Off-Policy Safe Reinforcement Learning with Cost-Constrained Optimistic Exploration

Anonymous authors
Paper under double-blind review

Abstract

When formulating safety as limits of cumulative cost, safe reinforcement learning (RL) learns policies that maximize rewards subject to these constraints during both data collection and deployment. While off-policy methods offer high sample efficiency, their application to safe RL faces substantial challenges from constraint violations caused by the cost-agnostic exploration and the underestimation bias in the cost value function. To address these challenges, we propose Constrained Optimistic eXploration Q-learning (COX-Q), an off-policy primal-dual safe RL method that integrates cost-bounded exploration and conservative distributional RL. First, we introduce a novel cost-constrained optimistic exploration strategy that resolves gradient conflicts between reward and cost in the action space, and adaptively adjusts the trust region to control constraint violation in exploration. Second, we adopt truncated quantile critics to mitigate the underestimation bias in costs. The quantile critics also quantify distributional, risk-sensitive epistemic uncertainty for guiding exploration. Experiments across velocity-constrained robot locomotion, safe navigation, and complex autonomous driving tasks demonstrate that COX-Q achieves high sample efficiency, competitive safety performance during evaluation, and controlled data collection cost in exploration. The results highlight the proposed method as a promising solution for safety-critical RL.

1 Introduction

Deploying reinforcement learning (RL) agents in many real-world tasks requires safety guarantees. For example, robots must not harm humans (Luo et al., 2025), and autonomous vehicles must avoid collisions (Feng et al., 2023). Such concerns motivate *safe RL*, which commonly formulates the problem as a constrained Markov decision process (CMDP) (Altman, 2021). In this setting, the agent aims to maximize return while keeping the cumulative safety cost below a threshold. Growing interest in RL deployment has driven increasing attention to safe RL (Brunke et al., 2022).

Collecting data directly from the environment is essential for many RL applications due to the limitations of simulation fidelity or the need for human-in-the-loop interactions. Domains such as autonomous driving in mixed traffic Chen et al. (2024) and healthcare (Gottesman et al., 2019) require agents to collect data safely in the real world. In this context, *sample efficiency* is critical for safe RL, as it directly determines the data collection cost.

Off-policy RL gains higher sample efficiency than on-policy methods by experience replay (Chen et al., 2021) and uncertainty-driven optimistic exploration (Ladosz et al., 2022). However, applying off-policy methods to safe RL faces significant challenges. First, the underestimation bias in cumulative cost often leads to constraint violations (Wu et al., 2024). In primal-dual safe RL (Stooke et al., 2020), the changing Lagrangian multiplier further destabilizes the safety performance. Second, the data collection in off-policy safe RL lacks cost constraints. Applying optimistic exploration can potentially lead agents into risky regions and result in uncontrolled data collection costs. As a result, existing safe RL methods are predominantly on-policy (Gu et al., 2024b). Off-policy approaches are found struggling to satisfy cost constraints in both data collection and deployment, as shown in the OmniSafe benchmark (Ji et al., 2024). These issues highlight a critical knowledge gap:

How can off-policy safe RL maintain high data efficiency and meanwhile achieve robust constraint satisfaction in both data collection and deployment, through cost-constrained exploration and reliable value learning?

To address this challenge, we propose *Constrained Optimistic eXploration Q-learning (COX-Q)*, an off-policy primal-dual safe RL algorithm that maintains data-efficient learning and achieves robust cost constraint satisfaction in data collection and deployment. COX-Q integrates a novel cost-bounded optimistic exploration strategy with conservative distributional value estimation and uncertainty quantification. Our method demonstrates competitive performance across diverse safe RL benchmarks, showcasing its effectiveness for safety-critical applications.

2 Related work

This section provides a concise overview of related work to contextualize the core contributions of this study. We first clarify some key terminologies and define the scope of the overview. Safe RL is a broad concept that involves a wide range of methodologies, such as Control Barrier Functions (CBFs) (Chen et al., 2024), reachability methods (Ganai et al., 2023). We focus on the formulation of safety as constraints on cumulative costs, and address it within the constrained RL framework (Altman, 2021). Additionally, this overview comprises only model-free safe RL methods. Model-based methods (e.g., Safe Dreamer (Huang et al., 2023)) are not included due to fundamental differences. Related methods are grouped into on-policy and off-policy categories.

Most existing safe RL methods are on-policy, as sharing the behaviour and target policies allows each update to directly enforce constraint satisfaction through adjusted gradients or trust region techniques. On-policy approaches include first-order methods such as FOCOPS (Zhang et al., 2020) and CUP (Yang et al., 2022), as well as second-order methods like CPO, PCPO (Achiam et al., 2017), and RCPO (Tessler et al., 2018). Other variants include the PID-Lagrangian method (Stooke et al., 2020), risk-aware scheduling methods such as Saute RL (Sootla et al., 2022a) and PPOSimmer (Sootla et al., 2022b), and the early terminated MDP formulation (Sun et al., 2021). These methods and their variants have demonstrated strong empirical performance in many safe RL benchmarks. For a comprehensive review, we refer readers to (Gu et al., 2024b).

In contrast, off-policy safe RL is less studied. Most approaches adopt primal-dual methods like Lagrangian and PID-Lagrangian (Stooke et al., 2020), but suffer from poor safety performance due to the underestimation bias in cost values, often leading to constraint violations. To mitigate this, conservative cost estimators have been proposed. For example, Worst-Case SAC (Yang et al., 2021) penalizes underestimated costs to improve constraint satisfaction. CAL (Wu et al., 2024) further accelerates training using local policy convexification and the augmented Lagrangian method, achieving strong safety and sample efficiency using a high update-to-data (UTD) ratio. In terms of exploration, Gao et al. (2025) proposed the so-called MICE to address the underestimation of cost. The key idea is to use a memory-based intrinsic cost around unsafe states so the cost critic conservatively overestimates risk. Although the original implementation is for on-policy methods, the idea can be potentially adopted to off-policy approaches. A recent study by McCarthy et al. (2025) incorporates optimistic actor-critic (OAC) (Ciosek et al., 2019) into off-policy safe RL. The resulting ORAC algorithm actively explores regions with potentially higher reward and lower cost. While ORAC shows robust safety performance in tests, as its appendix says, it does not enforce cost constraints in data collection. How to realize cost-compliant exploration remains an open challenge.

In summary, a key gap in off-policy safe RL is the lack of a principled cost-constrained exploration strategy integrated with conservative value learning. Our approach addresses this challenge from both theoretical and practical aspects.

3 Problem formulation

Consider a CMDP defined by $(S, A, r, c, p, p_0, \gamma, d)$. $S \subseteq \mathbb{R}^m$ is the state space. For a state $s_t \in S$, an agent controlled by a policy $a \sim \pi(\cdot|s)$ takes an action a_t in the action space $A \subseteq \mathbb{R}^n$, then the next state follows $p(s_{t+1}|s_t, a_t)$. The agent receives a reward $r_t \in \mathbb{R}$ and pays a non-negative cost $c_t \in \mathbb{R}^+$. The distribution of the initial state is $p_0(s_0)$. $\gamma \in (0, 1)$ is the discount factor shared by the cumulative reward Z_r^π and cost Z_c^π, which are both *random variables*:

$$Z_r^\pi(s_t, a_t) = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}, \quad Z_c^\pi(s_t, a_t) = \sum_{k=0}^{\infty} \gamma^k c_{t+k+1}. \tag{1}$$

The state-action value function (Q-function) is defined as the expectation of return for the policy:

$$Q_r^\pi(s_t, a_t) = \mathbb{E}_\pi[Z_r^\pi(s_t, a_t)], \quad Q_c^\pi(s_t, a_t) = \mathbb{E}_\pi[Z_c^\pi(s_t, a_t)]. \tag{2}$$

Safe RL considers a constrained optimization problem:

$$\max_\pi \mathbb{E}_{s\sim\rho_\pi, a\sim\pi(\cdot|s)}[Q_r^\pi(s,a)], \quad \text{s.t.} \quad \mathbb{E}_{s\sim\rho_\pi, a\sim\pi(\cdot|s)}[Q_c^\pi(s,a)] \le d, \tag{3}$$

where ρ_π is the state density function of π, and d is the cost threshold. The primal-dual approach constructs the following dual form, updating the policy π and Lagrangian multiplier λ iteratively:

$$\arg\min_{\lambda>0} \mathbb{E}_{s\sim\rho_\pi, a\sim\pi(\cdot|s)}[Q_r^\pi(s,a) - \lambda(Q_c^\pi(s,a) - d)], \tag{4}$$

$$\arg\min_{\lambda>0} \lambda \times (d - \mathbb{E}_{s\sim\rho_\pi, a\sim\pi(\cdot|s)}[Q_c^\pi(s,a)]). \tag{5}$$

It is useful to note that d is the cost limit for both data collection (training) and tests. This requirement is naturally satisfied for on-policy methods, but *not* for off-policy methods that use different data collection and target policies. Next, we introduce the proposed COX-Q algorithm in detail.

4 Cost-constrained optimistic exploration

This section introduces our core novelty, Cost-Constrained Optimistic eXploration (COX). COX focuses on addressing the first challenge: cost-constrained exploration during data collection, while preserving the off-policy training pipeline and its sample-efficiency properties. The theoretical results in this section are based on the assumption of Gaussian action distributions (Gaussian policies), which are compatible with most mainstream off-policy RL methods.

