., 2022). Due to the page limit, we leave the results and detailed discussions in Appendix C.9. In summary, all the vanilla safe RL methods suffer the vulnerability issue – though they are safe in noise-free environments, they are not safe anymore under strong attacks, which validates the necessity of studying the observational robustness of safe RL agents. In addition, the adversarial training can help to improve the robustness and make the FOCOPS agent much safer under attacks. Therefore, the problem formulations, methods, results, and analysis can be generalized to different safe RL approaches, hopefully attracting more attention in the safe RL community to study the inherent connection between safety and robustness.

6 Conclusion

We study the observational robustness regarding constraint satisfaction for safe RL and show that the optimal policy of tempting safe RL problems could be vulnerable. We propose two effective attackers to induce unsafe behaviors. An interesting and surprising finding is that maximizing the reward attack is as effective as directly maximizing the cost while keeping stealthiness. We further propose an adversarial training method to increase the robustness and safety performance for safe RL, and extensive experiments show that the proposed method outperforms the robust training techniques for standard RL settings. Our results show the existence of a previously unrecognized problem in safe RL, and we hope this work encourages other researchers to study safety from the robustness perspective, as both safety and robustness are important ingredients for real-world deployment.

REFERENCES

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

Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *Ieee Access*, 6:14410–14430, 2018.

Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

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

Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*, 2017.

Yarden As, Ilnura Usmanova, Sebastian Curi, and Andreas Krause. Constrained policy optimization via bayesian world models. *arXiv preprint arXiv:2201.09802*, 2022.

Felix Berkenkamp, Matteo Turchetta, Angela P Schoellig, and Andreas Krause. Safe model-based reinforcement learning with stability guarantees. *arXiv preprint arXiv:1705.08551*, 2017.

Shalabh Bhatnagar and K Lakshmanan. An online actor–critic algorithm with function approximation for constrained markov decision processes. *Journal of Optimization Theory and Applications*, 153 (3):688–708, 2012.

Lukas Brunke, Melissa Greeff, Adam W Hall, Zhaocong Yuan, Siqi Zhou, Jacopo Panerati, and Angela P Schoellig. Safe learning in robotics: From learning-based control to safe reinforcement learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 5, 2021.

Baiming Chen, Zuxin Liu, Jiacheng Zhu, Mengdi Xu, Wenhao Ding, and Ding Zhao. Context-aware safe reinforcement learning for non-stationary environments. *arXiv preprint arXiv:2101.00531*, 2021.

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.

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.

Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerik, Todd Hester, Cosmin Paduraru, and Yuval Tassa. Safe exploration in continuous action spaces. *arXiv preprint arXiv:1801.08757*, 2018.

Michael Dennis, Natasha Jaques, Eugene Vinitsky, Alexandre Bayen, Stuart Russell, Andrew Critch, and Sergey Levine. Emergent complexity and zero-shot transfer via unsupervised environment design. *Advances in Neural Information Processing Systems*, 33:13049–13061, 2020.

Benjamin Eysenbach and Sergey Levine. Maximum entropy rl (provably) solves some robust rl problems. *arXiv preprint arXiv:2103.06257*, 2021.

Yannis Flet-Berliac and Debabrota Basu. Saac: Safe reinforcement learning as an adversarial game of actor-critics. *arXiv preprint arXiv:2204.09424*, 2022.

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

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

Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. On the effectiveness of interval bound propagation for training verifiably robust models. *arXiv preprint arXiv:1810.12715*, 2018.

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

Peide Huang, Mengdi Xu, Fei Fang, and Ding Zhao. Robust reinforcement learning as a stackelberg game via adaptively-regularized adversarial training. *arXiv preprint arXiv:2202.09514*, 2022.

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

Garud N Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30(2): 257–280, 2005.

Sham Machandranath Kakade. *On the sample complexity of reinforcement learning*. University of London, University College London (United Kingdom), 2003.

Jernej Kos and Dawn Song. Delving into adversarial attacks on deep policies. *arXiv preprint arXiv:1705.06452*, 2017.

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

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Shiau Hong Lim, Huan Xu, and Shie Mannor. Reinforcement learning in robust markov decision processes. *Advances in Neural Information Processing Systems*, 26, 2013.

Yen-Chen Lin, Zhang-Wei Hong, Yuan-Hong Liao, Meng-Li Shih, Ming-Yu Liu, and Min Sun. Tactics of adversarial attack on deep reinforcement learning agents. *arXiv preprint arXiv:1703.06748*, 2017.

Zichuan Lin, Garrett Thomas, Guangwen Yang, and Tengyu Ma. Model-based adversarial meta-reinforcement learning. *Advances in Neural Information Processing Systems*, 33:10161–10173, 2020.

Zuxin Liu, Zhepeng Cen, Vladislav Isenbaev, Wei Liu, Steven Wu, Bo Li, and Ding Zhao. Constrained variational policy optimization for safe reinforcement learning. In *International Conference on Machine Learning*, pp. 13644–13668. PMLR, 2022.

Yuping Luo and Tengyu Ma. Learning barrier certificates: Towards safe reinforcement learning with zero training-time violations. *Advances in Neural Information Processing Systems*, 34, 2021.

