TAMING PHANTOM COSTS: INCREMENTAL LOWER BOUNDS FOR CONSTRAINED REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Constrained Reinforcement Learning (CRL) promises to improve the safety and efficiency of Reinforcement Learning under real world assumptions by replacing the need to tune uniterpretable reward weights with interpretable constraints. However, CRL comes with its own exploration challenges: Cost estimators suffer from random initialization errors and bad generalization which produces spuriously high cost estimates. Since high costs states are, by design, underexplored, those errors never get corrected and cause hallucinated "phantom costs" that needlessly constrain the CRL policy. Existing EM-style surrogate methods address this via monotone safety improvement, but their convergence speed is entangled with the scale of the Q-function through a manually tuned hyperparameter, leading to slow and brittle feasibility recovery as the optimization over- and under- estimates the local uncertainty. We propose ILB-CRL (Incremental Lower Bound CRL), which replaces the hyperparameter with an automatically computed trust coefficient α derived from a lower confidence bound (LCB) on the optimal safety improvement $G^\star$. This formulation bridges the gap between incremental improvements, hard projections, and statistical reliability of the cost estimates. We prove geometric convergence under very limited assumptions and demonstrate either above average or state of the art result on a large set of safety-gymnasium environments.

1 INTRODUCTION

While Reinforcement Learning (RL) has demonstrated remarkable efficacy in control and planning tasks by optimizing policies directly through environmental interaction (Sutton and Barto, 2018), its deployment is often hindered by safety concerns. Untrained agents frequently engage in behaviors that can damage hardware or endanger human operators. Furthermore, even after training it is not guaranteed that e.g., hardware constraints or safety constraints are upheld during deployment. Historically, such constraints have been modeled using reward shaping, which usually leads to highly uninterpretable "magic numbers" weighting individual contributors to rewards to prevent pathological or dangerous behavior. To give a concrete example: Berner et al. (2019) lists 21 contributing values to their reward, often with unintuitive weights (e.g., killing a hero produces highly negative reward).

Recent work has started considering Constrained Reinforcement Learning (CRL) as a powerful framework to mitigate these risks by mathematically enforcing constraints during the training phase. The standard formalism for this approach is the Constrained Markov Decision Process (CMDP), which augments the standard MDP with constraints on the policy's behavior:

$$\max_{\pi} R(\pi) \quad \text{s.t.} \quad C_i(s,a) \leq B_i \quad \forall i = 0, \ldots, N \tag{1}$$

Here, $R(\pi) = \mathbb{E}_\pi[\sum \gamma^t r_t]$ denotes the expected discounted return, and $C_i(s,a) = \mathbb{E}_\pi[\sum \gamma^t c_{i,t}]$ represents the expected discounted cost associated with the i-the constraint. In addition, one usually assumes some kind of constraint qualification, usually Slater's condition (Slater, 2014).[1] For the purpose of this study, we define $C(s,a)$ as the cost incurred by executing action a in state s (analogous to the Q-function). Constrained RL thus focuses on identifying a policy that maximizes reward while

[1] Notice that Eq. 1 is linear in occupation measure and therefore convex.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

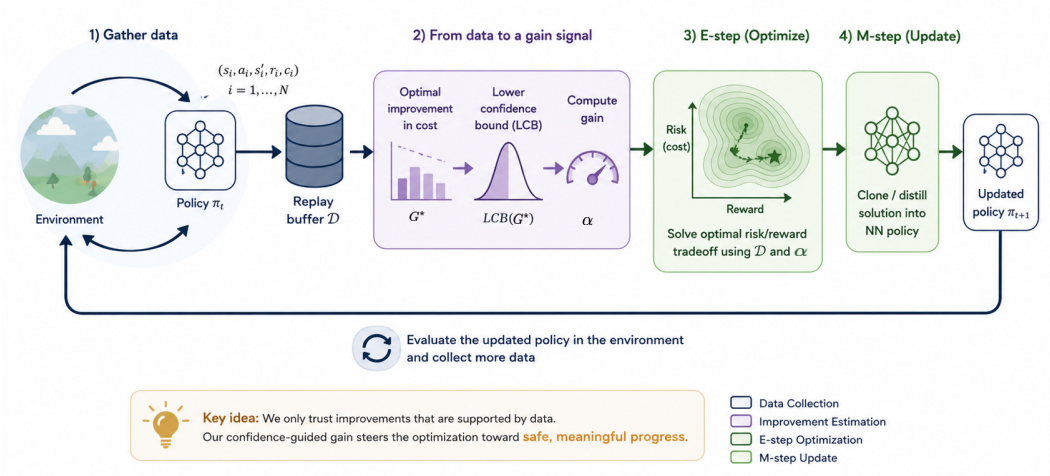


Figure 1: Overview of our method. We first gather data from the environment and store it in our replay buffer. From the buffer, we then compute the optimal improvement in cost $G^\star$ and the confidence in improvement α. The E-step then jointly optimizes safety $(G^\star, \alpha)$ and optimality $(\mathbb{E}_\pi[Q(a, s)])$. The M-step then clones the information into the neural network policy, which can finally be re-deployed inside the environment to gather more data.

remaining within the feasible region defined by Eq. 1. Explicit constraints are an easy way to ensure the designated behavior follows interpretable endstops (compared to e.g. reward coefficients), while also providing more information to the policy than a single scalar reward function.

The downside of CRL-based approaches is the need for new RL algorithms that are capable of handling complex compound objectives arising from the usage of constraints. One specific problem is the poor sampling behavior first noted by Mattick et al. (2026): In a finite sample setting, costs $C_i(s, a)$ may be randomly initialized to large numbers or have a high value due to errors in credit assignment or unlucky trajectories. Putting high trust into such states leads to a pathological response: The cost is high, meaning the state will be avoided. Since the state is avoided, no new samples arrive to correct the wrong $C_i(s, a)$ estimate. Since $C_i(s, a)$ stays high, the phantom constraint stays active. This means the constrained environment the agent "sees" will be vastly more complex than the true environment. It is important to note that this is slightly different from traditional RL exploration issues, as RL can be built to naturally explore uncertain regions by adding intrinsic rewards. CRL does not naively admit such an exploration since adding such a bonus is at odds with aiming to be as safe as possible during training. We further hypothesize that this issue will be more pronounced the better CRL algorithms try to enforce their constraints.

In this work we propose a novel CRL method that handles constraints by incrementally improving the constraint lower bound towards the feasible region. This side-steps traditional issues with over-trusting uncertain states and instead encourages the model to make statistically well-supported choices. We build our method on top of the seminal work from Mattick et al. (2026) in incorporating incremental improvements but provide cleaner convergence guarantees and nicer numerics. In essence, we provide the following contributions. First, we replace the brittle κ improvement with an optimal estimate of the possible improvement. We then connect this with insight from chance-constrained optimization to account for epistemic errors in the cost estimator. Finally we prove convergence using minimal assumptions on a very general set of cost estimators.

2 Related Work

Previous work on CRL can be broadly categorized into four categories, outlined as follows.

Inequality-based approaches such as Constrained Policy Optimization (CPO) (Achiam et al., 2017) or Interior Point Policy Optimization (IPO) (Liu et al., 2019) utilize worst-case bounds on the costs and design algorithms around maintaining feasibility once it is achieved. The latter is also one of the main issues with this line of research: If the current policy is unsafe, these methods usually rely on

some form of direct descend along the cost which can exacerbate issues with hallucinated high-cost states. The bounds are usually also only statistical and not tight, meaning they are generally not tight (since they need "space" for variance) or they allow for frequent excursions out of the feasible region.

Projection-based approaches are a direct descendent of the inequality-based approaches, with the main difference being that instead of adapting the update itself, the methods repeatedly re-project updates onto the current estimate of the feasible set. Examples of these methods are PCPO (Narasimhan, 2020) and CUP (Yang et al., 2022). As these methods stem from the first group, they have the same issue of trusting the approximate cost estimate too much.

Lagrangian-based approaches solve the min-max problem induced by Eq. 1 direcly by ascending the reward and descending on the dual variables. Examples of this are PPO-Lag (Achiam and Amodei, 2019) which proposes straight ascend-descend, PID-based methods (Stooke et al., 2020) which aim to reduce oscillations in the duals, and RCPO (Tessler et al., 2018) which includes the Lagrange-multiplier as a signal to the value function.

Surrogate-based approaches such as First Order Constrained Policy Search (FOCOPS) (Zhang et al., 2020) first solve a surrogate problem which is later transferred into the neural network. This comes at the cost of needing to solve a potentially complex surrogate problem, but has the potential to ensure safety to a much higher degree as constraints can be handled using advanced techniques from mathematical optimization. A very recent line of work aims to accomplish this by introducing surrogate problems based on the Maximum A-Posteriori Policy Optimisation (MPO) (Abdolmaleki et al., 2018) algorithm. MPO uses a two step approach: First, a convex nonparametric problem is solved to optimality, which in a second step is cloned into the neural network. This structure is appealing for CRL as it allows one to take advantage of efficient constrained optimizers. CVPO (Liu et al., 2022) simply adds the constraints from Eq. 1 into the surrogate problem proposed by Abdolmaleki et al. (2018). However, this has the disadvantage of providing degenerate problems in cases where a safe policy does not exist within a ε KL-ball. Furthermore, this formulation implicitly assumes that the learned cost function $C(a, s)$ is 100% reliable and generalizes, which is not true in practice.

3 PRELIMINARIES: SAFEMPO

Our work fits into the last category: Our setup is similar that from Mattick et al. (2026) with the main difference being that instead of relying on a logarithm-style monotonic improvement, we estimate the *largest statistically justifiable improvement in cost*.