Off-policy RL can actively explore using Optimistic Actor Critic (OAC) (Ciosek et al., 2019) for continuous control tasks. In single-objective RL, OAC first estimates an optimistic upper bound of Q-value $\hat{Q}^{\text{UB}}(s,a)$ from an ensemble of critics, then maximizes this objective under a KL divergence constraint (trust region). If the action distribution of the target policy is $\mathcal{N}(\mu_T, \Sigma_T)$, the exploration policy $\mathcal{N}(\mu_E, \Sigma_E)$ for collecting data is given by the theorem in OAC (Ciosek et al., 2019):

$$\mu_E = \mu_T + \sqrt{2\delta} \times \frac{\Sigma_T[\nabla_a \hat{Q}^{\text{UB}}(s,a)]_{a=\mu_T}}{\left\| [\nabla_a \hat{Q}^{\text{UB}}(s,a)]_{a=\mu_T} \right\|_{\Sigma_T}}, \quad \Sigma_E = \Sigma_T, \tag{6}$$

where δ is the KL-divergence threshold. For safe RL, we now have:

- A cost limit d divides (s,a) into safe ($Q_c^\pi(s,a) \le d$) and unsafe ($Q_c^\pi(s,a) > d$) regions.
- Two objectives of cumulative reward $Q_r^\pi(s,a)$ and cost $Q_c^\pi(s,a)$ that impact exploration.

Ideally, we hope that the exploration policy fully explores the safe region and minimizes the visits to the unsafe region (constraint violations). To this end, we must determine *(1) what is the effective exploration direction?* and *(2) what is the exploration step length?*.

4.1 Policy-MGDA for exploration gradient conflict resolution

We first determine the effective exploration direction (gradient). Safe RL involves two objectives, making exploration a multi-task problem in nature. We denote (omitting superscript π):

$$g_r = \nabla_a \hat{Q}_r^{\text{UB}}(s,a)|_{a=\mu_T} \quad g_c = \nabla_a \hat{Q}_c^{\text{LB}}(s,a)|_{a=\mu_T}, \quad g_m = \nabla_a \hat{Q}_c^{\text{mean}}(s,a)|_{a=\mu_T}, \tag{7}$$

where superscripts "UB" and "LB" represent estimated optimistic upper and lower bounds, respectively. Note that the dual form in equation 4 favours higher reward and lower cost. We also rewrite the shift of the mean action Δ and the trust region in equation 6 as follows (g_t is the total gradient):

$$\Delta = \mu_E - \mu_T = \eta\Sigma_T g_t, \quad \eta = \sqrt{\frac{2\delta}{g_t^\intercal \Sigma_T g_t}}. \tag{8}$$

Within the safe area ($Q_c^{\pi}(s,a) \leq d$), the KKT condition of equation 3 indicates that we can directly explore along g_r. In the unsafe area, the gradient is computed using the overall objective in equation 4, giving $g_t = g_r - \lambda g_c$. However, this naive sum cannot be directly used. We further want to ensure that both reward and cost are improving:

$$\Delta \hat{Q}_c^{\text{LB}}(s,\mu_E) = g_c^{\intercal}\Delta = \eta \times g_c^{\intercal}\Sigma_T g_t \leq 0 \quad \text{and} \quad \Delta \hat{Q}_r^{\text{UB}}(s,\mu_E) = g_r^{\intercal}\Delta = \eta \times g_r^{\intercal}\Sigma_T g_t \geq 0. \quad (9)$$

If one of the conditions in equation 9 is violated, we say that *exploration gradients conflict*, indicating that either reward or cost is damaged in the exploration. If reward dominates the exploration in unsafe areas, then agents may not explore towards the safe area. If cost dominates, then the exploration of reward may be hindered. Note that η is non-negative. So, the conflict is defined in the action space, measured by Σ*-metric*:

$$\langle g_i, g_j \rangle_{\Sigma_T} = g_i^{\intercal}\Sigma_T g_j, \tag{10}$$

which is different from multi-task learning using the direct inner product (Zhang & Yang, 2021).

To resolve exploration gradient conflicts, we extend the Multiple Gradient Descent Algorithm (MGDA) (Désidéri, 2012) to the action space, forming the so-called *Policy-MGDA*. We first define a space of gradients in which both conditions of equation 9 are satisfied:

$$K := \{g : v_r = \langle g_r, g \rangle_{\Sigma_T} \geq 0, v_c = \langle -g_c, g \rangle_{\Sigma_T} \geq 0\}. \tag{11}$$

For two gradient vectors, such K always exists except for degenerated or colinear cases. Then we find the optimal u^* that best aligns with the original direction $g_t = g_r - \lambda g_c$:

$$u^* = \arg\min_{u \in K} \|u - g_t\|^2_{\Sigma_T}. \tag{12}$$

Lemma 1 *We denote the following Gram-scalars and multipliers:*

$$s_{ij} = \langle g_i, g_j \rangle_{\Sigma_T}, \quad v_i = \langle g_t, g_i \rangle_{\Sigma_T}, \quad \mu_r = \frac{-s_{cc}v_r + s_{rc}v_c}{s_{rr}s_{cc} - s_{rc}^2}, \quad \mu_c = \frac{-s_{rc}v_r + s_{rr}v_c}{s_{rr}s_{cc} - s_{rc}^2} \tag{13}$$

Then the optimal solution for equation 12 is:

$$u^* = \begin{cases} g_t & \textit{if } g_t \in K \\ g_t - \dfrac{v_r}{s_{rr}} g_r & \textit{if } v_r < 0 \textit{ and } v_c \leq 0 \\ g_t - \dfrac{v_c}{s_{cc}} g_c & \textit{if } v_r \geq 0 \textit{ and } v_c > 0 \\ g_t - \mu_r g_r + \mu_c g_c & \textit{if } v_r < 0 \textit{ and } v_c > 0 \end{cases} \tag{14}$$

The proof is in Appendix A.1. u^* is the aligned, effective exploration direction in unsafe regions. Note that policy-MGDA operates in the action space during the online data collection stage with frozen network parameters, which makes it fundamentally different in both role and design from existing gradient manipulation methods in safe RL (Gu et al., 2024a; Chow et al., 2021; Liu et al., 2022).

4.2 Adaptive step length for exploration cost control

After determining the exploration gradient, we adjust the step length adaptively to control the cost of explorative data collection. To this end, we consider both the microscopic single-step exploration and the macroscopic training progress.

For each exploration step, the original OAC does not involve the cost constraint in equation 3. To address this issue, we explicitly bound the cost expectation by adjusting the step length η. Given exploration direction g^* (u^* in unsafe area or g_r in safe area), the threshold of non-negative violation along this direction is the hinge:

$$\phi(\eta) = [\Delta \hat{Q}_c^{\text{mean}} - (d - \hat{Q}_c^{\text{mean}})]_+ = [\eta \langle g_m, g^* \rangle_{\Sigma_T} - (d - \hat{Q}_c^{\text{mean}})]_+. \tag{15}$$

Denote η_{KL} as the full step length, we can formulate the following bi-level optimization problem:

$$\arg\max_{\eta^*} \eta^* \quad \text{s.t.} \quad 0 \leq \eta^* \leq \eta_{\text{KL}}, \quad \phi(\eta^*) = \min_{0 \leq \xi \leq \eta_{\text{KL}}} \phi(\xi). \tag{16}$$

It means that, once the full exploration step length makes the mean cost exceed d, then we choose the maximum η^* in the trust region to ensure the cost constraint violation $\phi(\eta)$ is 0 or minimized.

Lemma 2 *We denote (g_m is define in equation 7):*

$$s = \langle g_m, g^* \rangle_{\Sigma_T}, \quad r = d - \hat{Q}_c^{mean} \tag{17}$$

Then the optimal solution for equation 16 is:

$$\eta^* = \begin{cases} \eta_{KL} & \textit{if } s < 0 \\ 0 & \textit{if } s > 0 \textit{ and } r < 0; \textit{or } s = 0 \\ \min(\eta_{KL}, r/s) & \textit{if } s > 0 \textit{ and } r \geq 0 \end{cases} \tag{18}$$

The proof is given in Appendix A.2. η^* thus minimizes the cost constraint violation (not minimizing the cost itself) for each exploration step.

Nevertheless, equation 18 is not always valid. When g^* tends to 0 around the optimum, $s \to 0$. So, the oscillating sign of g^* makes η^* jump between $\pm\eta_{\text{KL}}$, manifesting as a pure extra action noise. To address this issue, we further adaptively adjust δ (thus the maximum step length η_{KL}) based on the near-on-policy cost in a recent replay buffer $\mathcal{B}_{\text{recent}}$, similar to the Lagrangian multiplier:

$$\arg\min_{0<\delta\leq\bar{\delta}} \delta \times (d - \mathbb{E}_{c_i \in \mathcal{B}_{\text{recent}}} c_i). \tag{19}$$

As a result, the exploration cost is governed by d. The adaptive step length tends to fully utilize the budget in safe regions without violation, while remaining conservative in unsafe regions. By using the two lemmas above, we can get the adjusted exploration direction u^* and the step length η^*. Inserting them back into the OAC theorem in equation 6 gives the final cost-constraint compliant exploration policy.

So far, we have explained the "COX-" part, including the effective exploration direction and the adaptive step length under cost constraints. It is useful to note that the theories in this section are based on accurate value estimation, particularly for costs. If the critics cannot provide reliable cost estimates due to the lack of data or function approximation errors, especially in the early stage of training, the data-collection cost cannot be effectively controlled. Plausible improvements include incorporating classical methods such as reachability analysis (Ganai et al., 2023), or combining COX with model-based RL, such as SafetyDreamer (Huang et al., 2023).

Next, we introduce the "-Q" part about distributional value learning and the uncertainty quantification method for estimating optimistic bounds.

5 TQC-BASED VALUE LEARNING AND UNCERTAINTY QUANTIFICATION

This section introduces how to mitigate the underestimation bias in cost estimation by using TQC and conservative value learning. The objective function in equation 4 indicates that the Bellman update favours overestimation bias of reward and underestimation bias of cost (Wu et al., 2024). In this paper, we adopt Truncated Quantile Critics (TQC) (Kuznetsov et al., 2020) to mitigate the bias and promote exploration by distribution-level epistemic uncertainty.