Gabriel Resende Machado, Eugênio Silva, and Ronaldo Ribeiro Goldschmidt. Adversarial machine learning in image classification: A survey toward the defender's perspective. *ACM Computing Surveys (CSUR)*, 55(1):1–38, 2021.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Amram Meir and Emmett Keeler. A theorem on contraction mappings. *Journal of Mathematical Analysis and Applications*, 28(2):326–329, 1969.

Janosch Moos, Kay Hansel, Hany Abdulsamad, Svenja Stark, Debora Clever, and Jan Peters. Robust reinforcement learning: A review of foundations and recent advances. *Machine Learning and Knowledge Extraction*, 4(1):276–315, 2022.

Rémi Munos, Tom Stepleton, Anna Harutyunyan, and Marc G Bellemare. Safe and efficient off-policy reinforcement learning. *arXiv preprint arXiv:1606.02647*, 2016.

Fabio Muratore, Felix Treede, Michael Gienger, and Jan Peters. Domain randomization for simulation-based policy optimization with transferability assessment. In *Conference on Robot Learning*, pp. 700–713. PMLR, 2018.

Santiago Paternain, Luiz FO Chamon, Miguel Calvo-Fullana, and Alejandro Ribeiro. Constrained reinforcement learning has zero duality gap. *arXiv preprint arXiv:1910.13393*, 2019.

Anay Pattanaik, Zhenyi Tang, Shuijing Liu, Gautham Bommannan, and Girish Chowdhary. Robust deep reinforcement learning with adversarial attacks. *arXiv preprint arXiv:1712.03632*, 2017.

Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In *International Conference on Machine Learning*, pp. 2817–2826. PMLR, 2017.

Nikolaos Pitropakis, Emmanouil Panaousis, Thanassis Giannetsos, Eleftherios Anastasiadis, and George Loukas. A taxonomy and survey of attacks against machine learning. *Computer Science Review*, 34:100199, 2019.

Rémy Portelas, Cédric Colas, Lilian Weng, Katja Hofmann, and Pierre-Yves Oudeyer. Automatic curriculum learning for deep rl: A short survey. *arXiv preprint arXiv:2003.04664*, 2020.

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

Matthew Riemer, Miao Liu, and Gerald Tesauro. Learning abstract options. *Advances in neural information processing systems*, 31, 2018.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

William Saunders, Girish Sastry, Andreas Stuhlmueller, and Owain Evans. Trial without error: Towards safe reinforcement learning via human intervention. *arXiv preprint arXiv:1707.05173*, 2017.

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

Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 acm sigsac conference on computer and communications security*, pp. 1528–1540, 2016.

Aivar Sootla, Alexander I Cowen-Rivers, Taher Jafferjee, Ziyan Wang, David H Mguni, Jun Wang, and Haitham Ammar. Sauté rl: Almost surely safe reinforcement learning using state augmentation. In *International Conference on Machine Learning*, pp. 20423–20443. PMLR, 2022.

Krishnan Srinivasan, Benjamin Eysenbach, Sehoon Ha, Jie Tan, and Chelsea Finn. Learning to be safe: Deep rl with a safety critic. *arXiv preprint arXiv:2010.14603*, 2020.

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

Richard S Sutton, Andrew G Barto, et al. Introduction to reinforcement learning. 1998.

Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211, 1999.

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

Chen Tessler, Yonathan Efroni, and Shie Mannor. Action robust reinforcement learning and applications in continuous control. In *International Conference on Machine Learning*, pp. 6215–6224. PMLR, 2019.

Brijen Thananjeyan, Ashwin Balakrishna, Suraj Nair, Michael Luo, Krishnan Srinivasan, Minho Hwang, Joseph E Gonzalez, Julian Ibarz, Chelsea Finn, and Ken Goldberg. Recovery rl: Safe reinforcement learning with learned recovery zones. *IEEE Robotics and Automation Letters*, 6(3): 4915–4922, 2021.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 23–30. IEEE, 2017.

Jingkang Wang, Yang Liu, and Bo Li. Reinforcement learning with perturbed rewards. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 6202–6209, 2020.

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

Qisong Yang, Thiago D Simão, Simon H Tindemans, and Matthijs TJ Spaan. Wcsac: Worst-case soft actor critic for safety-constrained reinforcement learning. In *AAAI*, pp. 10639–10646, 2021.

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

Ming Yu, Zhuoran Yang, Mladen Kolar, and Zhaoran Wang. Convergent policy optimization for safe reinforcement learning. *arXiv preprint arXiv:1910.12156*, 2019.

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*, 33:21024–21037, 2020a.

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. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21024–21037. Curran Associates, Inc., 2020b. URL https://proceedings.neurips.cc/paper/2020/file/f0eb6568ea114ba6e293f903c34d7488-Paper.pdf.

Shangtong Zhang and Shimon Whiteson. Dac: The double actor-critic architecture for learning options. *Advances in Neural Information Processing Systems*, 32, 2019.

Yiming Zhang, Quan Vuong, and Keith Ross. First order constrained optimization in policy space. *Advances in Neural Information Processing Systems*, 2020c.