For this, we start with the ELBO setup introduced by Mattick et al. (2026):

$$\log P_\pi(O = 1, S = 1) = \log \int p_\pi(\tau) p(O = 1|\tau) p(S = 1|\tau) d\tau \tag{2a}$$

$$\geq \mathbb{E}_q \left[\sum_t r_t/\alpha \right] + \mathbb{E}_{\tau\sim q}[\log p(S = 1|\tau)] - KL(q(\tau)\|p_\pi(\tau)) \tag{2b}$$

where

$$p(O = 1|\tau) \propto \exp\left(\sum_t r_t/\alpha\right) \qquad p(S = 1|\tau) \propto \begin{cases} \exp\left(\frac{-(C(a,s)-B)}{\beta}\right) & C(a, s) \geq B \\ 1 & \text{otherwise} \end{cases} \tag{3}$$

are the likelihoods for "Policy is **O**ptimal" and "Policy is **S**afe" respectively. Mattick et al. (2026) defines the shorthand

$$G_{\text{MPO}}(a, s) = \log p(S = 1|\tau) = -\max\left(\frac{C(a, s) - B}{\beta}, 0\right). \tag{4}$$

for the safety likelihood function.

This objective can then be optimized using the Expectation-Maximization technique (Neal and Hinton, 1998; Abdolmaleki et al., 2018), which is essentially a coordinate descend scheme on the q and π distributions. For this one samples batches of N states $p(s)$ from the replay buffer and computes K actions for each of them from $\pi(a|s)$. The resulting $K \times N$ tensor can then be used to optimize Eq. 2 nonparameterically in q (the E-step). The resulting q can then be "cloned" into π by minimizing

$\min_\pi \{KL(q\|\pi)\ s.t.\ KL(\pi_{\text{old}}\|\pi) \leq \varepsilon\}$ (the M-step). This efficiently turns policy optimization into an alternating optimization of a surrogate q and a supervised learning problem of cloning q into π. [2]

4 Incremental Lower Bound CRL (ILB-CRL)

While SafeMPO is conceptually sound, in practice it hits three notable limitations: **First**, we find that the safety likelihood function used in SafeMPO is slightly misaligned to the CMDP constraint when randomness is present in the environment. **Second**, the convergence of SafeMPO may be arbitrarily slow due to its usage of a log-barrier function. **Third**, the uncertainty in cost is not calibrated to a true uncertainy measure and is instead implicitly fixed to always assume low confidence κ. In this section we will tackle all three of these concerns in turn.

4.1 Safety Likelihood Function

The first limitation we notice is precisely due to the definition of the safety likelihood in Eq. 4: Notice that the safety likelihood is defined *per trajectory* not *per policy*. This is theoretically fine and may even be identical in some circumstances, but not all of them. Consider a setting with a cost limit of 25 and two trajectories, a lucky τ_1 with costs $C_1 = 0$ and an unlucky one with an irreducible cost $C_2 = 50$ (e.g., the environment is initialized in a collision course already). The per-trajectory constraint would see C_2 as unsaveable, hence no improvement, and C_1 as underexploiting its cost: It could choose riskier actions and remain below the cost limit. However, in a CMDP sense, the policy is in aggregate exactly at its limit $\frac{50+0}{2} = 25$. Repairing this is straight forward: we replace the conditional on one trajectory with a conditional on a set of trajectories

$$G_{\text{ILB-CRL}}(a, s) = \log p(S = 1|\tau_{1:M}) \propto -\frac{\sum_{t=0}^{T} \gamma^t c_t - B}{\beta}. \tag{5}$$

This aligns the safety likelihood with the notion of safety inside CMDPs.[3] We now instead clip the likelihood target rather than the per-trajectory likelihood. Notice that in contrast to the value function this contains both the costs-to-go *and* the already consumed costs. In this work we approximate the total cost using a learned critic and take the prefix directly from the replay buffer. We also use state augmentation with the consumed budget.

4.2 Greedy Safety Improvement Model

Aside from the existing safety likelihood being misaligned, the main issue with a naive ascend on the ELBO is that Eq. 2 introduces a multiobjective optimization problem, not a constrained problem: The likelihood of a policy can increase $P(O = 1|\tau)$ but decrease safety $P(S = 1|\tau)$. Unfortunately, Mattick et al. (2026) (Theorem 1) proves that simply enforcing

$$\mathbb{E}_{(a,s)\sim q}[G(a, s)] \geq \mathbb{E}_{(a,s)\sim \pi}[G(a, s)] \tag{6}$$

is insufficient to guarantee convergence to the feasible set even in the limit of infinite steps as no proper contraction is guaranteed to take place.[4] Mattick et al. (2026) resolves this by using an interior-point guarantee on q that ensures the inequality from Eq. 6 is strict, while still guaranteeing that a solution exists. The downside of this is that improvement in $G(a, s)$ might be incredibly slow: Their proof of convergence relies on a technical argument based on the fact that returns are assumed to be bounded $Q(a, s) \in [-M, M]$, while a log barrier function $\log(\mathbb{E}_{\tau\sim q}[G(a, s)] - \mathbb{E}_{\tau\sim\pi}[G(a, s)])$ is unbounded. This means there will always be a nonzero improvement where trading reward for cost improvement is "worth it". While true, this means that convergence speed depends on the range of Q and does not converge to the feasible space with an optimal rate.

To account for this, Mattick et al. (2026) introduce a κ-coefficient to weight the contributions of the log-barrier against Q. According to their experiments the impact of this κ is small, but that might

[2] While we restrict ourselves to the analysis of the classic MPO (Abdolmaleki et al., 2018), all methods discussed here work with V-MPO (Song et al., 2019) as well.

[3] This does not mean the SafeMPO safety likelihood is "wrong", it just captures a different notion of safety than what is necessary for CMDPs.

[4] In fact, the proof works for a very general class of order-preserving functions. For the sake of this work, we will restrict ourselves to the natural "Linear Safety Preserving Function".

also be due to all tested Q functions being in a narrow range of $\approx [-1, 30]$. The parameter κ itself is hard to tune optimally, since it depends on both the range of Q *and* the trust one has in the estimation accuracy of the expected gain $\mathbb{E}_{(a,s)\sim q}[G(a,s)] - \mathbb{E}_{(a,s)\sim\pi}[G(a,s)]$.

Our insight is the following: Notice that the maximal improvement in terms of G is itself easily determined by solving the problem

$$G^\star = \max_{q:\mathcal{P}(A\times S)} \mathbb{E}_{\tau\sim q}[G(a,s)] \tag{7a}$$

$$\text{s.t. } KL(q\|\pi) \leq \varepsilon \tag{7b}$$

In fact, from an optimization point-of-view, this problem is equivalent to the MPO E-step (Abdolmaleki et al., 2018) applied to the safety-likelihood. One could simply enforce that the term $\mathbb{E}_{\tau\sim q}[\log p(S=1|\tau)]$ in the ELBO (Eq. 2) equals $G^\star$. Since $G^\star$ need not improve beyond the safety threshold, we clip it at zero (notation $G^\star_+$).

4.3 Calibrated Uncertainty

One key issue with this is that $G^\star$ is highly sensitive to approximation errors. This is same problem discussed in Sec. 1: Hallucinated costs mean states stop being explored, which leads to phantom constraints never being corrected.

This means that realistically, one needs to solve an improvement equation of the following form

$$\max_{q:\mathcal{P}(A\times S)} \mathbb{E}_{(a,s)\sim q}[Q(a,s)] \tag{8a}$$

$$\text{s.t. } KL(q\|\pi) \leq \varepsilon \tag{8b}$$

$$\mathbb{E}_{(a,s)\sim q}[G(a,s)] \geq \mathbb{E}_{(a,s)\sim\pi}[G(a,s)] + \alpha(G^\star_+ - \mathbb{E}_{(a,s)\sim\pi}[G(a,s)]) \tag{8c}$$

Here α acts as a "trust" coefficient: if $\alpha = 1$ we believe $G^\star$ completely (No sampling errors, no function approximation errors, etc). If $\alpha \to 0$ we believe $G^\star$ is really noisy and cannot be trusted (and $\alpha = 0$ makes the constraint unproductive (Mattick et al., 2026)). Notice that this problem always has a solution for any $\alpha \in [0, 1]$ and if $\alpha \geq \alpha_{\min} > 0$ we obtain convergence towards the feasible set *iff* $G(a,s)$ can model the function, and we have a nonzero improvement implied by $G^\star$. The latter already follows immediately by (Mattick et al., 2026), which shows a lower bound on the improvement in safety exists. The former will be proven in Theorem 1. Also note that the above is trivially convex (KL and linear inequality) and has an interior point if $\alpha < 1$ since then an interior point has to exist in the form of the solution to Eq. 7. This means by Slater's condition, we know that strong duality holds and we can instead solve the dual problem (which has only two variables).

While this is a fine idea, in practice it gains very little: We have simply replaced the "magic" κ parameter from Mattick et al. (2026) with our own α parameter! Luckily, there is a key difference in κ and α: We can estimate an optimal α based on the trust we have in our function approximation. This effectively turns this into a **chance-constrained optimization** problem where we try to consider the "chance" that our estimator is wrong. Specifically, consider N samples from the distribution over parameters of $G(a,s|\theta)$. We can compute a lower confidence bound

$$\mu_G \geq \hat{G}^\star - g(\hat{V}, c, N, \delta) \tag{9}$$

where $\hat{V}$ is the estimated variance along N samples, and δ is the confidence level, and c is a bound on the RV. Notice that LCBs, like student-t based lower bounds fit this description. We can now use an LCB to guide the rate of improvement:

$$\alpha = \text{clip}\left(\frac{LCB - \mathbb{E}_{(a,s)\sim\pi}[G(a,s)]}{\hat{G}^\star_+ - \mathbb{E}_{(a,s)\sim\pi}[G(a,s)]}, 0, 1\right) \tag{10}$$

Note that the clipping here is to introduce prior knowledge: being better than $G^\star$ is impossible, just like being worse than the already existing policy $G_{\text{old}} = \mathbb{E}_{(a,s)\sim\pi}[G(a,s)]$. Notice that since α is just a stepsize, we only need to solve eq. (7) once and can then compute α with the fixed q weights.