TQC follows Quantile Regression RL (Dabney et al., 2018). Each critic learns the return distribution by a certain number of evenly distributed quantiles. The key difference is that TQC mixes and sorts quantiles from all critics, and then truncates the top k atoms to mitigate the overestimation bias. Specific to safe RL, we truncate the *top* k_r atoms for reward and the *bottom* k_c atoms for cost critics (note that the signs for reward and cost are different). The mixed quantiles provide low-variance gradients to stabilize the learning, and the number of truncated atoms controls biases with high flexibility.

Another advantage of TQC is that we can quantify distributional epistemic uncertainty. Assume we have N cost critics and N reward critics. Each critic predicts M quantiles. For instance, $q_{m,r}^{(n)}(s,a)$ is the approximated quantile function value at the corresponding level $\tau_m = (m - 0.5)/M$ for reward. Following a recent paper, ORAC (McCarthy et al., 2025), optimistic bounds are estimated by computing per-quantile bounds across the critic ensemble and aggregating them using Conditional Value at Risk (CVaR) (Rockafellar et al., 2000).

$$\hat{q}_{m,r}(s,a) = \hat{\mu}_{m,r}(s,a) + \beta_r \hat{\sigma}_{m,r}(s,a) \quad \hat{Q}_r^{\text{UB}}(s,a) = \frac{1}{M}\sum_{m=1}^{M} \hat{q}_{m,r}(s,a). \tag{20}$$

$$\hat{q}_{m,c}(s,a) = \hat{\mu}_{m,c}(s,a) - \beta_c \hat{\sigma}_{m,c}(s,a) \quad \hat{Q}_c^{\text{LB}}(s,a) = \frac{1}{\alpha}\sum_{m=1}^{\alpha} \hat{q}_{m,c}(s,a). \tag{21}$$

Here $\hat{\mu}_{m,r/c}$ and $\hat{\sigma}_{m,r/c}$ are mean and standard variance of the m-th quantile across N critics, respectively. For the cost lower bound, we use the α head quantiles only (CVaR is α/M). The two hyperparameters, β_r and β_c, adjust the level of optimism for both objectives.

Combining COX and TQC-based conservative learning yields the full COX-Q algorithm. It addresses both unconstrained exploration and underestimated cost in an integrated framework that maintains the inherent sample efficiency of off-policy RL. The implementation is based on the paper of CAL (Wu et al., 2024). We keep the augmented Lagrangian method in CAL to accelerate the training. The pseudo-code of COX-Q, the key differences from other baselines, and more details are provided in Appendix B.

6 Experiments

This section compares COX-Q to off-policy and on-policy baselines on three representative safe RL benchmarks: (1) *Velocity-constrained locomotion* is a dense-reward task with immediate cost signals. The robots move alone without interaction with other objects or agents. (2) *Safe navigation* poses a hard exploration challenge with sparse rewards and costs. The robots need to avoid touching static or fixed-route moving hazard objects. This is a typical open-loop control task. (3) *SMARTS autonomous driving* represents a strict safety task with a zero-cost threshold. Further, the vehicle needs to interact with other road users in a closed-loop manner, making it substantially challenging. In all cases, costs are binary: 0 for safe, and a fixed positive value for unsafe states.

6.1 Velocity-constrained robot locomotion

The first experiment is conducted on SafetyVelocity-v1, a velocity-constrained robot locomotion benchmark based on MuJoCo (Todorov et al., 2012). The objective is to maximize reward while keeping the velocity below a threshold; exceeding it incurs a cost of 1, otherwise 0. The episode cost limit (for 1000 steps) is set to 25. We evaluate four robot configurations, *hopper*, *walker2d*, *ant*, and *humanoid*, which share the same reward structure. For faster training, experiments are run in Brax (Freeman et al., 2021). Detailed environment settings are provided in Appendix C.1.

Baselines Selected baselines include representative on-policy and recent off-policy methods. For on-policy baselines, we select one from each of the categories introduced in Section 2. They are *CUP* (Yang et al., 2022), *RCPO* (Tessler et al., 2018), *PPOSaute* (Sootla et al., 2022a), *PPOSimmerPID* (Sootla et al., 2022b), and *CPPOPID* (Stooke et al., 2020).

For off-policy baselines, we choose (1) *SACUCB-PID* (Stooke et al., 2020), which augments the original SACPID by conservative cost learning; (2) *CAL* (Wu et al., 2024) uses a conservative estimate of cost and the augmented Lagrangian method (Luenberger et al., 1984). We choose UTD=1 for CAL so that the impact of high UTD ratios is excluded. Using UTD=1 for all baselines makes the role of conservative (or distributional) cost learning and the proposed exploration mechanism comparable across methods. Further, to clarify the contribution of different components in COX, we include three ablation baselines: (3) *TQC* uses TQC-based conservative value learning alone, without optimistic exploration. (4) *TQC+OAC* uses the direct gradient summation for exploration [1]. (5) *TQC+OAC (with step length auto-tuning)* further adds the adaptive length tuning in equation 19, but does not use gradient conflict resolution. Details about COX-Q and baselines are provided in Appendix D.

Results The results are presented in Figure 1. Overall, COX-Q demonstrates superior sample efficiency, achieves high cumulative returns, and has nearly-zero test costs after convergence, meanwhile keeping data collection costs below the predefined budget. More specifically: (1) COX-Q exhibits a clear advantage in data efficiency over on-policy baselines, particularly for high-dimensional

[1] A recent paper ORAC McCarthy et al. (2025) combines IQN (Dabney et al., 2018) with OAC, which is similar to the TQC+OAC baseline here. However, ORAC's code is not open yet, and only its hyperparameters for safe navigation are provided. Thus, we use the original IQN-based ORAC for safe navigation tasks only. For SafetyVelocity-v1 and SMARTS, we adopt a TQC-based variant instead.

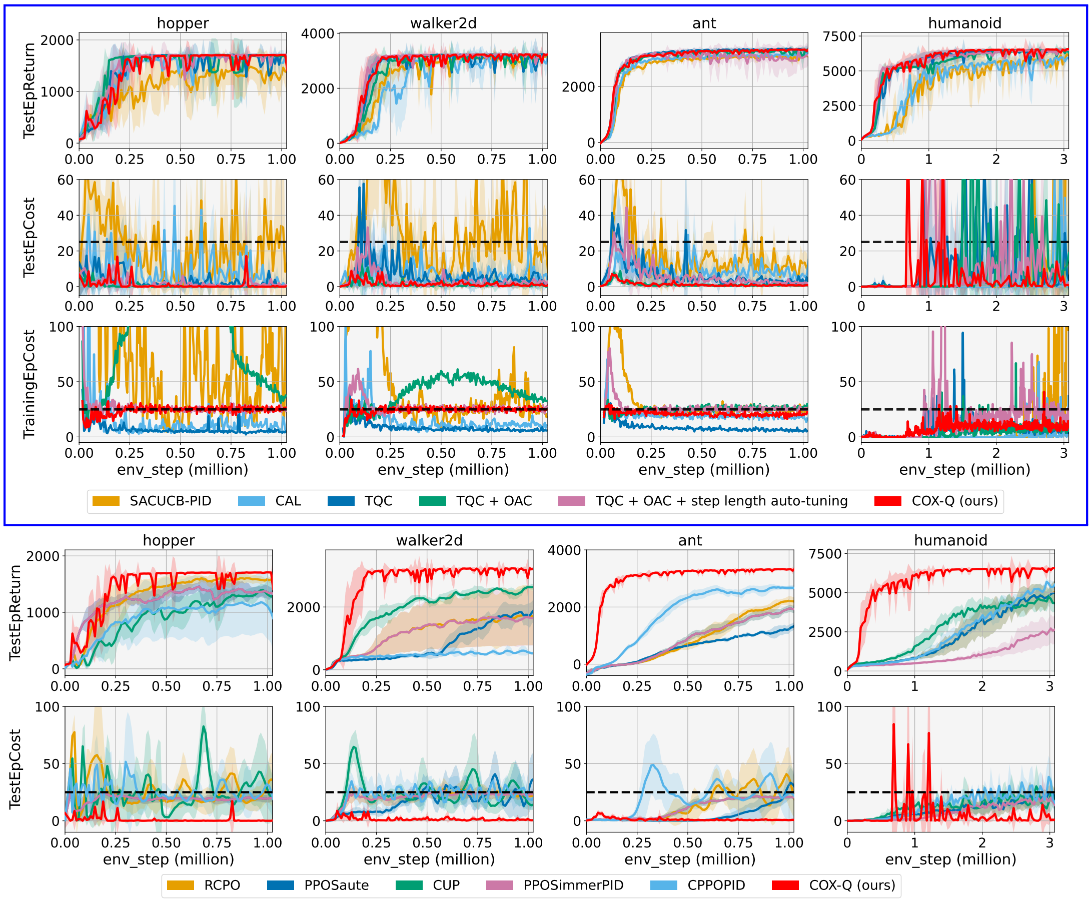

Figure 1: Benchmark of COX-Q against off-policy (top) and on-policy (bottom) baselines. TrainingEpCost is for data collection, which is expected to stay near or below the threshold throughout the training. For TestEpCost, the performance after convergence is important. Note that training and test costs are identical for on-policy methods.

action spaces such as *ant* and *humanoid*. Its ability to decouple the exploration and target policies enables seeking a deployment cost significantly lower than the constraint, a property that on-policy methods cannot realize. (2) By comparing TQC-based methods against *SACUCB-PID* and *CAL*, we observe that distributional RL has higher sample efficiency than point-value based baselines, particularly for bipedal robots. (3) Without optimistic exploration, *TQC* has lower data-collection costs but higher testing costs than other TQC-based baselines. This characterizes the exploration-exploitation trade-off with respect to costs. (4) The step length auto-tuning effectively regulates the data-collection cost, especially in the middle and late training phases. This is evidenced by the smooth and horizontal (near the threshold) training cost profiles of *TQC+OAC (with step length auto-tuning)* and COX-Q in the third row of Figure 1. In contrast, the naive combination of *TQC+OAC* suffers from elevated training costs in tasks with low-dimensional action spaces (*hopper*, *walker2d*) due to unregulated optimistic exploration. (5) The incorporation of gradient conflict resolution and step-wise cost-constrained exploration in COX-Q is critical for maintaining exploration cost constraint satisfaction in the early stage of training. Without these components, *TQC+OAC* and *TQC+OAC (with step length auto-tuning)* exhibit training cost constraint violations in the early learning phase, whereas COX-Q consistently adheres to this constraint across the entire training process.

