Constrained Offline Policy Optimization

Nicholas Polosky [1] **Bruno C. da Silva** [2] **Madalina Fiterau** [2] **Jithin Jagannath** [1]

Abstract

In this work we introduce Constrained Offline Policy Optimization (COPO), an offline policy optimization algorithm for learning in MDPs with cost constraints. COPO is built upon a novel offline cost-projection method, which we formally derive and analyze. Our method improves upon the state-of-the-art in offline constrained policy optimization by explicitly accounting for distributional shift and by offering non-asymptotic confidence bounds on the cost of a policy. These formal properties are superior to those of existing techniques, which only guarantee convergence to a point estimate. We formally analyze our method and empirically demonstrate that it achieves state-of-the-art performance on discrete and continuous control problems, while offering the aforementioned improved, stronger, and more robust theoretical guarantees.

1. Introduction

This work addresses the problem of identifying safe policies in reinforcement learning based on existing static data sets of previously-collected experiences. In particular, we tackle the problem of providing robust guarantees of the performance of policies optimized under the Constrained Markov Decision Process (CMDP) (Altman, 1999) framework, given a finite amount of offline data. We improve upon the existing literature by designing the first algorithm that offers non-asymptotic confidence bounds on the true cost of a policy and that achieves state-of-the-art performance on a variety of control problems.

Our work combines ideas from offline reinforcement learning (RL) and from constrained Markov Decision Processes. Offline (or batch) RL (Levine et al., 2020; Sascha Lange, 2012) is concerned with estimating the value of a policy, or directly learning a policy, from a static data set. This is relevant whenever one needs to evaluate novel candidate policies without directly deploying them, which might be costly or risky. Furthermore, many real-world applications of RL require the eventual safe operation of an agent. Safety is often modeled in the CMDP framework, where constraint functions define behaviors an agent should avoid.

Techniques that operate at the intersection of these areas, i.e., that identify cost-safe policies given offline data, have been studied. The Batch Policy Learning under Constraints (BPLC) (Le et al., 2019) algorithm, for example, is currently considered the state-of-the-art in this field. It employs ensemble policies and Fitted-Q methods to learn offline constrained policies that are empirically shown to satisfy a permissible cost budget once deployed. BPLC, however, is limited in two ways: *(i)* it only provides high probability guarantees that a *point estimate* of the policy cost will converge to a point below a pre-specified budget; importantly, however, *(ii)* this estimate is based on a (possibly small) single set of experiences, which may not be sufficient to fully characterize the stochasticity of the MDP and the true distribution underlying the data generating policy. As a result, the safety guarantees provided by BPLC are not necessarily robust to variability inherent to finite data.

In this work, we improve upon the state-of-the-art by developing a new constrained offline policy optimization (COPO) algorithm capable of producing *high probability confidence bounds on the true cost value of the policy*. These confidence bounds provide a more robust way of estimating the true cost of a policy in cases where the finite amount of training data may not be sufficient to properly characterize the process from which samples where generated. This type of robust guarantee on the cost of a policy is paramount in real-world scenarios where breaking a cost budget may carry extreme consequences, such as in medical applications of RL (Bastani, 2014; Saria, 2018).

In the next sections we *(i)* introduce the necessary mathematical background; *(ii)* formally derive our novel constrained projection technique; *(iii)* formally characterize the high probability confidence intervals that can be guaranteed by it; *(iv)* provide a finite sample analysis of our

[1] Marconi-Rosenblatt AI/ML Innovation Lab, ANDRO Computational Solutions, LLC, Rome, NY 13440 [2] University of Massachusetts at Amherst, Department of Computer Science, Amherst, MA 01003. Correspondence to: Nicholas Polosky <npolosky@androcs.com>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

method; and *(v)* demonstrate empirically that it achieves state-of-the-art performance on discrete and continuous control problems, while for the first time offering high confidence non-asymptotic confidence bounds on the true cost of a policy, given offline data.