To estimate $G(a,s|\theta)$ over θ, there are many options: The classical one are deep ensembles (Wild et al., 2023) and Laplace approximations (Daxberger et al., 2021). In this work we use the insights from (Osband et al.) that showcases simple bootstrapping with randomly initialized heads and masking is sufficient for training diverse ensembles.Interestingly, we can prove convergence even if we only have finitely many ensemble members under some limited assumptions

Theorem 1. *Let $G(a, s|\theta)$ be the likelihood over a compact domain $(a, s) \in \Omega$ and supported on a datset D_m of size m. For fixed $N \geq 2$ let $\theta_1, \ldots, \theta_N$ be posterior samples from $p(\theta|D_m)$. Let LCB $= \hat{\mu} - g(\hat{V}, c, N, \delta)$ be any LCB, such that $P(\mu \geq LCB) \geq 1 - \delta$, $g < \infty$ for $\delta \in (0, 1)$ and $g(\hat{V}, c, N, \delta) \to 0$ as $\hat{V} \to 0$.*

Assume that

1. $\forall \theta \in p(\theta). \|G(\cdot\|\theta)\|_\infty \leq B < \infty$
2. *Posterior consistency* $\forall \varepsilon > 0, P(\|G(\cdot|\theta) - G(\cdot|\theta^\star)\|_\infty > \varepsilon|D_m) \xrightarrow{m\to\infty} 0$

Let

$$\Delta = G^\star_+ - \mathbb{E}_\pi[G] > 0 \text{ and } \hat{\Delta}_m := \hat{G}^\star_{m,+} - \mu_{\pi,m} \xrightarrow{m\to\infty} \Delta$$

Then there exists some lower bound $\underline{\alpha} \in (0, 1)$ s.t.

$$P(\alpha_m > \underline{\alpha}) \xrightarrow{m\to\infty} 1,$$

Meaning α converges in probability to a nonzero value.

Corollary 1. *Since $\alpha \geq \underline{\alpha} > 0$ we get geometric convergence in probability with rate at least $\underline{\alpha}$.*

Remark 1. *Theorem 1's assumptions are intentionally weak. Assumption 1 is a boundedness condition ensuring the estimator and confidence bounds are well defined. In practice this can be trivially enforced via clipping, though one can also implicitly set the bound at the numerical precision range. Assumption 2 is a standard posterior consistency assumption that states that, given sufficient data, the posterior puts vanishing mass on suboptimal estimators. Crucially, this is a function space argument, meaning one has to identify the optimal function not parameters, preventing issues with e.g., overparameterization. The theorem does not require any particular contraction rate, nor does it require perfect recovery under finite datasets, just eventual recovery as data tends to infinity.*

Remark 2. *Of course the existence of such a bound for finite N does not guarantee one can find the optimal G by simple e.g. gradient descend. In practical deployment, one can still easily guarantee geometric convergence by replacing the* 0 *in eq.* (10) *with some small ε.*

Once $G^\star$ and α are known, the resulting problem can be solved very quickly using the dual

$$\min_{\lambda\geq 0, \eta \geq 0} \eta\varepsilon + \eta\mathbb{E}_{s\sim\mu} \log \mathbb{E}_{a\sim\pi(a|s)} \left[\exp\left(\frac{Q(a, s) + \lambda G(a, s)}{\eta}\right)\right] - \lambda\bar{G} \tag{11}$$

where $\bar{G} = \mathbb{E}_{\tau\sim\pi}[G(a, s)] + \alpha(G^\star_+ - \mathbb{E}_{\tau\sim\pi}[G(a, s)])$ is the required gain. We solve this problem using scipy's L-BFGS-B optimizer (Virtanen et al., 2020) which converges in 2-10 iterations. Solving the E-step wasn't a bottleneck in any of our experiments. The optimal q is recoverable in closed form:

$$q^\star(a, s) = \frac{\pi(a, s) \exp\left(\frac{Q(a,s)+\lambda G(a,s)}{\eta}\right)}{\mathbb{E}_{(a,s)\sim\pi}\left[\exp\left(\frac{Q(a,s)+\lambda G(a,s)}{\eta}\right)\right]}. \tag{12}$$

4.4 Choice of LCB

The remaining question is how confidence bounds should be determined. One of the most used options for LCBs (or UCBs) is given by Hoeffding's inequality. Unfortunately, Hoeffding's inequality does not work nicely since it only scales with the range of the RV, not the variance. In fact, without assumptions on the shape of the RV this is always the case: All completely nonparametric LCBs have to bound the risk of the RV estimated being fat tailed. Fortunately, we have good reasons to assume this is not the case here: By a Bernstein-von Mises argument, we assume the posterior is asymptotically gaussian and ensemble members are iid draws from that distribution. This means we can obtain much better and more faithful estimates using a student-t based confidence bound

$$\mu_G \geq \hat{G}^\star - t_{N-1,\delta}\sqrt{\frac{\hat{v}}{N}}, \tag{13}$$

where $t_{N-1,\delta}$ is the $1 - \delta$ quantile for an $N - 1$ dimensional student-t distribution (see Appendix D)

We can improve this further by going into the Bayesian domain and including the knowledge that our $\Delta = G^{\star}_{+} - \mathbb{E}_{(a,s)\sim\pi}[G(a,s)]$ cannot be negative and, in fact, has to be positive as per (Mattick et al., 2026) [5]. Therefore we can assume that the improvement Δ follows a truncated normal distribution since we can remove all mass that corresponds to a negative improvement from the sample space:

$$\Delta|\text{data}, \Delta \geq 0 \sim \mathcal{TN}\left(\hat{\Delta}, \frac{\hat{v}_{\Delta}}{N}, 0, \infty\right), \tag{14}$$

where $\hat{\Delta}$ corresponds to the finite-sample approximation, $\Delta = \hat{G}^{\star}_{+} - \mathbb{E}_{(a,s)\sim\pi}[G(a,s)]$, and $\hat{v}_{\Delta}$ corresponds to the variance of the sample mean. The $\delta-$quantile can be computed as

$$\Delta^{+}_{\text{LCB}} = \hat{\Delta} + \frac{\hat{\sigma}_{\Delta}}{\sqrt{N}}\Phi^{-1}\left((1-\delta)\Phi\left(\frac{-\hat{\Delta}\sqrt{N}}{\hat{\sigma}_{\Delta}}\right) + \delta\right). \tag{15}$$

This incorporates both the normality assumption and the known non-negativity constraint as $(1-\delta)\Phi(\cdot) + \delta > \Phi(\cdot)$ we have $\Delta^{+}_{\text{LCB}} > 0$, ensuring $\alpha > 0$ without needing a minimum threshold m_0. In fact, this construction arguably fits the proof for Theorem 1 better, as the truncated $1-\delta$ quantile has Bayesian coverage of $1-\delta$ by definition and the positivity constraint is sound, meaning the proof of Theorem 1 works identically for this setting (see Appendix B). Indeed, using this perspective gives the natural interpretation of α as the fastest chance-constrained convergence rate possible:

Theorem 2. *Assume the current policy is infeasible (if it is, there is no improvement to trust and Equation* (8c) *is vacuous independent of α). Let $\Delta_{true}|data, \Delta_{true} \geq 0 \sim \mathcal{TN}(\hat{\Delta}, \hat{\sigma}^2_{\Delta}/N, 0, \infty)$ be the posterior over the true improvement, incorporating the structural constraint $\Delta_{true} \geq 0$. Let*

$$\alpha^{\star} = \max_{\alpha\in[0,1]} \alpha \quad \textit{s.t.} \quad P(\Delta_{true} \geq \alpha\hat{\Delta} \mid data, \Delta_{true} \geq 0) \geq 1-\delta. \tag{16}$$

Then:

1. *$\alpha^{\star} = clip(\Delta^{+}_{LCB}/\hat{\Delta}, 0, 1)$, where Δ^{+}_{LCB} is the δ-quantile of the truncated normal*
2. *$\alpha^{\star} > 0$ for all $\delta \in (0,1)$ and finite N;*
3. *Improvement Eq. 8 is feasible wrt the true risk with posterior probability at least $1-\delta$.*

Remark 3. *Theorem 2 proves that $\alpha^{\star}$ is not a hyperparameter but the closed-form solution to an auxiliary optimization: it is the* fastest achievable convergence rate *towards the feasible set that is statistically supported at the credibility level $1-\delta$ associated with the empirical safety-improvement constraint. We pick the standard setting of a 95% confidence/credibility level for this work, though this parameter can be freely increased or decreased.*