The SafetyVelocity-v1 benchmark highlights the key strengths of COX-Q, including maintaining the high data efficiency of off-policy RL, improved deployment-time safety, and controlled training costs of explorative data collection. We next assess its performance in exploration-challenging and more complex environments.

6.2 Safe navigation

The second experiment evaluates COX-Q on safe navigation tasks from Safety-Gymnasium (Ji et al., 2023), which are characterized by sparse reward and cost signal. In these tasks, a mobile robot needs to reach a goal, such as navigating to a target location, pressing a specific button, or pushing an object, while avoiding static and dynamic hazards, including fragile obstacles. The observation space consists of Lidar-based point cloud data. We select four high-difficulty tasks: *SafetyPointButton2*, *SafetyPointGoal2*, *SafetyCarButton2*, and *SafetyPointPush1*, where the suffix "2" denotes the highest difficulty level. Detailed task descriptions are provided in Appendix C.2.

Different from SafetyVelocity-v1, due to the sparse rewards and costs in SafeNavigation, truncating too many atoms for cost-critics can suppress the learning of rewards. Therefore, we preserve the mixed quantiles in TQC but do not apply truncation. Instead, we use the estimated CVaR-based upper bound of cost to update the actor and Lagrangian multiplier, same as in Worst-Case SAC (Yang et al., 2021):

$$\hat{Q}_c^{\mathrm{UB}}(s,a) = \frac{1}{N(M-\alpha+1)} \sum_{n=1}^{N} \sum_{m=\alpha}^{M} \hat{q}_{m,c}^{(n)}(s,a). \tag{22}$$

Baselines The experiments are conducted using OmniSafe (Ji et al., 2024), a safe RL benchmark platform. The off-policy baselines include the provided SACPID (Stooke et al., 2020), CAL (UTD = 1) (Wu et al., 2024), and our implementation of the original IQN-based ORAC (McCarthy et al., 2025) (added with step length auto-tuning), which is a strong safe RL baseline in safe navigation. On-policy methods need significantly more interactions. Their performances are presented in Appendix E. All hyperparameter settings are provided in Appendix D.

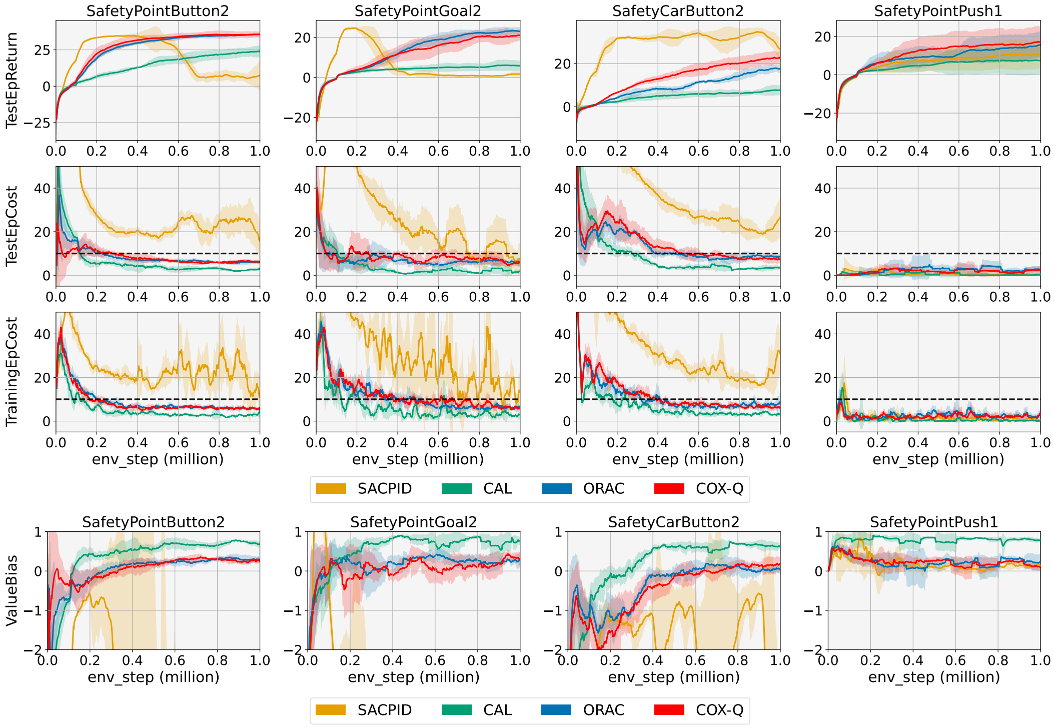

Figure 2: Benchmark of COX-Q against off-policy baselines on safe navigation tasks (episode cost limit is 10). The bottom figure is the cost value estimation bias, computed from cost critic outputs and the recorded trajectories in the evaluation phase. Below 0 means underestimation.

Results The performance and cost estimation biases are presented in Figure 2, with corresponding numerical results provided in Appendix E. In general, conservative distributional safe RL methods

(ORAC and COX-Q) have both higher returns and better cost constraint satisfaction than point-value based safe RL (CAL with UTD=1 and SACPID). However, although using different exploration strategies, COX-Q and ORAC exhibit close performance. This highlights two critical factors of the task. (1) Gradient conflict in the action space is weak in SafeNavigation, so the exploration gradients of ORAC and COX-Q become identical in most cases. In Appendix E, the analysis shows that the ratio of triggered gradient conflicts in the first 200K steps is below 10%, and even below 2% for SafetyPointPush1. Therefore, the difference in exploration policy is minor between ORAC and COX-Q. (2) The learning of the cumulative cost in SafeNavigation is highly biased due to the sparsity of cost signals. As shown at the bottom of Figure 2, the cost is underestimated in the first three tasks during the early training stage. Correspondingly, the violation of cost constraints during training and testing is observed in the same stage. The result indicates that, for constrained RL with sparse costs, the underestimation bias in the cumulative cost is the major bottleneck, rather than the exploration mechanism. For off-policy approaches, the cost learning can be made more robust by, e.g., using multi-step returns / TD learning, prioritized experience replay (Schaul et al., 2015), or Hindsight Experience Replay (HER) (Andrychowicz et al., 2017). These potential improvements need more investigation.

6.3 Safe autonomous driving in SMARTS

The objects in the previous safe navigation tasks follow certain motion patterns or stay static. In the third experiment, we evaluate COX-Q in challenging autonomous driving tasks in which surrounding vehicles have closed-loop interactions with our RL agent.

The experiments are conducted on the SMARTS autonomous driving simulation platform (Zhou et al., 2020). We select three scenarios with intensive vehicle interactions. (1) *Overtaking* on a two-lane highway. (2) *Intersection* without traffic lights. (3) *T-junction* without traffic lights. In the last two scenarios, the vehicle needs to execute an unprotected left turn and a lane change sequentially. Both the policy and critic networks employ a large WayFormer-like structure (Nayakanti et al., 2023). The reward includes a small distance progress towards the goal and a big bonus if the vehicle reaches the goal. The cost is -10 if the vehicle collides, drives off-road, or violates traffic rules severely (drives into the opposite direction). Also, if a collision or off-road happens, the episode is terminated immediately. If the vehicle fails to reach the goal in one minute, the episode terminates (marked as a timeout). More details about this task are provided in Appendix C.3, including a discussion about our reward and cost design that might be useful for some interested readers.

Autonomous driving is a typical strict safety task. We set a nearly-zero cost limit (0.01) like in SafetyDreamer's MetaDrive task (Huang et al., 2023). The vehicle stays in "unsafe" regions (the cumulative cost is above 0.01) during data collection and aims to minimize the test cost as much as possible. Unlike safe navigation, this setting in SMARTS intentionally increases the frequency of exploration gradient conflict and the proportion of constrained exploration. Larger networks also accelerate return approximation. We do not add the step length auto-tuning in equation 19 to avoid it converging to zero.

Baselines Due to the long training time, we selected 4 baselines and conducted one experiment using a fixed random seed only. *CPPOPID* was selected as the only on-policy baseline. For off-policy baselines, we selected *SACLag*, *CAL*, and TQC-based *ORAC* (which is essentially TQC+OAC as explained in Sec. 6.1). After 512K steps of training, we run 2000 episodes with stochastic initial states to obtain the test performance.

Results The test performance is presented in Table 1, and the number of unsafe events (collisions and off-road) during data collection is listed in Table 2. Overall, COX-Q achieves the best safety performance in testing without incurring excessive exploration cost or exhibiting over-conservative driving behaviours. Moreover, compared to ORAC, COX-Q significantly reduces both unsafe events during data collection and timeouts during testing. This shows that resolving conflicting gradients in a direction that simultaneously reduces cost and improves reward can effectively maximize return while keeping exploration cost under control.