2. RL via Linear Programming

We start by reviewing important concepts related to policy evaluation and optimization in RL via linear programming. Our method, COPO, will be built upon these ideas. We consider the problem of identifying an optimal policy for a given Markov Decision Process (MDP). An MDP is defined as the tuple $(\mathcal{S}, \mathcal{A}, P : \mathcal{S} \times \mathcal{A} \to \mathcal{S}, R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}, \gamma \in (0, 1], \mu : \mathcal{S} \to [0, 1])$ representing the state space, action space, transition function, reward function, discount factor, and initial state distribution, respectively. In offline RL, we assume that we have been provided with a static data set (collected via an arbitrary set of policies) with which we aim to learn an optimal policy. This data set of N transition tuples, $\mathcal{D} = \{(s_i, a_i, r_i, s_{i+1})\}_{i=0}^{N}$, is assumed to be generated by the interaction of some unknown number of unknown policies with an MDP. In this work we examine the undiscounted cost setting. Throughout the rest of the paper, then, $\gamma = 1$ is assumed unless stated otherwise. In general, identifying an optimal policy often involves iterative learning procedures that require estimating the value of a policy. When operating in an undiscounted, infinite-horizon setting, the value of a policy π, $\rho(\pi)$ is defined as the average per-step reward:

$$\rho(\pi) := \lim_{t_{stop}\to\infty} \mathbb{E}\left[\frac{1}{t_{stop}} \sum_{t=0}^{t_{stop}} R(s_t, a_t) \Big| \; s_0 \sim \mu, \text{and } a_t \sim \pi(s_t), s_{t+1} \sim P(s_t, a_t) \; \forall t \right]$$

When estimating the above quantity using samples collected from a policy different than π, the value estimation problem is called the off-policy evaluation (OPE) problem. The OPE problem can, alternatively, be modeled using a linear programming (LP) representation. For the undiscounted case, the primal form of the OPE problem, often referred to as the Q-LP, and its associated dual, d-LP, are presented below in the upper and lower equations, respectively:

$$\begin{aligned} \min_{\substack{\lambda\in\mathbb{R},\\ Q:\mathcal{S}\times\mathcal{A}\to\mathbb{R}}} \quad & \lambda \\ \text{subject to:} \quad & Q(s,a) \geq R(s,a) + \mathcal{P}^{\pi} Q(s,a) - \lambda \end{aligned}$$

$$\begin{aligned} \max_{d:\mathcal{S}\times\mathcal{A}\to\mathbb{R}_+} \quad & \mathbb{E}_{(s,a)\sim d}\left[R(s,a)\right] \\ \text{subject to:} \quad & d(s,a) = \mathcal{P}_*^{\pi} d(s,a) \\ & \sum_{s\in\mathcal{S}, a\in\mathcal{A}} d(s,a) = 1 \end{aligned}$$

where λ is normalizing variable, $d(s, a)$ is the normalized state-action visitation density, $\mathcal{P}^{\pi}$ is the transition operator under policy π, and $\mathcal{P}_*^{\pi} d(s, a) := \pi(a|s) \sum_{\tilde{s},\tilde{a}} P(s|\tilde{s}, \tilde{a}) d(\tilde{s}, \tilde{a})$ is the adjoint transition operator. The solution to the Q-LP is Q^{π}, the action-value function for policy π. Analogously, the solution to the d-LP is d^{π}, the normalized state-action visitation density under policy π. The linear programs above exhibit strong duality and thus share the same objective values at their optimums, which can be shown to be equal to $\rho(\pi)$, as previously defined.

Note that in the constrained RL setting the environment is represented using a CMDP, which augments the traditional MDP formulation with a cost function $C : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ and a cost budget $\beta \in \mathbb{R}$. Accordingly, to characterize the average per-step cost one needs only to replace the reward function R with the cost function C in the linear programs previously defined.

3. Constrained Offline Policy Optimization

Our method, COPO, is applicable to any constrained RL problems in the offline setting. Before introducing the technical contributions underlying our technique, we describe it at a high level. COPO starts by finding a reward-optimal policy and subsequently projecting it onto the feasible set of policies that satisfy the cost constraints. This latter step is performed by a novel offline projection step that we introduce here. Our new offline projection technique takes an arbitrary policy as input, as well as an offline data set, and identifies the nearest policy (with respect to a metric supplied by a designer) that satisfies all cost budget constraints. Our projection method is derived by first constructing a policy objective comprised of a distance loss and a cost off-policy evaluation (OPE) component, and then transforming the objective via Fenchel duality (Boyd & Vandenberghe, 2004), thereby producing the final composed optimization problem.

The novelty of our method may be split into three separate contributions. The first contribution is a novel constrained policy projection technique in which a state-action visitation density is provided as input, and which returns a *cost-feasible policy* with visitation nearest to the one provided.[1] The second contribution is the capability of producing high-confidence bounds on the cost-value of the returned policy. The last contribution is a complete algorithm, COPO, which (given a batch of offline data) identifies a constraint-feasible policy that is optimized w.r.t. a given reward function.