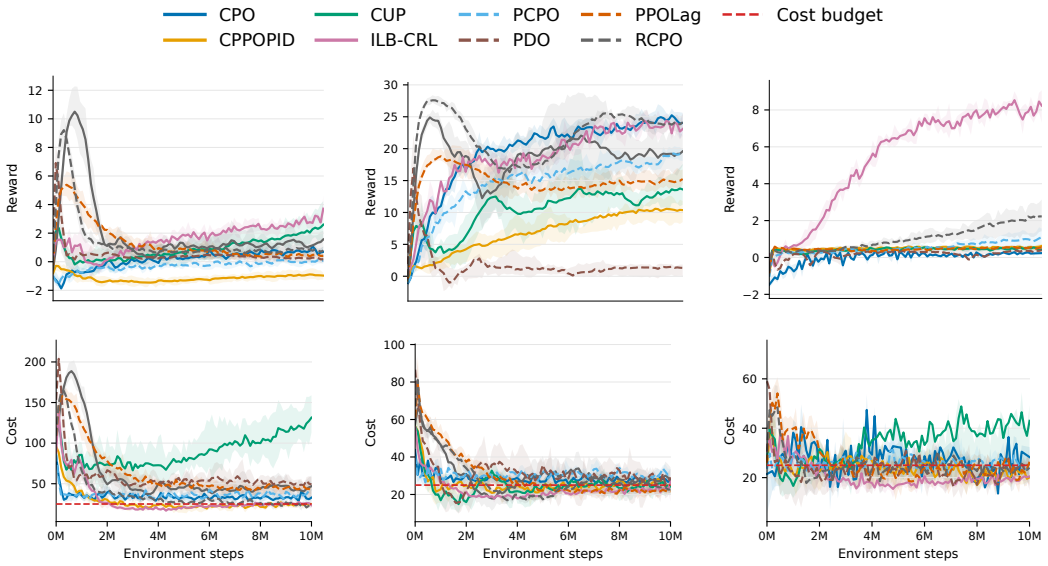
The complete method is shown in Figure 1: Gathered data is accumulated into the replay buffer, from which the training samples for asynchronous training (step 1). During training an optimal gain and confidence bound are computed (step 2) which can then be used to formulate the main risk-aware E-step (step 3). Finally the model is updated (step 4) and re-deployed into the environment. We provide a pseudocode for the training step in Section A.

5 Experiments

5.1 Comparisons against the state of the art

Benchmarking Deep RL algorithms is notoriously difficult: For this reason, we adopt the best practices presented by Agarwal et al. (2021).We compare ILB-CRL amongst a wide set of environments and algorithms. In particular, we extend the commonly evaluated set of navigation tasks with the much less commonly evaluated CarPush, AntGoal and HalfCheetah tasks. The latter three tasks evaluate rigid-body interactions (CarPush), and high dimensional Control Problems (AntGoal/HalfCheetah) by producing constrained RL tasks that utilize the robots from the Deepmind control suite (Tassa et al., 2018). Just like previous experiments by e.g. Ji et al. (2024) we set the constraint threshold to

[5] Intuitively, the optimal gain cannot be negative since we already know π itself. The positive gain follows from slater's condition forcing the existence of an (potentially small) improvement direction around π.



(a) CarButton1 **(b) CarGoal1** **(c) CarPush1**

Figure 2: Training performance across the Car tasks. Lines show the mean across seeds and shaded regions indicate the interquartile range. Reward is shown above cost. Our method is generally the one of or the highest performing across these settings during the entire timeframe.

Algorithm	Metric	AntGoal1	CarButton1	CarGoal1	CarPush1	HalfCheetahVel.	PointGoal1
CPO	Return	−0.0 (−0.0, −0.0)	0.7 (0.5, 0.9)	24.4 (23.2, 25.0)	−0.0 (−0.2, 0.1)	2252.9 (1798.9, 2740.5)	19.9 (19.0, 20.8)
	Cost	0.4 (0.4, 0.4)	32.8 (31.6, 34.1)	27.7 (27.3, 28.2)	27.6 (26.1, 29.1)	20.6 (18.8, 22.2)	27.4 (26.8, 28.4)
CPPOPID	Return	0.1 (0.0, 0.1)	−1.0 (−1.3, −0.6)	9.8 (8.2, 12.4)	0.5 (0.4, 0.7)	3000.8 (2780.7, 3022.3)	7.4 (5.9, 10.0)
	Cost	0.5 (0.4, 0.5)	23.9 (22.5, 25.5)	26.6 (24.3, 28.2)	21.1 (18.7, 22.9)	24.7 (24.3, 25.6)	26.2 (24.0, 28.5)
CUP	Return	10.6 (8.9, 11.6)	2.1 (1.1, 2.9)	12.8 (10.6, 14.6)	0.4 (0.4, 0.6)	1985.9 (1714.9, 2504.4)	9.5 (5.3, 14.7)
	Cost	19.9 (17.8, 21.5)	117.4 (89.2, 142.3)	23.7 (20.2, 28.4)	30.2 (23.6, 53.4)	18.7 (13.8, 23.3)	32.4 (27.5, 41.9)
PCPO	Return	5.8 (4.7, 9.7)	−0.0 (−0.2, 0.2)	18.4 (17.2, 19.2)	0.9 (0.5, 1.3)	1334.2 (1294.0, 1366.5)	17.1 (16.0, 18.0)
	Cost	14.1 (11.8, 18.3)	36.5 (34.5, 38.9)	30.7 (30.1, 31.4)	25.9 (24.3, 27.3)	19.0 (17.1, 19.8)	28.5 (28.0, 29.1)
PDO	Return	1.9 (1.4, 2.8)	0.2 (0.0, 0.5)	0.8 (−0.1, 2.7)	0.4 (0.1, 0.5)	2656.9 (2166.4, 2761.9)	2.5 (0.5, 6.6)
	Cost	2.1 (1.6, 3.4)	44.1 (35.8, 66.0)	26.3 (18.7, 37.0)	22.7 (19.6, 25.1)	24.2 (23.7, 24.7)	26.7 (22.0, 31.0)
PPOLag	Return	0.1 (0.1, 0.1)	0.4 (0.2, 0.7)	15.0 (13.2, 16.5)	0.5 (0.4, 0.6)	3011.1 (2818.5, 3022.9)	13.9 (10.4, 15.5)
	Cost	0.6 (0.5, 0.6)	44.5 (40.7, 49.7)	22.1 (20.1, 23.9)	23.8 (20.6, 25.9)	22.0 (21.4, 23.6)	23.8 (21.2, 28.7)
RCPO	Return	15.2 (13.9, 16.6)	0.7 (0.4, 1.2)	24.1 (22.9, 25.6)	1.9 (1.2, 2.7)	2333.6 (1878.5, 2803.6)	14.8 (12.1, 18.4)
	Cost	24.9 (23.0, 25.9)	24.4 (22.3, 26.9)	25.0 (24.1, 26.6)	23.1 (22.0, 24.8)	24.1 (23.3, 25.1)	19.9 (17.7, 22.8)
ILB-CRL@5Mio	Return	7.8 (5.3, 10.0)	1.6 (0.9, 2.6)	18.5 (17.2, 19.9)	6.8 (6.1, 7.7)	2265.2 (2195.4, 2335.0)	15.8 (14.2, 17.1)
	Cost	8.0 (5.5, 11.0)	19.5 (17.8, 23.8)	19.9 (18.7, 20.7)	17.8 (16.9, 18.2)	18.9 (14.9, 22.9)	20.1 (18.6, 23.4)
ILB-CRL@10Mio	Return	14.6 (12.1, 17.0)	2.8 (2.5, 3.2)	22.9 (20.3, 25.0)	8.4 (7.7, 9.1)	2501.1 (2414.6, 2587.6)	18.2 (16.5, 19.5)
	Cost	14.3 (10.9, 15.8)	25.4 (23.4, 27.7)	22.6 (21.2, 25.7)	20.2 (18.3, 21.5)	21.1 (18.6, 23.6)	24.3 (22.6, 27.0)

Table 1: Return and cost (IQM) with 95% confidence intervals for each environment.

$B = 25$. All ICB-CRL tests were performed with identical hyperparameters (aside from obvious changes in e.g. input and output dimensions) and baselines use the settings from (Ji et al., 2024).

Table 1 shows the results of state-of-the-art CRL algorithms with their interquartile mean and 95% confidence interval across 10 different seeds (Agarwal et al., 2021) for both reward and cost. ILB-CRL demonstrates superior (CarButton, CarGoal, CarPush, PointGoal) or competitive (AntGoal, HalfCheetah) performance compared to existing methods. In particular, the method shows a large performance gap in complex maneuver scenarios like CarButton and CarPush, where dynamic physics objects like obstacles and push-blocks move during the experiment making cost prediction difficult. CarButton is also particularly good at demonstrating the foils of hard-projection type algorithms: CUP (one of the most prominent projection methods), while exceeding our model in reward, is not capable of locating the feasible set. In CarGoal and PointGoal, our method is highly competitive while satisfying the constraints. For the deepmind-style robotics environments, our model is above average (beating 4 and losing to 3) but not leading in performance. Notably, the best performing methods for

Method	SafetyCarGoal		SafetyCarButton		SafetyPointGoal	
	Reward	Cost	Reward	Cost	Reward	Cost
SafeMPO (Mattick et al., 2026)	21.43 ± 5.10	32.23 ± 7.43	0.67 ± 0.61	30.87 ± 4.47	13.09 ± 2.79	32.92 ± 3.43
ILB-CRL + eq. (4)	18.10 ± 0.47	21.94 ± 2.13	2.94 ± 0.41	49.02 ± 11.54	13.61 ± 1.18	31.30 ± 3.34

Table 2: Comparison of our Method against SafeMPO using the SafeMPO likelihood function

HalfCheetahVelocity are Lagrangian+PPO variants. This makes sense since the HalfCheetahVelocity benchmark was created by taking a vanilla PPO agent's final velocity and adding a constraint at 50% of that speed. This might lead to PPO-like algorithms being favored in this environment due to learning behaving like the reference PPO until the constraint is hit, especially considering these methods do not tend to perform well on other environments. In general, we find a meaningful qualitative difference between the cost distributions depending on algorithm choice. We present behavioural differences in Appendix F. While we do not claim any behavior is superior to any other (all satisfy the CMDP objective), we do think it might explain large environment-to-environment differences common in CRL. ILB-CRL also demonstrates high performance early (see Table 1@5Mio) which is not unexpected due to off-policy's data efficiency. We also perform experiments with fixed α rather than choosing it dynamically in Appendix H. Fixing α causes the algorithm to get stuck, particularly in complex CarButton1 environments. We also show the full timeseries for all car tasks in Figure 2.