Another notable point is that the safety performance of all methods in the overtaking is relatively worse than in the other scenarios. We found the reason is that SMARTS uses an instantaneous lane change model from SUMO (Krajzewicz et al., 2012), making collision avoidance inherently hard due to the lack of warning (e.g., turn signals).

Table 1: Test safety performance on SMARTS (512K steps, 2000 stochastic runs)

Scenario	Metric	CPPOPID	SACLag	CAL	ORAC	**COX-Q (ours)**
Overtaking	Collision	331	194	186	97	99
	Off-road	96	2	7	3	4
	Rule violation	3	0	0	0	0
	Timeout	0	2	1	887	0
Intersection	Collision	183	33	23	18	12
	Off-road	22	2	1	1	2
	Rule violation	9	18	0	0	0
	Timeout	0	0	1	12	0
T-junction	Collision	195	55	36	28	21
	Off-road	91	2	0	5	0
	Rule violation	3	24	0	0	0
	Timeout	0	0	17	86	5

Table 2: Number of unsafe events in data collection (512K steps, excluding the initial 5120 steps)

Scenario	CPPOPID	SACLag	CAL	ORAC	**COX-Q (ours)**
Overtaking	3697	1570	1544	3215	1665
Intersection	4969	1755	739	3589	1123
T-junction	5513	1965	1675	3837	1794

7 Conclusions

This paper proposes an off-policy primal-dual safe RL method, constrained optimistic exploration Q-learning, involving a novel cost-constrained optimistic exploration strategy and TQC-based conservative value learning. The proposed COX-Q is evaluated in three representative safe RL benchmarks. The results demonstrate that COX-Q has significantly higher data efficiency than on-policy baselines in all experiments. When the exploration gradient conflict between reward and cost is significant, and the critic networks are large enough to approximate the cost return (in SafeVelocity-v1 and SMARTS), COX-Q shows superior safe performance in tests, meanwhile effectively controlling exploration cost in data collection. When the exploration gradient conflict is weak or the bias in cost estimation is high due to sparse cost signal (in safe navigation), COX-Q is on par with the state-of-the-art method. In addition, the autonomous driving experiment showcases that the proposed method can be used in complex environments with large neural networks. In conclusion, COX-Q is a promising solution to RL applications with data efficiency and safety concerns.

Limitations The major limitation of this study is the reliability of quantified epistemic uncertainty. TQC mixes quantiles from all critics and learns the entire return distribution. Therefore, the diversity of critics for nearly Out-Of-Distribution samples might be suppressed due to highly correlated gradients for all critics. Implementing improved methods such as diverse ensemble projection (Zanger et al., 2023) or random priors (Osband et al., 2018) to enhance the quality of epistemic uncertainty quantification is a potential future research direction. Another future research direction is how to effectively implement COX in sparse-cost tasks such as SafeNavigation. A key step is to use, e.g., HER (Andrychowicz et al., 2017) or prioritized experience replay (Schaul et al., 2015) to robustify the cost-critic learning.

References

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

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

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

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(1):411–444, 2022.

Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Xinyue Chen, Che Wang, Zijian Zhou, and Keith Ross. Randomized ensembled double q-learning: Learning fast without a model. *arXiv preprint arXiv:2101.05982*, 2021.

Yinlam Chow, Ofir Nachum, Aleksandra Faust, Edgar Dueñez-Guzman, and Mohammad Ghavamzadeh. Safe policy learning for continuous control. In *Conference on Robot Learning*, pp. 801–821. PMLR, 2021.

Kamil Ciosek, Quan Vuong, Robert Loftin, and Katja Hofmann. Better exploration with optimistic actor critic. *Advances in Neural Information Processing Systems*, 32, 2019.

Will Dabney, Mark Rowland, Marc Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Jean-Antoine Désidéri. Multiple-gradient descent algorithm (mgda) for multiobjective optimization. *Comptes Rendus Mathematique*, 350(5-6):313–318, 2012.

Shuo Feng, Haowei Sun, Xintao Yan, Haojie Zhu, Zhengxia Zou, Shengyin Shen, and Henry X Liu. Dense reinforcement learning for safety validation of autonomous vehicles. *Nature*, 615(7953): 620–627, 2023.

C Daniel Freeman, Erik Frey, Anton Raichuk, Sertan Girgin, Igor Mordatch, and Olivier Bachem. Brax–a differentiable physics engine for large scale rigid body simulation. *arXiv preprint arXiv:2106.13281*, 2021.

Milan Ganai, Zheng Gong, Chenning Yu, Sylvia Herbert, and Sicun Gao. Iterative reachability estimation for safe reinforcement learning. *Advances in Neural Information Processing Systems*, 36:69764–69797, 2023.

Shiqing Gao, Jiaxin Ding, Luoyi Fu, and Xinbing Wang. Controlling underestimation bias in constrained reinforcement learning for safe exploration. In *Proceedings of the International Conference on Machine Learning*, 2025.

Thomas Gillespie. *Fundamentals of vehicle dynamics*. SAE international, 2021.

Omer Gottesman, Fredrik Johansson, Matthieu Komorowski, Aldo Faisal, David Sontag, Finale Doshi-Velez, and Leo Anthony Celi. Guidelines for reinforcement learning in healthcare. *Nature medicine*, 25(1):16–18, 2019.

Shangding Gu, Bilgehan Sel, Yuhao Ding, Lu Wang, Qingwei Lin, Ming Jin, and Alois Knoll. Balance reward and safety optimization for safe reinforcement learning: A perspective of gradient manipulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 21099–21106, 2024a.

Shangding Gu, Long Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, and Alois Knoll. A review of safe reinforcement learning: Methods, theories and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024b.

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

Weidong Huang, Jiaming Ji, Chunhe Xia, Borong Zhang, and Yaodong Yang. Safedreamer: Safe reinforcement learning with world models. *arXiv preprint arXiv:2307.07176*, 2023.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Josef Dai, and Yaodong Yang. Safety gymnasium: A unified safe reinforcement learning benchmark. *Advances in Neural Information Processing Systems*, 36:18964–18993, 2023.

Jiaming Ji, Jiayi Zhou, Borong Zhang, Juntao Dai, Xuehai Pan, Ruiyang Sun, Weidong Huang, Yiran Geng, Mickel Liu, and Yaodong Yang. Omnisafe: An infrastructure for accelerating safe reinforcement learning research. *Journal of Machine Learning Research*, 25(285):1–6, 2024.

Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, Laura Bieker, et al. Recent development and applications of sumo-simulation of urban mobility. *International journal on advances in systems and measurements*, 5(3&4):128–138, 2012.

Arsenii Kuznetsov, Pavel Shvechikov, Alexander Grishin, and Dmitry Vetrov. Controlling overestimation bias with truncated mixture of continuous distributional quantile critics. In *International conference on machine learning*, pp. 5556–5566. PMLR, 2020.

Pawel Ladosz, Lilian Weng, Minwoo Kim, and Hyondong Oh. Exploration in deep reinforcement learning: A survey. *Information Fusion*, 85:1–22, 2022.

Quanyi Li, Zhenghao Peng, Lan Feng, Qihang Zhang, Zhenghai Xue, and Bolei Zhou. Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning. *IEEE transactions on pattern analysis and machine intelligence*, 45(3):3461–3475, 2022.

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.

David G Luenberger, Yinyu Ye, et al. *Linear and nonlinear programming*, volume 2. Springer, 1984.

Jianlan Luo, Charles Xu, Jeffrey Wu, and Sergey Levine. Precise and dexterous robotic manipulation via human-in-the-loop reinforcement learning. *Science Robotics*, 10(105):eads5033, 2025.

James McCarthy, Radu Marinescu, Elizabeth Daly, and Ivana Dusparic. Optimistic exploration for risk-averse constrained reinforcement learning. *arXiv preprint arXiv:2507.08793*, 2025.

Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2980–2987. IEEE, 2023.

Ian Osband, John Aslanides, and Albin Cassirer. Randomized prior functions for deep reinforcement learning. *Advances in neural information processing systems*, 31, 2018.

R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

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, 2022a.

Aivar Sootla, Alexander I Cowen-Rivers, Jun Wang, and Haitham Bou Ammar. Effects of safety state augmentation on safe exploration. *arXiv preprint arXiv:2206.02675*, 2022b.

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.

Hao Sun, Ziping Xu, Meng Fang, Zhenghao Peng, Jiadong Guo, Bo Dai, and Bolei Zhou. Safe exploration by solving early terminated mdp. *arXiv preprint arXiv:2107.04200*, 2021.

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

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*, pp. 5026–5033. IEEE, 2012.

Chen Wang, Yuanchang Xie, Helai Huang, and Pan Liu. A review of surrogate safety measures and their applications in connected and automated vehicles safety modeling. *Accident Analysis & Prevention*, 157:106157, 2021.

Zifan Wu, Bo Tang, Qian Lin, Chao Yu, Shangqin Mao, Qianlong Xie, Xingxing Wang, and Dong Wang. Off-policy primal-dual safe reinforcement learning. In *ICLR*, 2024.

Long Yang, Jiaming Ji, Juntao Dai, Linrui Zhang, Binbin Zhou, Pengfei Li, Yaodong Yang, and Gang Pan. Constrained update projection approach to safe policy optimization. *Advances in Neural Information Processing Systems*, 35:9111–9124, 2022.

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 AAAI Conference on Artificial Intelligence*, volume 35, pp. 10639–10646, 2021.

Moritz A Zanger, Wendelin Böhmer, and Matthijs TJ Spaan. Diverse projection ensembles for distributional reinforcement learning. *arXiv preprint arXiv:2306.07124*, 2023.

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

Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE transactions on knowledge and data engineering*, 34(12):5586–5609, 2021.