3.1. Constrained Projection

In this section, we focus on the first challenge involved in identifying optimal policies that satisfy cost constraints. In particular, we focus on problem of finding a constraint-feasible policy whose visitation density is at minimal distance from a reference visitation density. Suppose we are provided with reference state-action visitation density d^R. Our objective, then, is:

$$\begin{aligned} \min_{\pi} \quad & \alpha D(d^\pi, d^R) \\ \text{subject to:} \quad & \rho_C(\pi) \leq \beta \end{aligned}$$

where $\rho_C(\pi)$ is the average per-step cost of policy π (defined similarly to the average per-step reward, as introduced in Section 2), $\alpha > 0$ is a scaling parameter, d^π is the normalized state-action visitation density under policy π, and D is a metric or pseudo-metric chosen by a designer. Examples of D include the family of f-divergences or the Wasserstein metric. To solve the above problem, we begin by writing the Lagrangian and expanding the OPE problem using the d-LP, respectively:

$$\begin{aligned} \min_{\pi} \max_{\lambda \geq 0} \quad & \alpha D(d^\pi, d^R) + \lambda \rho_C(\pi) - \lambda\beta \\ \min_{\pi} \max_{\lambda \geq 0} \quad & \alpha D(d^\pi, d^R) \\ & + \min_{d} \sum_{(s,a)} d(s,a)(\lambda C(s,a)) - \lambda\beta \\ \text{subject to:} \quad & d(s,a) = P_*^\pi d(s,a) \\ & \sum d(s,a) = 1 \end{aligned}$$

We note, here, that the solution to the OPE problem above, d^*, is equal to the visitation density d^π of the policy π. We proceed by moving the distance penalty component of the objective, D, inside the OPE objective and then making a change of variables. Note that this does not affect the outer optimization problems nor their solutions. After performing such a change of variables (from d^π to d) and moving the sum-to-one equality constraint into the objective, we obtain:

$$\min_{\pi} \max_{\lambda \geq 0, \nu} \min_{d} \quad \alpha D(d, d^R) + \tag{1}$$

$$\sum_{(s,a)} d(s,a)(\lambda C(s,a) + \nu) - (\lambda\beta + \nu) \tag{2}$$

$$\text{subject to:} \quad d(s,a) = P_*^\pi d(s,a)$$

where ν is the Lagrange multiplier for the sum-to-one constraint. The above optimization model provides us with the general form of the constraint projection problem. Once a particular distance function D is provided, we can transform the inner optimization problem by setting:

$$f(d) = \alpha D(d, d^R) + \tag{3}$$

$$\sum_{(s,a)} d(s,a)(\lambda C(s,a) + \nu) - (\lambda\beta + \nu) \quad \text{and} \tag{4}$$

$$g(Ad) = \delta_{\{0\}}(Ad), \qquad A = I - P_*^\pi$$

where δ_0 is is the zero indicator function. Finally, we use the following Fenchel-Rockafeller duality identity (Rockafellar, 1970) from Nachum and Dai (2020). The primal optimization problem

$$\min_{x \in X} f(x) + g(Ax)$$

(for semi-continuous $f, g : X \to \mathbb{R}$ and linear operator A) yields the following dual problem:

$$\max_{y \in X_*} -f_*(A_* y) - g_*(y),$$

where A_* is the adjoint of A. Using this identity to transform f, g we obtain a final unconstrained saddle point problem. Below, we provide results associated with the inner optimization transformation for commonly-used distance functions over distributions such as the f-divergence and Wasserstein distance. The distance functions below should be substituted into Equation (4); the transformed objectives are written below the distance functions. We note that, for the policy optimization saddle-point objective, each of the examples should be wrapped in a $\min_\pi$ operation.

f-divergence distance:

$$D(d, d^R) = \mathbb{E}_{(s,a)\sim d^R}\left[f\left(\frac{d}{d^R}\right)\right]$$

f-divergence objective:

$$\begin{aligned} \max_{\substack{\lambda \geq 0, \nu, \\ Q_c : \mathcal{A} \times \mathcal{A} \to \mathbb{Z}}} \quad & -\alpha \mathop{\mathbb{E}}_{(s,a)\sim d^R} [\, f_*((P^\pi Q_c(s,a) - Q_c(s,a) \\ & - \lambda C(s,a) - \nu)/\alpha) \,] - \lambda\beta - \nu \end{aligned}$$

[1] Distances are with respect to a metric or pseudo-metric selected by a designer.