5.2 Comparison against SafeMPO and Ablation of Safety Likelihood Function

While SafeMPO could be simply added to the above comparison table, the flexibility of our method can be used to make the comparison against SafeMPO more meaningful: Notice that one of the main differences of SafeMPO compared to ILB-CRL is the usage of a different notion of safety provided by the safety likelihood function Eq. 4. We can simply emulate this in our method by replacing our new safety likelihood Eq. 5 with the SafeMPO likelihood. Note that the other direction (replacing SafeMPO's likelihood with ours) would technically also work but necessitates a full re-tuning of the κ parameter. We were unable to make SafeMPO work with our safety likelihood.

We also utilize the mean+stddev benchmark of Mattick et al. (2026) instead of the full (Agarwal et al., 2021) bootstrapping estimation since we are not 100% confident it applies to the SafeMPO likelihood function (as it is a lot closer to "per trajectory" than "in aggregate"). The results can be seen in Table 2: SafeMPO demonstrates a steady-state cost error (see Mattick et al. (2026)) compared to our method's exact satisfaction. Interestingly, our method has the exact same property for all but CarGoal with an even more pronounced gap at CarButton. This suggests that the steady-state error depends more on the notion of safety than on the $\log$-Barrier gap: Note that our method is easily able to reach the same reward at a cost budget of 25 when using our safety likelihood function (see Table 1). Therefore, we argue that the notion of safety present in SafeMPO forces the model to see a higher "virtual" budget in some circumstances which causes the model to exceeed the threshold. This makes sense as SafeMPO punishes only trajectories exceeding the cost threshold, which means high cost trajectories cannot be compensated through much lower cost ones. This "no compensation" setup is interesting but not 100% aligned with the expected-cost notion used in many CRL tasks

This does however open up an interesting avenue for future work: If the behavior of the SafeMPO line of algorithms is readily influenceable by changing the safety likelihood and ILB-CRL allows for flexible adaptation of likelihoods without re-tuning, one can try to do so on purpose to follow e.g. Conditional Value at Risk or percentile constraints.

6 Conclusion

We proposed ILB-CRL, a completely off-policy uncertainty aware Constrained Reinforcement Learning technique. Our method improves on prior work by producing a superior safety-likelihood function, solving a greedy safety-improvement problem, and selecting the optimal uncertainty-weighted convergence rate. We demonstrate superior or high performance and data efficiency in standard constrained RL benchmarks using statistically well supported aggregations. Finally, we use the comparison between SafeMPO's and our likelihood function to motivate further research into more complex notions of safety like CVaR.

AI USE STATEMENT

We used generative AI tools for the overview figure and to implement basic functions (like the per-ensemble-member RETRACE, generating plotting scripts,...). The code was manually checked as well as tested on known input-output pairs (e.g. RETRACE should return 0 if all inputs are zero, a full TD(N) for likelihood ratios of 1, etc...). We also used generative models to double check our proofs, though this did not yield any actionable changes beyond the typographical. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Rémi Munos, Nicolas Manfred Otto Heess, and Martin A. Riedmiller. Maximum a posteriori policy optimisation. *ArXiv*, abs/1806.06920, 2018. URL https://api.semanticscholar.org/CorpusID:49310753.

Josh Achiam and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. 2019. URL https://api.semanticscholar.org/CorpusID:208283920.

Joshua Achiam, David Held, Aviv Tamar, and P. Abbeel. Constrained policy optimization. *ArXiv*, abs/1705.10528, 2017. URL https://api.semanticscholar.org/CorpusID:10647707.

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C. Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Neural Information Processing Systems*, 2021. URL https://api.semanticscholar.org/CorpusID:237353084.

Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemyslaw Debiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Christopher Hesse, Rafal Józefowicz, Scott Gray, Catherine Olsson, Jakub W. Pachocki, Michael Petrov, Henrique Pondé de Oliveira Pinto, Jonathan Raiman, Tim Salimans, Jeremy Schlatter, Jonas Schneider, Szymon Sidor, Ilya Sutskever, Jie Tang, Filip Wolski, and Susan Zhang. Dota 2 with large scale deep reinforcement learning. *ArXiv*, abs/1912.06680, 2019. URL https://api.semanticscholar.org/CorpusID:209376771.

Erik A. Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, M. Bauer, and Philipp Hennig. Laplace redux - effortless bayesian deep learning. In *Neural Information Processing Systems*, 2021. URL https://api.semanticscholar.org/CorpusID:235658031.

Michael Girstl, Alexander Mattick, and Christopher Mutschler. Constrained reinforcement learning using successor representations. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=6zUq7knzwA.

Matteo Hessel, Hubert Soyer, Lasse Espeholt, Wojciech M. Czarnecki, Simon Schmitt, and H. V. Hasselt. Multi-task deep reinforcement learning with popart. *ArXiv*, abs/1809.04474, 2018. URL https://api.semanticscholar.org/CorpusID:52194333.

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. URL http://jmlr.org/papers/v25/23-0681.html.

Yongshuai Liu, J. Ding, and Xin Liu. Ipo: Interior-point policy optimization under constraints. In *AAAI Conference on Artificial Intelligence*, 2019. URL https://api.semanticscholar.org/CorpusID:204824159.

Zuxin Liu, Zhepeng Cen, Vladislav Isenbaev, Wei Liu, Zhiwei Steven Wu, Bo Li, and Ding Zhao. Constrained variational policy optimization for safe reinforcement learning. *ArXiv*, abs/2201.11927, 2022. URL https://api.semanticscholar.org/CorpusID:246411597.

Alexander Mattick, Dominik Seuß, and Christopher Mutschler. SafeMPO: Constrained reinforcement learning with probabilistic incremental improvement. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=1m0EU6QXj6.

Rémi Munos, Tom Stepleton, Anna Harutyunyan, and Marc G. Bellemare. Safe and efficient off-policy reinforcement learning. In *Neural Information Processing Systems*, 2016. URL https://api.semanticscholar.org/CorpusID:492962.

Karthik Narasimhan. Projection-based constrained policy optimization. *ArXiv*, abs/2010.03152, 2020. URL https://api.semanticscholar.org/CorpusID:210903109.

Radford M. Neal and Geoffrey E. Hinton. *A View of the Em Algorithm that Justifies Incremental, Sparse, and other Variants*, pages 355–368. Springer Netherlands, Dordrecht, 1998. ISBN 978-94-011-5014-9. doi: 10.1007/978-94-011-5014-9_12. URL https://doi.org/10.1007/978-94-011-5014-9_12.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep Exploration via Bootstrapped DQN.

Morton L. Slater. Lagrange multipliers revisited. 2014. URL https://api.semanticscholar.org/CorpusID:118523562.

H. Francis Song, Abbas Abdolmaleki, Jost Tobias Springenberg, Aidan Clark, Hubert Soyer, Jack W. Rae, Seb Noury, Arun Ahuja, Siqi Liu, Dhruva Tirumala, Nicolas Manfred Otto Heess, Dan Belov, Martin A. Riedmiller, and Matthew M. Botvinick. V-mpo: On-policy maximum a posteriori policy optimization for discrete and continuous control. *ArXiv*, abs/1909.12238, 2019. URL https://api.semanticscholar.org/CorpusID:202889322.

Adam Stooke, Joshua Achiam, and P. Abbeel. Responsive safety in reinforcement learning by pid lagrangian methods. In *International Conference on Machine Learning*, 2020. URL https://api.semanticscholar.org/CorpusID:220404280.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA, 2018. ISBN 0262039249.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy P. Lillicrap, and Martin A. Riedmiller. Deepmind control suite. *ArXiv*, abs/1801.00690, 2018. URL https://api.semanticscholar.org/CorpusID:6315299.

Chen Tessler, Daniel Jaymin Mankowitz, and Shie Mannor. Reward constrained policy optimization. *ArXiv*, abs/1805.11074, 2018. URL https://api.semanticscholar.org/CorpusID:44095973.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Veit Wild, Sahra Ghalebikesabi, D. Sejdinovic, and Jeremias Knoblauch. A rigorous link between deep ensembles and (variational) bayesian methods. *ArXiv*, abs/2305.15027, 2023. URL https://api.semanticscholar.org/CorpusID:258865414.

Long Yang, Jiaming Ji, Juntao Dai, Linrui Zhang, Binbin Zhou, Pengfei Li, Yaodong Yang, and Gang Pan. Constrained update projection approach to safe policy optimization. *ArXiv*, abs/2209.07089, 2022. URL https://api.semanticscholar.org/CorpusID:252280735.

Yiming Zhang, Quan Ho Vuong, and Keith W. Ross. First order optimization in policy space for constrained deep reinforcement learning. *ArXiv*, abs/2002.06506, 2020. URL https://api.semanticscholar.org/CorpusID:211132768.

A PSEUDOCODE

We summarize the final algorithm in Algorithm 1. For our C- and Q-critics we utilize RETRACE (Munos et al., 2016) and PopArt (Hessel et al., 2018) to stabilize the targets and remove hyperparameters. We found that the overall solving speed of Algorithm 1 is roughly on-par with the algorithm proposed by (Mattick et al., 2026), since while an additional step is needed for solving eq. (7), the final problem eq. (8) is much better conditioned (due to a lack of log-barrier) and is therefore solved in much fewer steps. In fact, even compared to other CMDP algorithms our method does not lose in terms of wall-clock time due to the ability of off-policy algorithms to asynchronously roll out and update the policy (though this of course depends on the exact rollout to training ratio).