Ming Zhou, Jun Luo, Julian Villella, Yaodong Yang, David Rusu, Jiayu Miao, Weinan Zhang, Montgomery Alban, Iman Fadakar, Zheng Chen, et al. Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving. *arXiv preprint arXiv:2010.09776*, 2020.

A Proofs of the two lemmas

A.1 Lemma-1

The solution of equation 11 and equation 12 is given as follows. For simplicity, we denote $g_1 = g_r$, $g_2 = -g_c$, and $\Sigma = \Sigma_T$ here.

Let $S = \text{span}\{g_1, g_2\}$. Decompose $g_t = g_S + g_\perp$ with $g_S \in S$ and $\langle g_\perp, g_1\rangle_\Sigma = \langle g_\perp, g_2\rangle_\Sigma = 0$. Constraints depend only on g_S. Therefore, it suffices to solve in S and then add back $g_\perp$. This becomes a 2D problem. We next derive the KKT conditions. With the inequalities $c_1(u) = -\langle g_1, u\rangle_\Sigma \leq 0$ and $c_2(u) = -\langle g_2, u\rangle_\Sigma \leq 0$, we add two *non-negative* multipliers to form the Lagrangian:

$$\mathcal{L}(u, \mu_1, \mu_2) = \frac{1}{2}\|u - g_t\|_\Sigma^2 + \mu_1(-\langle g_1, u\rangle_\Sigma) + \mu_2(-\langle g_2, u\rangle_\Sigma). \tag{A.1}$$

Then the stationarity is:

$$\nabla_u \mathcal{L} = \Sigma(u - g_t) - \mu_1 \Sigma g_1 - \mu_2 \Sigma g_2 = 0 \quad \Rightarrow \quad u = g_t + \mu_1 g_1 + \mu_2 g_2 \tag{A.2}$$

The primal feasibility gives:

$$\langle g_1, u\rangle_\Sigma \geq 0, \quad \langle g_2, u\rangle_\Sigma \geq 0. \tag{A.3}$$

The complementary slackness gives

$$\mu_1 \langle g_1, u\rangle_\Sigma = 0, \quad \mu_2 \langle g_2, u\rangle_\Sigma = 0. \tag{A.4}$$

So, we define the so-called Σ-Gram scalars and target correlations as:

$$s_{ij} = \langle g_i, g_j\rangle_\Sigma, \quad v_i = \langle g_i, g_t\rangle_\Sigma, \tag{A.5}$$

and plug stationarity into the constraints:

$$\begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} = -\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{A.6}$$

Because the Gram matrix is apparently SPD if g_1 and g_2 are not co-linear, the solution is unique whenever both constraints are active. For the degenerated co-linear cases, we assume that $g_1 = \alpha g_2$. If $\alpha > 0$, then K is a half-space, then the solution is a direct projection:

$$g^* = u = g_t - \min(0, \frac{v_1}{s_{11}}) g_1. \tag{A.7}$$

If $\alpha < 0$, constraints reduce to $\langle g_1, u\rangle_\Sigma = 0$ (hyper-plane):

$$g^* = u = g_t - \frac{v_1}{s_{11}} g_1, \tag{A.8}$$

and $\alpha = 0$ is trivial.

For non-degenerate cases, we apply the optimal active set $A = \{c_1(u), c_2(u)\}$. There are four possibilities:

(1) No constraint active: Then $\mu_1 = \mu_2 = 0$, $g^* = g_t$.

(2) Only $c_1(u)$ active: Set $\mu_2 = 0$. From $\langle g_1, u\rangle_\Sigma = 0$, we get:

$$\mu_1 = -\frac{v_1}{s_{11}} \quad \Rightarrow \quad u^* = g_t - \frac{v_1}{s_{11}} g_1. \tag{A.9}$$

(3) Only $c_2(u)$ active: Similar to the previous case, we have:

$$u^* = g_t - \frac{v_2}{s_{22}} g_2. \tag{A.10}$$

(4) Both boundaries active: Then we solve equation A.6. That gives:

$$\mu_1 = \frac{-s_{22}v_1 + s_{12}v_2}{\det G}, \quad \mu_2 = \frac{s_{12}v_1 - s_{11}v_2}{\det G}, \quad \det G = s_{11}s_{22} - s_{12}^2 \geq 0 \tag{A.11}$$

$$u^* = g_t - \mu_1 g_1 - \mu_2 g_2. \tag{A.12}$$

Replace g_1 and g_2 by g_r and $-g_c$, respectively, then the proof of Lemma 1 is done.

A.2 Lemma-2

The solution of equation 16 is derived based on two cases;

Case A: $s < 0$, which means moving along the u^* does not increase the expected cost. The hinge $\phi(\eta)$ is non-increasing w.r.t. η. Therefore, minimal violation is achieved by taking the largest trust region:

$$\eta^* = \eta_{\text{KL}} \tag{A.13}$$

Case B: $s > 0$, which means moving along the u^* increase the expected cost. Then we check the feasibility of a zero-violation set on the ray $\{\eta : \ \eta s \leq r\}$. If $r \leq 0$, then the zero-violation set is empty on $[0, \eta_{\text{KL}}]$ and the hinge increases with η. Therefore, the minimizer is trivial $\eta^* = 0$. If $r \geq 0$, simply take the boundary as the zero-violation set:

$$\eta^* = \min(\eta_{\text{KL}}, \frac{r}{s}) \tag{A.14}$$

Case C: $s = 0$, which means the hinge becomes a constant. In this case. If $r \geq 0$, every $\eta \in [0, \eta_{\text{KL}}]$ is optimal. If $r < 0$, violation is unavoidable. We set $\eta^* = 0$ by rule for conservativeness.

Combining the three cases above gives the complete proof of Lemma 2.

B Implementation details of COX-Q

Algorithm 1 COX-Q based on SAC, with optional Augmented Lagrangian Method(ALM)

```
Input and initialization: policy network $\pi_\theta(s)$, $N$ reward quantile critic networks $\{q_{\psi_i,r}\}_{i=1}^N$,
$N$ cost quantile critic networks $\{q_{\psi_i,c}\}_{i=1}^N$, both with default 25 quantile heads.
replay buffer $\mathcal{D}$, truncation parameters $k_r$ and $k_c$, exploration optimism parameters $\beta_r$ and $\beta_c$,
cost limit $d$, maximum trust region size $\eta_{\text{KL}}$, Lagrangian multiplier $\lambda$,
risk-level CVaR $\alpha$
repeat
    Observe State $s_t$,
    if use COX then
        Compute the target policy $\mathcal{N}(\mu_T, \Sigma_T) = \pi(s_t)$
        Compute $\hat{Q}_r^{\text{UB}}$, $\hat{Q}_c^{\text{LB}}$, $\hat{Q}_c^{\text{mean}}$ from critics using equation 20 and equation 21
        Compute their gradients $g_r, g_c, g_m$ w.r.t $\mu_T$
        if $\hat{Q}_c^{\text{mean}}$ in safe area then
            compute $g^* = g_t = g_r - \lambda g_c$
        else
            Compute aligned exploration gradient $g^*$ using equation 14
        end if
        Compute adjusted step length $\eta^*$ using equation 18.
        Compute action shift $\Delta$ using OAC formula from $\eta^*$ and $g^*$
        select action $a_t = \text{clip}(\mu_e + \epsilon, a_{\text{lower}}, a_{\text{upper}})$, where $\epsilon \sim \mathcal{N}(\Delta, \Sigma_t)$
    else
        select action $a_t = \text{clip}(\mu_\theta(s_t) + \epsilon, a_{\text{lower}}, a_{\text{upper}})$, where $\epsilon \sim \mathcal{N}(0, \Sigma_t)$
    end if
    Execute $a_t$, observe next state $s_{t+1}$, reward $r_t$ and cost $c_t$
    Store the transition $(s_t, a_t, (r_t, c_t), s_{t+1})$ in $\mathcal{D}$
    if critic/actor update then
        Execute TQC or Worst-Case SAC updates, with optional ALM (used by default)
    end if
    if $\eta_{\text{KL}}$ update then
        Sample a recent $N_r$ transitions from $\mathcal{D}$, compute the average cost
        Update $\eta_{\text{KL}}$ using equation 19.
    end if
until Convergence
```

In the pseudo-code of Algorithm 1, the updates of critics are the same as the original TQC (Kuznetsov et al., 2020) or WCSAC (Yang et al., 2021). The actor update involves the ALM proposed by Luenberger et al. (1984) and introduced in safe RL by Wu et al. (2024). ALM alters the optimization objective of the actor by the following equations:

$$\begin{cases} \max_\pi \mathbb{E}_{s\sim\rho_\pi, a\sim\pi(\cdot|s)}[\hat{Q}_r^{\text{mean}} - \lambda(\hat{Q}_c^{\text{UB}} - d) - \frac{c}{2}(\hat{Q}_c^{\text{UB}} - d)^2], & \text{if } \frac{\lambda}{c} \geq d - \mathbb{E}(\mathbb{Q}^{\text{UCB}}) \\ \max_\pi \mathbb{E}_{s\sim\rho_\pi, a\sim\pi(\cdot|s)}(\hat{Q}_r^{\text{mean}}), & \text{otherwise} \end{cases} \tag{B.1}$$

The added quadratic term helps conform to cost constraints and move the optimization direction towards the cost limit, which can accelerate the learning process. In our studies, we use $c = 10$ for all tasks. This ALM is used for CAL, ORAC, and COX-Q in all experiments

In addition, off-policy safe RL needs to set the cap on Q-values d in an "on-policy" approach, instead of directly using the test episode costs as in on-policy methods. This is explained in the paper of CVPO (Liu et al., 2022), using the following formula:

$$d = d_{episode}\frac{1-\gamma^T}{T(1-\gamma)}, \tag{B.2}$$

in which T is the episode length. In all off-policy methods used in this study, we use this formula to convert the episode cost limit to the limit on Q_c^π.