Wasserstein distance:

$$D(d,d^R)=\sup_{\|g\|_L\leq 1}\mathop{\mathbb{E}}_{(s,a)\sim d}[g(s,a)]-\mathop{\mathbb{E}}_{(s,a)\sim d^R}[g(s,a)]$$

Wasserstein objective:

$$\begin{aligned}&\max_{\substack{\lambda\geq 0,\nu,\\ Q_c:\mathcal{A}\times\mathcal{A}\to\mathbb{Z},\|g\|_L\leq 1}} -\mathop{\mathbb{E}}_{(s,a)\sim d^R}[g(s,a)]-\lambda\beta-\nu\\ &g(s,a)=P^\pi Q_c(s,a)-Q_c(s,a)-\lambda C(s,a)-\nu\end{aligned}$$

Wasserstein Entropy distance:

$$\begin{aligned}D(d,d^R)=\sup_{\|g\|_L\leq 1}&\mathop{\mathbb{E}}_{(s,a)\sim d}[g(s,a)+\log(d(s,a))]\\ &-\mathop{\mathbb{E}}_{(s,a)\sim d^R}[g(s,a)]\end{aligned}$$

Wasserstein Entropy objective:

$$\begin{aligned}\max_{\substack{\lambda\geq 0,\nu,\\ Q_c:\mathcal{A}\times\mathcal{A}\to\mathbb{Z},\|g\|_L\leq 1}} &-\sum_{(s,a)}\exp\left(x(s,a)-1\right)\\ &-\mathop{\mathbb{E}}_{(s,a)\sim d^R}[g(s,a)]-\lambda\beta-\nu\\ x(s,a)&:=P^\pi Q_c(s,a)-Q_c(s,a)\\ &-\lambda C(s,a)-\nu+g(s,a)\end{aligned}$$

In the above Q_c is the dual variable to d, $g(s,a)$ originates from the dual formulation of the Wasserstein distance, and $\|g\|_L\leq 1$ is a 1-Lipschitz constraint. In this subsection, we have shown how to express and solve the problem of identifying a constraint-feasible policy whose visitation density is at minimal distance w.r.t. a reference density. This results in a novel constrained projection step that we will exploit to construct our constrained offline policy optimization algorithm. In the next sections, we characterize the confidence intervals on the cost-value of the learned policy and present a finite-sample analysis. We then introduce our complete algorithm and evaluate its empirical performance on discrete and continuous control problems

3.2. Confidence Intervals

In real-world applications it is often important to have high confidence bounds on the performance of an algorithm. In this section we show that, because of our use of DICE estimation, it becomes possible to derive high confidence intervals on the true cost of the policy—bounds which hold even under finite amount of data. We derive these bounds by providing a proof that is structurally similar to the one introduced by Dai et al. (2020), but adapted to our cost projection problem. To achieve this goal, we update the derivation of the COPO projection step (Equation (2)) by substituting the OPE objective with the upper bound of a confidence set. We show that the solution to the resulting problem is an $(1-\alpha)$ upper confidence bound on the sum of the average cost incurred by the projected policy and its distance from the reward optimal policy. We now present the derivation of confidence intervals when operating under one possible distance metric—the Wasserstein distance. We begin by making a change of variables from the on-policy visitation density, $d(s,a)$, to the distribution correction ratios, $\tau(s,a)=\frac{d(s,a)}{d^R(s,a)}$. This allows us to rewrite the Lagrangian with expanded OPE constraints:

$$\begin{aligned}\min_\pi\max_{\lambda\geq 0}\max_{\|g\|_L\leq 1}\min_{\tau\geq 0}\quad &\alpha\mathop{\mathbb{E}}_{d^R}[\tau g]-\alpha\mathop{\mathbb{E}}_{d^R}[g]\\ &+\mathop{\mathbb{E}}_{d^R}[\tau(\lambda C(s,a))]-\lambda\beta\\ \text{subject to:}\quad &\mathop{\mathbb{E}}_{d^R}[\tau(s,a)-P_*^\pi\tau(s,a)]=0\\ &\mathop{\mathbb{E}}_{d^R}[\tau(s,a)]=1.\end{aligned}$$