Algorithm 1: A single step of ILB-CRL

Input: π, $\mathcal{D}$, batch size N, action samples K, constraint ε, EMA coefficient φ
Output: π
Asynchronously collect $\{(s_t, a_t, r_t, c_t, s_{t+1})\}_{t=0,t=T} \sim \pi$ and add to $\mathcal{D}$
for $t = 1$ **to** N **do**
  $\{s_i, a_i, r_i, c_i\}_{i=1}^{N} \sim \mathcal{D}$
  Evaluate ensemble $C(s_i, a_{i,k})$ with $a_{i,k} \sim \pi_{\text{old}}(\cdot|s_i) \quad \forall i, k$
  // E-step
  $g \leftarrow$ solve eq. (7). LCB $\leftarrow$ eq. (9), $\alpha \leftarrow$ eq. (10)
  $q \leftarrow$ optimize eq. (11)
  // M-step via ADAM
  $\pi \leftarrow \arg\min_\pi KL(q\|\pi)$ s.t. $KL(\pi_{\text{old}}\|\pi) \leq \varepsilon$
  // Update Critics
  $Q \leftarrow$RETRACE(s_i, a_i, r_i), $C \leftarrow$RETRACE(s_i, a_i, c_i)
end

B PROOFS

Proof for Theorem 1. For any $\varepsilon > 0$ since N is fixed, the union bound yields

$$P(\max_{i \leq N} \|G(\cdot|\theta_i) - G(\cdot|\theta^\star)\|_\infty > \varepsilon | D_m) \leq NP(\|G_\theta - G_{\theta^\star}\|_\infty > \varepsilon | D_m) \xrightarrow{m \to \infty} 0$$

Therefore, for any N all posterior samples concentrate uniformly around $G(\cdot|\theta^\star)$.

Now let

$$e_m := \max_{1 \leq i \leq N} \|G(\cdot|\theta_i) - G(\cdot|\theta^\star)\|_\infty$$

the posterior consistency with the union bound gives

$$e_m \xrightarrow{P} 0$$

where $\xrightarrow{P}$ denotes convergence in probability. For any fixed event (a, s) and every ensemble member i we have

$$|G(a, s|\theta_i) - G(a, s|\theta^\star)| \leq e_m$$

Now let

$$\hat{\mu}_m = \frac{1}{N} \sum_i^N G(a, s|\theta_i)$$

be the empirical mean. By triangle inequality

$$|\hat{\mu}_m - G^\star| = \left| \frac{1}{N} \sum_i^N G(a, s|\theta_i) - G^\star \right| \tag{17}$$

$$= \left| \frac{1}{N} \sum_i^N (G(a, s|\theta_i) - G^\star) \right| \tag{18}$$

$$\leq \frac{1}{N} \sum_i^N |G(a, s|\theta_i) - G^\star| \tag{19}$$

$$\leq \frac{1}{N} \sum_i^N e_m \tag{20}$$

$$= e_m \tag{21}$$

so via triangle inequality

$$|G(a, s|\theta_i) - \hat{\mu}_m| \leq |G(a, s|\theta_i) - G^\star(a, s|\theta^\star)| + |\hat{\mu}_m - G^\star(a, s|\theta^\star)| \leq 2e_m$$

The empirical variance $\hat{V}_m$ is therefore

$$\frac{1}{N-1} \sum_i^N (G(a, s|\theta_i) - \hat{\mu}_m)^2 \leq C_N e_m^2 \xrightarrow{P} 0$$

where C_N is a fixed constant depending on N.

Since

$$V_m \to 0$$

we have

$$Unc_m := g(\hat{V}, c, N, \delta) \xrightarrow{P} 0,$$

Because $\hat{\Delta}_m \xrightarrow{P} \Delta > 0$ as per (Mattick et al., 2026).

$$\alpha_m = 1 - \frac{Unc_m}{\hat{\Delta}_m} \xrightarrow{P} 1$$

Therefore we get

$$P(\alpha_m > \underline{\alpha}) \xrightarrow{P} 1$$

which in particular implies that α_m is bounded away from zero in probability for sufficiently large m. □

Remark 4. *Abstractly, what this theorem states is that if the epistemic distribution contracts around the true safety function as data accumulates, then any finite ensemble drawn from that distribution will also contract and therefore the "uncertainty penalty" $1 - \alpha$ will vanish and the algorithm seizes to suppress the full improvement.*

Lemma 1. *Let r be the feasibility residual implied by (Mattick et al., 2026)'s* log*-barrier function*

$$r(q_{SafeMPO}) \leq \rho r(\pi)$$

for some contraction rate $0 < \rho < 1$. Since $q^\star$ is the fastest contraction in the KL-ball, we have

$$r(q^\star) \leq r(q_{SafeMPO}) \leq \rho r(\pi)$$

Since we have an implied stepsize of $\alpha \geq \underline{\alpha} > 0$ we contract with at least a global rate of

$$r(q_{ILB}) \leq \underbrace{(1 - \underline{\alpha}(1 - \rho))}_{<1} r(\pi)$$

Proof for Corollary 1. Suppose the residual satisfies a Lemma 1 type $0 < \rho < 1$ such that

$$r_{m+1} \leq (1 - \alpha_m(1-\rho))r_m$$

Pick some $\underline{\alpha} > 0$ and let

$$c = 1 - \underline{\alpha}(1-\rho) < 1$$

be the minimal gain. We call an iteration "bad" if $\alpha_m < \underline{\alpha}$ and define the indicator RV

$$I_m = 1\{\alpha_m < \underline{\alpha}\}$$

since

$$P(I_m = 1) \to 1$$

so the fraction of bad steps converges to zero in probability.

$$\frac{1}{M}\sum_{m=1}^{M} I_m \xrightarrow{P} 0$$

Any good iteration contracts by at least c, a bad iteration still has $\alpha_m \geq 0$, hence

$$r_M \leq r_0 c^{M - B_M}$$

where $B_m = \sum_m I_m$. If we further note $B_M/M \xrightarrow{P} 0$ we get

$$r_M \lesssim r_0 c^{(1-o_p(1))M}$$

□

Remark 5. *Note that conditioning the LCB to only consider positive improvements is always at least as tight as the standard one, assuming (non strict) monotonicity in σ (i.e. as the variance drops so does $g(\hat{V}, c, N, \delta)$)*

Proof. Let F be the CDF of the untruncated gain. The CDF conditioned on $X \geq 0$ is

$$F_+(x) = \frac{F(x) - F(0)}{1 - F(0)}$$

its lower quantile is $F_+(q_\delta^+) = \delta$. Hence

$$F(q_\delta^+) = F(0) + \delta(1 - F(0)),$$

thus

$$q_\delta^+ = F^{-1}(F(0) + \delta(1 - F(0)))$$

monotonicity of CDFs yields $q_\delta^+ \geq q_\delta$. It is strict when $F(0) > 0$ and the CDF is strictly increasing □

Proof for Theorem 2. First note, that the CDF of $\mathcal{TN}(\mu, \sigma^2, 0, \infty)$ with $\mu = \hat{\Delta}$, $\sigma = \hat{\sigma}_\Delta/\sqrt{N}$ is

$$F(x) = \frac{\Phi\left(\frac{x-\hat{\Delta}}{\sigma}\right) - \Phi\left(\frac{-\hat{\Delta}}{\sigma}\right)}{\Phi\left(\frac{\hat{\Delta}}{\sigma}\right)}, \quad x \geq 0. \tag{22}$$

Also note that per (Mattick et al., 2026) we know that $\hat{\Delta} > 0$ for all not already feasible policies.

Statement 1 follows from $P(\Delta_{\text{true}} \geq \alpha\hat{\Delta}|data, \Delta_{\text{true}} \geq 0) \geq 1 - \delta$ is equivalent to $F(\alpha\hat{\Delta}) \leq \delta$ by definition of $\Delta_{\text{true}}|data, \Delta_{\text{true}} \geq 0$. Let

$$\Delta_{\text{LCB}}^+ = F^{-1}(\delta)$$

note that since $\sigma > 0$ F is continuous and strictly increasing on $[0, \infty)$.

For $\alpha \in [0, 1]$ we have

$$P(\Delta_{\text{true}} \geq \alpha\hat{\Delta}|\text{data}, \Delta_{\text{true}} \geq 0) \geq 1 - \delta \tag{23}$$

$$\iff F(\alpha\hat{\Delta}) \leq \delta \tag{24}$$

$$\iff \alpha\hat{\Delta} \leq \Delta^{+}_{\text{LCB}} \tag{25}$$

$$\iff \alpha \leq \frac{\Delta^{+}_{\text{LCB}}}{\hat{\Delta}}. \tag{26}$$

Thus

$$\alpha^{\star} = \min\left\{1, \frac{\Delta^{+}_{\text{LCB}}}{\hat{\Delta}}\right\}$$

This resolves Statement 1.