C Description of the three safe RL environments

C.1 SafetyVelocity-v1

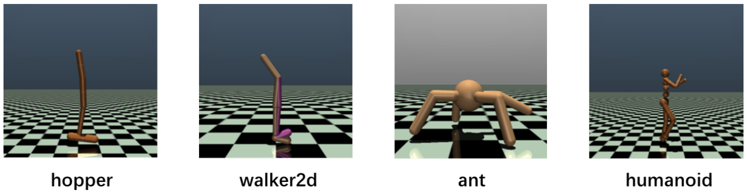

Figure C.1: The four selected robots in SafetyVelocity-v1 benchmark.

For the selected 4 robots, their configurations are shown in Figure C.1. They share the same reward structure as follows:

$$r_t = w_h \times r_{\text{health}} + w_v \times r_{\text{velocity}} - w_c \times r_{\text{ctrl}}, \tag{C.1}$$

in which r_{health} is a binary reward. If the robot keeps upright, get +1 reward; otherwise, get 0 and terminate the episode. r_{velocity} is a reward equal to the moving velocity along a given direction. r_{ctrl} is the control cost penalty, measuring how much torques are applied to the joints. w_h, w_v and w_c are three positive weights. Cost is binary. For hopper and walker2d, if the velocity along the +x axis exceeds the threshold, the cost is +1; otherwise, 0. For ant and humanoid, if the velocity along any direction exceeds the threshold, the cost is +1; otherwise, 0. The episodic cost limit is set to 25, as recommended in the original paper (Zhang et al., 2020). The weight coefficients, velocity thresholds, and the dimensionality of action spaces for different robots are listed in Table C.1. All implementations are based on the Brax (Freeman et al., 2021), using the same parameters (e.g. velocity thresholds) as in Safety-Gymnasium (Ji et al., 2023). Brax supports fully parallelized simulations on GPU, so it can save a lot of time for training. The default "generalized" backend is used for simulation.

C.2 Safe navigation in Safety-Gymnasium

We select four tasks in the safe navigation benchmark: SafetyPointButton2, SafetyPointGoal2, SafetyCarButton2, and SafetyPointPush1. The name is composed of two parts. "-Point-" or "-Car-" in the middle indicates what is the type of robot used, as shown on the top of Figure C.2. *Point* is a simple robot that has two actuators, one for rotation and the other for forward/backward movement. *Car*

Table C.1: Weight coefficients and velocity threshold for SafetyVelocity-v1

ROBOT	(w_h, w_v, w_c)	velocity threshold	Action dimension
hopper	(1, 1, 0.001)	0.7402	3
walker2d	(1, 1, 0.001)	2.3415	6
ant	(1, 1, 0.5)	2.6222	8
humanoid	(5, 1.25, 0.1)	1.4119	17

is a more complex robot that can move in three dimensions. It is equipped with two independently driven parallel wheels and a freely rotating rear wheel. Both steering and forward/backward motion require coordinated control of the two drive wheels, imposing more complex control dynamics. Both robots are equipped with 2D Lidars to perceive the environment. Their action dimensionalities are both 2.

The last part of the name indicates the type of task and its difficulty level. The three tasks used are shown at the bottom of Figure C.2.

- *Goal2:* The robot needs to reach a goal position (green pillar) while avoiding touching hazard pitfalls (blue circles) or move fragile vases (while cubes).
- *Button2:* The robot needs to reach the correct button (orange spheres) among 4 buttons, while avoiding touching blue-circle pitfalls or being hit by the moving gremlins (purple cubes moving in a circle).
- *Push1:* The robot needs to push the yellow object to the green goal position while avoiding blue pitfalls and the tall pillar.

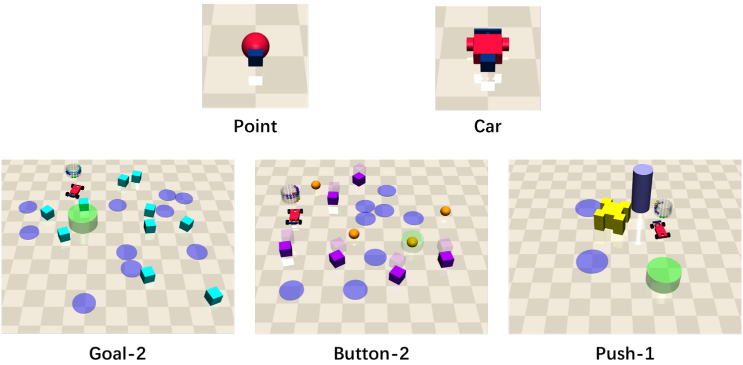

Figure C.2: The robots and the tasks in the safe navigation benchmark.

The reward and cost designs are complicated, depending on each specific task. We refer the readers to the public webpage of the Safety-Gymnasium for more details: https://safety-gymnasium.readthedocs.io/en/latest/environments/safe_navigation.html.

Additionally, to accelerate the learning process, the simulation time step is modified to 2.5 times the original value, according to the paper of CVPO (Liu et al., 2022) and CAL (Wu et al., 2024). While ORAC (McCarthy et al., 2025) does not release its code, the final reward performance implies that they probably used the same simulation settings. We therefore also keep the modification.

C.3 SMARTS AUTONOMOUS DRIVING

SMARTS is a scalable RL training platform for autonomous driving (Zhou et al., 2020), providing closed-loop simulation in diverse traffic scenarios. In this paper, we control an ego vehicle (red) to drive through the scenario. The ego vehicle has two actions: accelerations (between $\pm 6.5\,\mathrm{m\,s^{-2}}$) and steering rate (between $\pm 1.5\,\mathrm{rad\,s^{-1}}$ for intersections and $\pm 0.7\,\mathrm{rad\,s^{-1}}$ for highways). Then the vehicle's motion is controlled by a bicycle model (Gillespie, 2021). In the simulation, the vehicle can only change its actions every $0.25\,\mathrm{s}$ to avoid oscillating trajectories. Note that our settings are more realistic than the original SMARTS. In their default action spaces, the ego vehicle has infinite acceleration and can completely stop from the highest speed in $0.1\,\mathrm{s}$.

The three scenarios are illustrated in Figure C.3. For the intersection and the T-junction, the ego vehicle needs to first pass an unsignalized area and execute an unprotected left turn, then change to the right lane to reach the goal. For highway over-taking, the leading vehicle is slow, and other vehicles can change their lanes arbitrarily. The ego vehicle needs to overtake the slow vehicle and reach the goal on the same lane. All surrounding traffic vehicles are controlled by a set of predefined driving models with a distribution of inner parameters, providing diverse interactions.

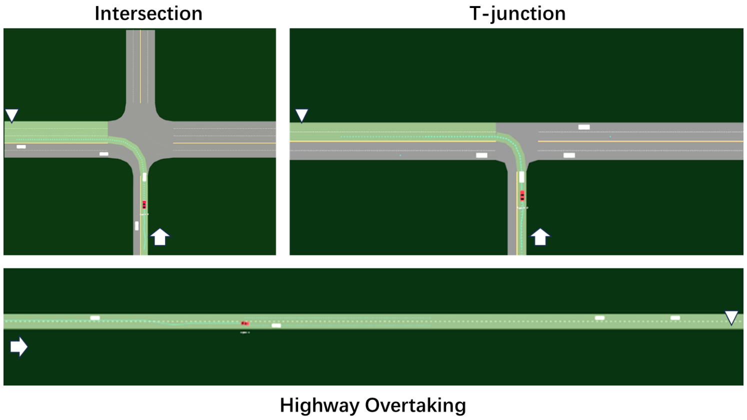

Figure C.3: The three autonomous driving scenarios in SMARTS benchmarks. Arrows are the entering lane of the ego vehicle, and triangles are goal positions. Highlighted green lanes are the "on-route" areas for the ego vehicle. White boxes are surrounding traffic vehicles.

The reward and cost design follows the minimalist principle:

$$R = r_{\text{distance}} + r_{\text{goal}}. \tag{C.2}$$

The first term is the travelled distance (in meters) within one decision step ($0.25\,\mathrm{s}$). The second term is +30 if reaching the goal. The cost is 0 when staying safe. When collisions, off-road, driving on the wrong side of the road, or off-route happen, the cost is -10. The first three situations also trigger the termination of the episode.

We hereby give a short discussion about our reward and cost design that might be useful for interested readers. We actually tried many other different designs, but this simplest version works the best. The observed issues of other settings are summarized below:

- *Do not terminate the episode when an unsafe event happens:* This is similar to the method used in Safety Dreamer's MetaDrive task (Huang et al., 2023). However, in our intersection and T-junction scenarios, due to the complexity of the road layout, the replay buffer is filled with meaningless, unsafe cases in the early stage of training. For example, when the ego vehicle drives off-road, it may stay there for a long time until the episode ends. This severely hinders policy learning.
- *Assign different costs to different unsafe events:* Many RL studies on autonomous driving tasks (e.g., MetaDrive (Li et al., 2022)) give a higher penalty for severe events like collisions, and a smaller penalty for traffic rule violations. In our trials, we found that the agent tends to do "reward-hacking" in such settings. For example, the vehicle will choose to drive off-road to get a lower penalty instead of learning how to avoid collisions. This reward-hacking is particularly severe when the vehicle needs to do a series of actions to solve the final potential collision, as is our case (restricting the acceleration and steering rate).