For completeness, we also state the quantile explicitly

$$\Phi\left(\frac{\alpha^{\star}\hat{\Delta} - \hat{\Delta}}{\sigma}\right) = \delta\,\Phi\left(\frac{\hat{\Delta}}{\sigma}\right) + \Phi\left(\frac{-\hat{\Delta}}{\sigma}\right) \tag{27}$$

$$= (1 - \delta)\Phi\left(\frac{-\hat{\Delta}}{\sigma}\right) + \delta \tag{28}$$

where we used the integral-limit identity $\Phi(x) + \Phi(-x) = 1$. Solving for $\alpha^{\star}\Delta$

$$\alpha^{\star}\hat{\Delta} = \hat{\Delta} + \sigma\Phi^{-1}\left((1 - \delta)\Phi\left(\frac{-\hat{\Delta}}{\sigma}\right) + \delta\right) = \Delta^{+}_{\text{LCB}}. \tag{29}$$

This can be directly computed in closed form.

Statement 2 follows from $\Delta^{+}_{\text{LCB}} > 0$. Since $\delta \in (0, 1)$ we know that

$$(1 - \delta)\Phi(-\hat{\Delta}/\sigma) + \delta > \Phi(-\hat{\Delta}/\sigma)$$

so $\Delta^{+}_{\text{LCB}} > \hat{\Delta} + \sigma(-\hat{\Delta}/\sigma) = 0$. Therefore $\alpha^{\star} > 0$ by Statement 1.

Finally, **Statement 3** follows directly from Statement 1: Constraint (8)c is feasible iff there exists q with $KL(q\|\pi) \leq \varepsilon$ achieving the required gain. The maximum achievable gain within the KL ball is $\Delta_{\text{true}} = G^{\star}_{\text{true}} - \mathbb{E}_{\pi}[G]$. Therefore feasibility requires $\Delta_{\text{true}} \geq \alpha^{\star}\hat{\Delta}$, which holds with probability $1 - \delta$ by construction of $\alpha^{\star}$. □

C Architecture and implementation notes

To guarantee boundedness for infinite horizon problems one typically has an exponential falloff coefficient γ which scales down distant rewards (see also eq. (1)). Typically, benchmarks in CRL (such as the ones in (Ji et al., 2024)) use undiscounted costs and finite length horizon (H) environments to account for this. In this work, to keep the implementation general, we use a $\gamma_{\text{cost}} = 0.999$ and then employ a standard horizon correction term $\tilde{B}_i = B_i \cdot \frac{(1-\gamma^{H}_{\text{cost}})}{H(1-\gamma_{\text{cost}})}$ on the budget. This modification was already done in (Liu et al., 2022) to account for the technically infinite time horizon in MDPs. Increasing the γ_{cost} beyond that leads to a tighter horizon-corrected bound, but ends up being hard to adequately cover for the cost-critic (specifically large outlier spikes become a problem for $\gamma_{\text{cost}} = 1$). Improving this might be a very interesting direction for future work as e.g., safe DQN methods also benefit from this (e.g., (Girstl et al., 2026)). Another potential avenue for improvement in estimator quality would be the usage of distributional critics as this might also reduce the outlier problem.

D Student-T based results

Table 5 showcases the results after 2 Million steps of environment interactions. We perform the evaluation described in (Agarwal et al., 2021) using the last 100 episodes of the 2 Mio steps at a

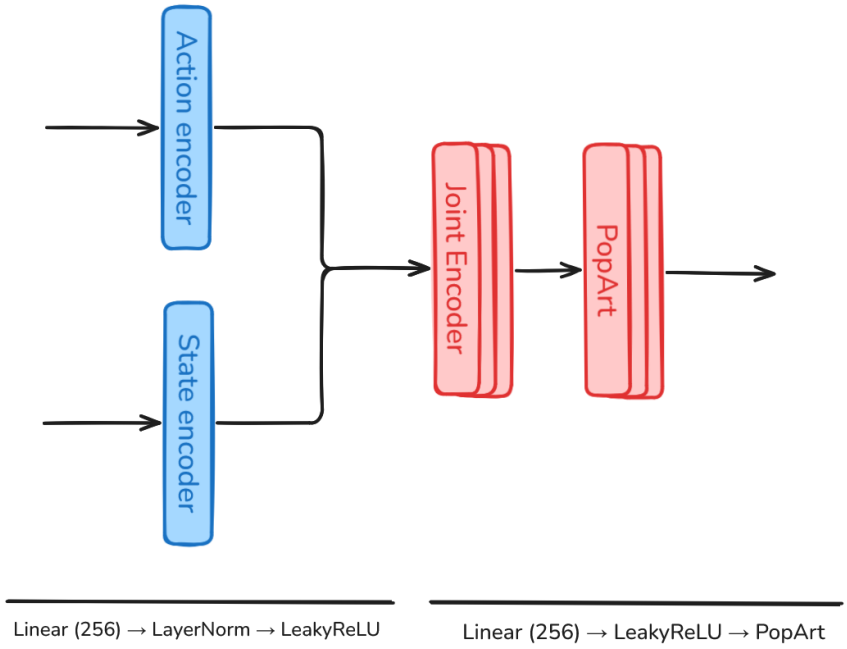


Figure 3: Overview of the ILB-CRL architecture. We use a single layer "trunk" for both the action and state embedding, and a 2 layer ensemble consisting out of a linear transformation and the PopArt layers. This mirrors the design of many modern Deep Ensemble architectures where a common (sometimes frozen) backbone provides embeddings for the shallow ensemble. We use 8 ensemble members but have not found a strong dependence on ensemble size.

cost budget of 25. Notice that even after only 2 million steps we can observe high performance while (approximately) satisfying the constraints in almost all of them. The main exception to this is "SafetyCarButton1" which is known to be unsolved for model-free algorithms.

We observe a general limitation of the student-t based approach where small violations cannot be adequately removed since the associated LCB drops below zero.

Table 3: IQM with 95% Stratified Bootstrap CI after 2 Mio steps.

Environment	Cost (IQM [95% CI])	Return (IQM [95% CI])
SafetyAntGoal1-v0	18.77 [15.06, 20.70]	9.99 [8.07, 11.95]
SafetyCarButton1-v0	58.19 [53.57, 65.20]	5.08 [4.41, 5.64]
SafetyCarGoal1-v0	27.92 [26.05, 28.36]	25.63 [24.63, 26.48]
SafetyCarPush1-v0	22.48 [20.29, 25.56]	7.51 [7.05, 7.87]
SafetyHalfCheetahVelocity-v1	17.69 [17.24, 18.93]	2144.41 [2100.13, 2168.53]
SafetyPointGoal1-v0	28.27 [26.06, 29.48]	19.45 [16.86, 20.92]

E Extending to multiple constraints

While no standard benchmark tests the ability of CRL algorithms to handle multiple constraints, our method does trivially allow for multiple constraints to be solved simultaneously by simply adapting the maximal improvement subproblem. Instead of solving against a single target eq. (7), we solve

against multiple at the same time with a "no descent" constraint for all individual constraints:

$$G^\star = \max_{q:\mathcal{P}(A\times S)} \mathbb{E}_{\tau\sim q}\left[\sum_i^K G_i(a,s)\right] \tag{30a}$$

$$\text{s.t. } KL(q\|\pi) \le \varepsilon \tag{30b}$$

$$\mathbb{E}_{\tau\sim q}\left[G_i(a,s)\right] \ge \mathbb{E}_{\tau\sim\pi}\left[G_i(a,s)\right]_+ \tag{30c}$$

This gives a trivial dual problem

$$\min_{\eta,\lambda_i\ge 0} \eta\varepsilon - \sum_i^K \lambda_i \bar{G}_i + \eta \log \int \exp\left(\frac{\sum_i^K (1+\lambda_i) G_i(a,s)}{\eta}\right) \tag{31}$$

In practice we found that this can be solved very quickly by first determining the global temperature η first as a warm start and then solving the dual in λ, η jointly. For a reasonable number of constraints this solves quite quickly with e.g. Newton's method. The resulting $G_i^\star$ are now by design jointly feasible and can be used inside

$$\max_{q:\mathcal{P}(A\times S)} \mathbb{E}_{(a,s)\sim q}[Q(a,s)] \tag{32a}$$

$$\text{s.t. } KL(q\|\pi) \le \varepsilon \tag{32b}$$

$$\mathbb{E}_{(a,s)\sim q}[G_i(a,s)] \ge \mathbb{E}_{(a,s)\sim\pi}[G_i(a,s)] + \alpha_i(G^\star_{i,+} - \mathbb{E}_{(a,s)\sim\pi}[G_i(a,s)]) \tag{32c}$$

with α_i determined independently for all $G_i^\star$. Since α_i are determined independently, the general convergence theorems theorem 1 and theorem 2 apply as before.

Note that without further assumptions on the dependence of RVs we can only make very coarse union-bound guarantees on the probability of failure: If each constraint separately has $P(E_i) \ge 1-\delta_i$ the combined failure without dependence assumptions is

$$P\left(\bigcap_i E_i\right) \ge 1 - \sum_i \delta_i$$

to get the same guarantees as in the single variable case, one can apply the classical bonferroni correction $\delta_i = \delta/N$ but this is usually very conservative.