- *Use risk field or Surrogate Safety Measures (SSMs) as costs:* Using SSMs (Wang et al., 2021), such as Time-to-Collision (TTC) or risk field, to shape the reward is also a widely-used technique in RL-based autonomous driving. Our trials found that using TTC and the capsule risk field can indeed accelerate learning in the early stage. However, the final performance is worse than our simplest setting. One of the possible reasons could be that these SSMs add inductive biases to safety. They focus on one or several specific types of unsafe (potential collision) cases. This may restrict the exploration power of RL. The simple end-oriented costs, in contrast, can encourage exploring diverse and better solutions.

Both policy and critic networks use the WayFormer (Nayakanti et al., 2023) structure. For reward and cost critics, they share the torso and use different MLP heads to give multiple predictions of returns. Their network structures are briefly illustrated in Figure C.4.

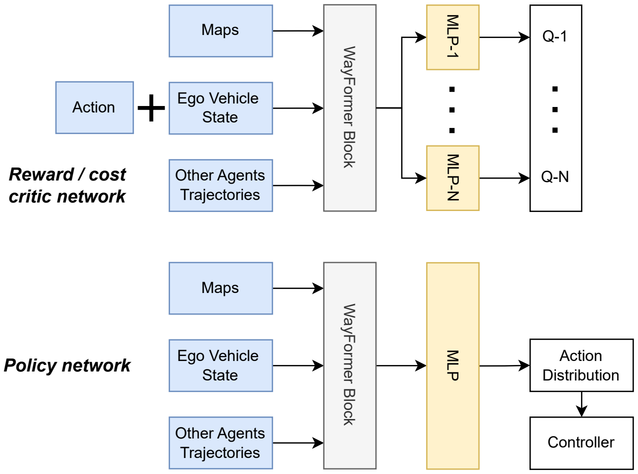

Figure C.4: The policy and critic network structure for SMARTS.

D Hyperparameter settings

For on-policy baselines, we use the same 1M step hyperparameter settings recommended by the OmniSafe benchmark platform (Ji et al., 2024) for all experiments. Details are provided on their public webpage https://github.com/PKU-Alignment/omnisafe.

For COX-Q, the implementation is based on SAC (Haarnoja et al., 2018). The shared parameters are listed in Table D.1, and the environment-specific parameters are listed in Table D.2. For CAL (Wu et al., 2024), we use the same hyperparameters in the original paper, except for the randomized ensemble and the UTD ratio (1 in our experiments). The code of ORAC (McCarthy et al., 2025) is not available yet. For safe navigation tasks, we use the recommended hyperparameters in the ORAC paper in our own implementation. While for SafetyVelocity-v1 and SMARTS, we did not find a proper set of hyperparameters for the original IQN-based ORAC. The performance is quite unstable. Therefore, we choose to modify it based on our TQC-based implementation. We explicitly mark that the used ORAC models are based on TQC or IQN throughout the experimental section. For all off-policy methods, we use the same discount factor and episode length listed in Table D.2 for consistency.

To accelerate the training for SafetyVelocity-v1 and SMARTS, we use a high number of parallel environments (128) and a lower offline update frequency (64). These choices are based on the recommendation of Brax (Freeman et al., 2021).

E Supplementary results

The performance of on-policy baselines on safe navigation tasks is listed in Table E.1, and the learning curves are presented in Figure E.1. Although they adhere to the cost constraints, the rewards

Table D.1: Shared off-policy parameters

Parameters	Value)
Policy learning rate	3e-4
Critic learning rate	3e-4
Entropy learning rate	3e-4
Entropy auto-tuning	True
Batch size	256
Tau	0.005
Convexification c	10
Number of quantiles M	25
Number of cost critics	5
Number of reward critics	5

Table D.2: Environment-specific off-policy parameters

Parameters	SafeVelocity-v1	SafeNavigation	SMARTS
Episode length	1000	400	240
discount factor γ	0.99	0.975	0.975
Episode cost limit	25	10	0.01
Number of parallel envs	128	1	128
Gradient steps	64	1	64
Policy update steps	64	1	64
Lagrangian initial value	1	0	1
Lagrangian learning rate	3e-4	5e-4	3e-4
Step length auto-tuning learning rate	1e-4	1e-4	NA
Initial steps	16380	5000	5120
Buffer size	1024000	1000000	512000
Policy network	256×2	256×2	complex
Critic network	256×5	256×2	complex
Layer Normalization	False	False	NA
Truncation (k_r, k_c)	(5, 5)	(0, 0)	(1, 0)
Optimism (β_r, β_c)	(3, 3)	(4, 1)	(3, 3)
Maximum step length η_{KL}	3	4	3
Cost CVaR α	25	13	13
Target update frequence	64	2	64

are significantly lower than off-policy methods due to the low sample efficiency. Table E.2 further lists the numerical results of off-policy methods for comparisons.

Figure E.2 gives the percentage of triggered exploration gradient conflicts for the first 200K steps in the safe navigation benchmark using COX-Q. We see that the reward and cost objectives rarely conflict with each other ($< 10\%$); therefore, the differences between ORAC and COX-Q are small. We hereby give a possible explanation. First, just like in conventional multi-task learning, the gradient conflicts often happen between two loss functions with significantly different scales. However, for safe navigation, both reward and cost are on the same scale (0-30). Second, as both reward and cost are sparse signals (or at least highly skewed), most exploration gradients are near zero, making it highly stochastic.

F The Use of Large Language Models (LLMs)

LLMs are used for polishing writing only, such as selecting proper words.

Table E.1: Performance of on-policy baselines (1M steps) on safe navigation (mean ± std)

Environment	Metric	CUP	PPOSimmerPID	PPOSaute	RCPO	CPPOPID
PointButton2	Return	6.1 ± 3.4	-0.7 ± 2.3	1.1 ± 0.3	10.7 ± 1.8	5.9 ± 3.9
	Cost	11.7 ± 6.6	7.4 ± 7.4	12.8 ± 9.6	5.4 ± 1.8	9.6 ± 2.6
PointGoal2	Return	3.2 ± 2.9	2.1 ± 0.2	2.5 ± 0.3	4.1 ± 1.5	2.3 ± 1.3
	Cost	10.3 ± 12.6	9.7 ± 4.9	15.9 ± 11.7	5.1 ± 3.8	9.1 ± 6.2
CarButton2	Return	0.8 ± 0.6	-0.9 ± 1.8	0.6 ± 0.2	4.1 ± 1.7	0.8 ± 0.3
	Cost	6.5 ± 6.6	2.2 ± 1.3	9.3 ± 3.0	6.0 ± 3.7	9.5 ± 2.7
PointPush1	Return	1.9 ± 1.1	3.3 ± 0.8	4.1 ± 2.3	2.9 ± 0.7	3.3 ± 2.3
	Cost	3.1 ± 5.2	0.6 ± 0.8	1.1 ± 2.0	1.1 ± 1.4	1.9 ± 3.6

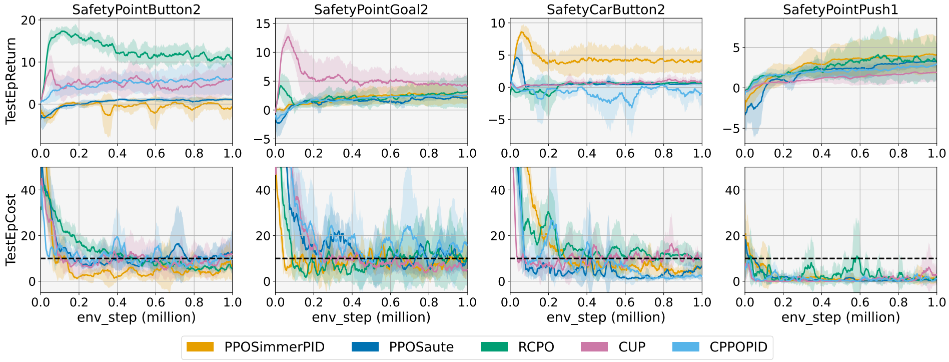

Figure E.1: Training curves of on-policy baselines for safe navigation tasks

Table E.2: Performance of off-policy baselines (1M steps) on safe navigation (mean ± std)

Environment	Metric	SACPID	CAL	IQN-ORAC	COX-Q
PointButton2	Return	7.3 ± 9.9	23.9 ± 4.1	35.6 ± 0.3	35.5 ± 2.2
	Cost	13.5 ± 8.5	3.6 ± 1.7	5.7 ± 0.6	6.1 ± 0.6
PointGoal2	Return	1.7 ± 1.4	5.9 ± 2.6	23.1 ± 2.0	21.0 ± 2.6
	Cost	10.4 ± 7.6	2.4 ± 1.7	6.1 ± 1.4	6.0 ± 1.6
CarButton2	Return	26.6 ± 5.3	7.8 ± 2.3	17.6 ± 1.9	22.8 ± 3.1
	Cost	28.9 ± 10.1	3.4 ± 0.8	8.7 ± 3.0	6.7 ± 1.0
PointPush1	Return	10.4 ± 7.3	7.5 ± 7.1	15.5 ± 7.1	17.1 ± 8.1
	Cost	0.4 ± 0.5	0.2 ± 0.3	2.8 ± 1.7	3.0 ± 1.5

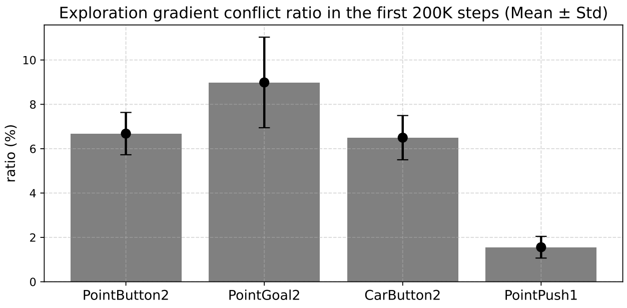

Figure E.2: Exploration gradient conflict analysis for safe navigation tasks