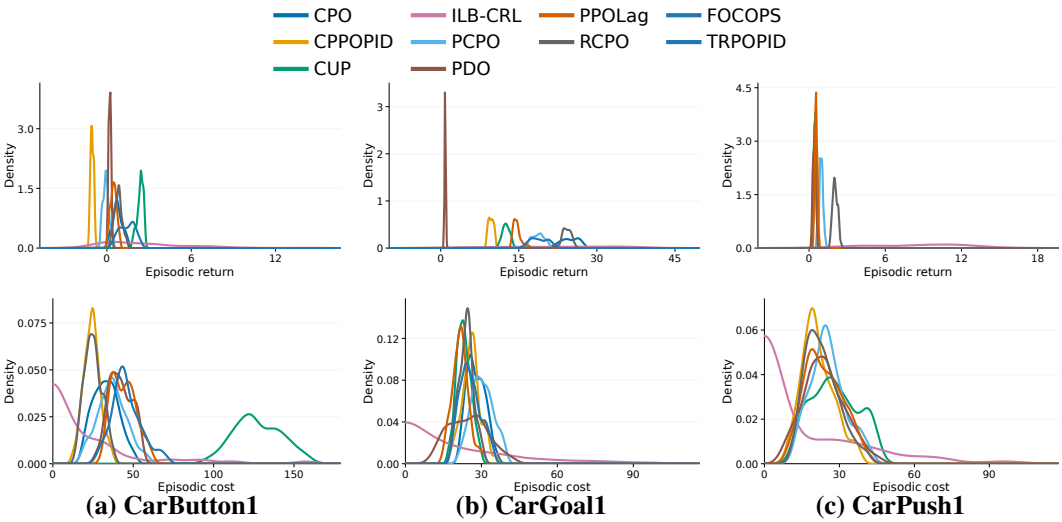
In practice, there should be strong dependence of the different constraints simply because they are tied to the same environment dynamics. In practice one can probably assume that the set of constraints comes from a multivariate truncated normal, which would then give you chance constraints analogous to theorem 2 ($P(\forall i \Delta_i \ge \alpha_i \hat{\Delta}_i) \ge 1-\delta$, which is valid without bonferroni)

F HISTOGRAMS

We showcase the histograms in fig. 4. The histograms are computed on the last 100 trajectories for each policy across multiple seeds and the aggregated using a Wasserstein median. Confidence intervals are omitted to reduce clutter in the visualization.

Different algorithms tend to produce different output distribution shapes even when ignoring the specific performance numbers. ILB-CRL tends to produce much flatter histograms, effectively paying for large "accident" episodes with many safe ones. Comparing this to e.g. CPPOPID, or PDO, which has peaks right around $B = 25$ with few very safe or very costly episodes. This comparison should be seen without judgement: The CMDP constraint only assumes the expected value is below the threshold, not necessarily the shape of the density. Depending on usecase having a wider or narrower cost band may be beneficial: In Robotics one might want fewer outliers, but when e.g., constraining the number of reasoning tokens in an LLM, you would want extreme thinking for complex answers and instant answers for simple ones.

However, we do believe it is interesting that our off-policy method seemingly is the only one opting for the "fat tail" style distribution: We hypothesize this might be the case since batches are much more heterogenous in off-policy learning compared to on-policy. Since off-policy learning makes use of much more information, such long tailed policies are possible (and indeed superior on the



(a) CarButton1 **(b) CarGoal1** **(c) CarPush1**

Figure 4: Histograms of Cost and returns after training. Reward is shown above cost. Methods generally perform differently even outside their absolute cost and rewards. For instance ILB-CRL tends to produce much wider distributions compared to e.g. CPPOPID.

Table 4: Comparison of IQM and raw arithmetic mean performance across seeds. Although we report IQM in the main results following (Agarwal et al., 2021), the corresponding arithmetic means are very similar.

Environment	Return			Cost		
	IQM	Mean	std	IQM	Mean	Std
SafetyCarButton1	3.26	3.34	0.40	25.18	25.23	2.55
SafetyCarGoal1	22.32	21.82	3.08	23.76	24.88	3.20
SafetyCarPush1	8.42	8.26	0.89	18.76	18.53	2.15
SafetyPointGoal1	17.35	17.46	1.86	24.65	25.22	2.90

environments tested), but hard to discover if one only sees the current e.g. 10 policies in an on-policy setting. Off-policy allows the methods to consume millions of transitions, meaning that the long-tail trajectories can be modeled, and therefore taken advantage of, much more easily.

While this does not give any immediately qualitative or quantitative guidance for algorithm development, we do think it was interesting to include, even if only to explicitly note the "Character" of the density, which often goes under-looked in CRL. Controlling for the exact shape of the density is, as in unconstrained RL, a question of replacing expected-value targets with distribution aware variants like CVaR or percentile risk. For these types of risk, off-policy might even be better due to its ability to accurately measure the tails of the cost density.

G Raw Mean and Standard Deviation

H Fixed α Results

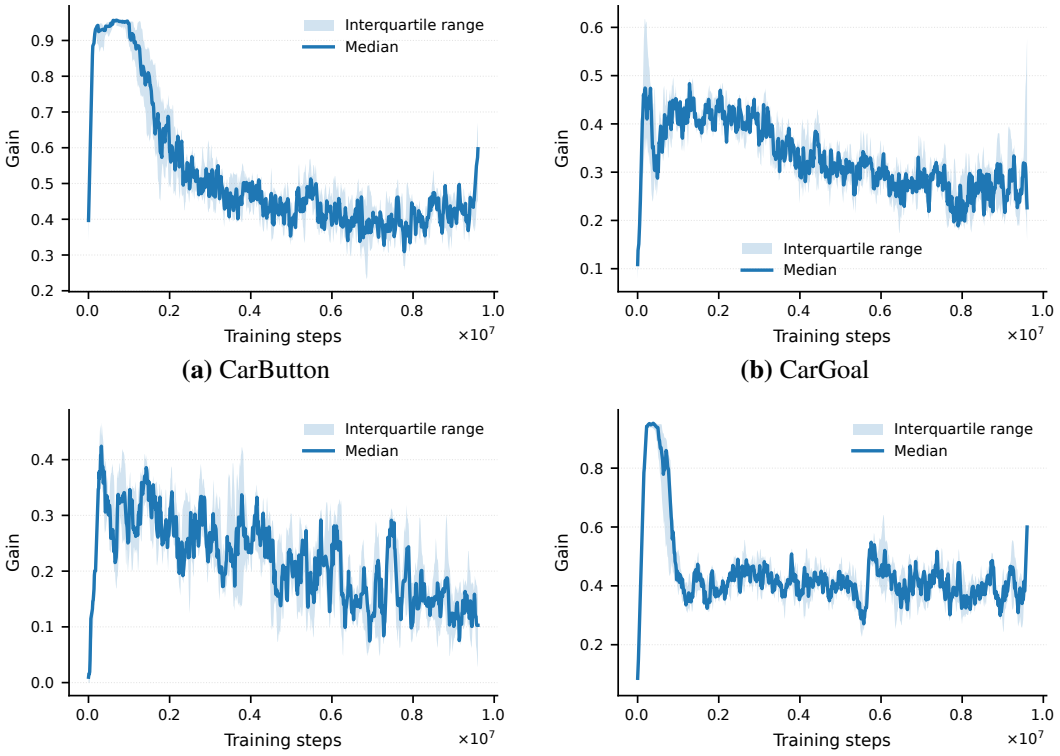
Fixing α generally harms the policy significantly, specifically in complex scenarios where costs are usually high (e.g. CarButton1): If α is set to a full step $\alpha = 1$ we obtain an overly cautious model which has low costs but barely tries to look for rewards. This holds up for all environments aside from CarPush, which is generally safe even for small α and therefore ends up improving by a small amount (though the CIs end up overlapping a lot).

Having an automatic estimation of α means ILB-CRL can select a more aggressive step for more complex environments and a smaller step for more open ones. This distinction arises automatically:

Algorithm	Metric	CarButton1	CarGoal1	CarPush1	PointGoal1
ILB-CRL@$\alpha = 1.0$	Return	$-2.3\,(-4.6, -0.9)$	$2.6\,(-4.7, 14.7)$	$7.7\,(5.3, 10.2)$	$0.0\,(-10.4, 18.5)$
	Cost	$1.4\,(0.0, 2.9)$	$5.1\,(0.0, 14.2)$	$15.2\,(8.3, 19.4)$	$8.1\,(0.0, 20.2)$
ILB-CRL@$\alpha = 0.5$	Return	$6.0\,(4.2, 8.2)$	$18.6\,(15.2, 21.7)$	$9.5\,(9.2, 9.8)$	$15.5\,(14.0, 18.4)$
	Cost	$48.9\,(40.2, 57.8)$	$18.0\,(14.2, 20.5)$	$15.3\,(13.3, 18.8)$	$23.9\,(19.0, 33.1)$
ILB-CRL@$\alpha = 0.25$	Return	$6.1\,(4.8, 7.4)$	$18.3\,(15.4, 20.3)$	$9.1\,(8.0, 10.0)$	$17.6\,(15.3, 19.9)$
	Cost	$38.9\,(31.3, 45.8)$	$18.2\,(17.2, 19.8)$	$18.0\,(12.6, 25.5)$	$22.3\,(19.0, 27.4)$

Table 5: Return and cost (IQM) with 95% confidence intervals for each environment with fixed α rather than the adaptive truncated normal LCB (here over $N = 5$ environments rather than $N = 10$). We find that using a fixed rather than dynamic α completly breaks the method with the exception of CarPush which ends up better, mostly due to "hero runs" which end up increasing the upper bound.

If a model stops making progress on the cost, the estimator will automatically become better, which increases α until a new regime is discovered, which drops α potentially making way for bigger reward steps.



(a) CarButton **(b)** CarGoal

(c) CarPush **(d)** PointGoal

Figure 5: Evolution of the α-gain over training. The solid line shows the median across seeds and the shaded region denotes the interquartile range. Curves are smoothed using a moving-average window of 300 logged updates.

We further highlight this in Figure 5 where we plot the α over the course of training. Notice that the found α changes substantially during training: α usually starts off at zero (since the critic is randomly initialized) before jumping up to relatively high values compared to the rest of training. Qualitatively, this corresponds to "easy wins" in safety such as not directly driving across banned regions of the state space. Once those easy wins are solved, the certainty generally drops since the remaining cost is hidden in e.g. precise knowledge of the size of the illegal regions or knowledge on the typical paths of moving obstacles.