Next we use the function space embedding technique from Dai et al. (2020) to obtain generalized estimating equations (Lam & Zhou, 2017) and further simplify the above model as follows:

$$\begin{aligned}&\min_\pi\max_{\lambda\geq 0}\max_{\|g\|_L\leq 1}\min_{\tau\geq 0}\quad \alpha\mathop{\mathbb{E}}_{d^R}[\tau g-g+\lambda\tau C]-\lambda\beta\\ &\text{subject to:}\quad \mathop{\mathbb{E}}_{d^R}[\phi(s',a')(\tau(s',a')-\tau(s,a))]=0\\ &\qquad\qquad\quad \mathop{\mathbb{E}}_{d^R}[\tau(s,a)]=1.\end{aligned}$$

where $\phi:\mathcal{S}\times\mathcal{A}\to\Omega\subset\mathbb{R}^p$, with p potentially infinite but less than $|\mathcal{S}|\times|\mathcal{A}|$, is a feature map. Applying the generalized empirical likelihood method (Duchi et al., 2021) to the above quantity, we obtain the following confidence set:

$$\begin{aligned}C^f_{n,\xi}=\Bigg\{\tilde{\rho}(\pi)=\min_\tau\alpha\mathop{\mathbb{E}}_w[\tau g-g+\lambda\tau C-\lambda\beta]\Bigg|\\ w\in\mathcal{K}_f,\mathop{\mathbb{E}}_w[\Delta(x;\tau,\phi)]=0,\mathop{\mathbb{E}}_w[\tau-1]=0\Bigg\}\\ \text{where }\ \mathcal{K}_f=\left\{w\in\mathcal{P}^{n-1}(\hat{p}_n),D_f(w\|\hat{p}_n)\leq\frac{\xi}{n}\right\},\end{aligned}$$

and where $\Delta(x;\tau,\phi)=\phi(s',a')(\tau(s',a')-\tau(s,a))$, $\hat{p}_n$ is the empirical data distribution, n is the number of data samples, ξ is the divergence tolerance, w are the uncertainty weights, and $\mathcal{P}^{n-1}$ is the simplex on the support of the empirical data distribution. The upper confidence bound on the sum of the cost of the policy and its distance from the reference distribution can then be obtained by:

$$\min_{\tau\geq 0}\min_{\mu\in\mathbb{R}^p,\nu}\max_{\|g\|_L\leq 1,w\in\mathcal{K}_f}\mathop{\mathbb{E}}_w[l(x;\tau,\mu,\nu)]\tag{5}$$

where $l(x;\tau,\mu,\nu)=\tau g-g+\lambda\tau C-\lambda\beta+\mu^T\Delta(x;\tau,\phi)+\nu-\nu\tau$. Thus $l(x;\tau,\mu,\nu)$ is the Lagrangian constructed from the confidence set constraints and the Lagrange multiplier $\mu \in \mathbb{R}^p$ takes the feature representation of the d-LP constraints back to $\mathbb{R}$. Setting ξ to be $\chi^{2,1-\alpha}_{(1)}$ in the construction of the set $\mathcal{K}_f$ renders the above upper bound an asymptotic $(1-\alpha)$ confidence interval for the COPO projection objective. Here, $\chi^{2,1-\alpha}_{(1)}$ is the $(1-\alpha)$ quantile for the Chi-square distribution with 1 degree of freedom. Lastly, we note that a lower bound may be derived in a similar manner following the procedure introduced by Dai et al. (2020).

3.2.1. Finite Sample Analysis

We now provide high probability finite-sample, non-asymptotic guarantees that the true policy cost identified by COPO will be lower than the computed upper bound. Previous offline constrained methods typically only guarantee that a point estimate of the policy cost will converge to a point below a pre-specified budget. Importantly, however, these estimates are based on a *single* (potentially small) set of experiences, which may not be sufficient to fully characterize the stochasticity of the problem. Here, we provide analysis that allows us to formally characterize how COPO performs under the more realistic finite-data setting.We first state some of the necessary assumptions (most of which are also assumed in Dai et al. (2020)) and show the boundedness and Lipschitz continuity of the loss functional. We then state a few Lemmas and present a corresponding proof of the finite sample analysis in Dai et al. (2020). Here, we assume the more general case of discounted settings.

Assumption 3.1. (Compactness of $\mathcal{S}$ and $\mathcal{A}$) The state and action spaces, $\mathcal{S}, \mathcal{A}$ are compact.

Assumption 3.2. (Stationary ratio regularity (Dai et al., 2020)) The target distribution correction ratio τ^* is bounded (i.e. $\|\tau^*\|_\infty \leq C_\tau < \infty$) and $\tau^* \in \mathcal{F}_\tau$, where $\mathcal{F}_\tau$ is a convex, compact and bounded Reproducing Kernel Hilbert Space (RKHS) with bounded kernel function $\|k((\cdot,\cdot,(s,a))\|_{\mathcal{F}_\tau} \leq K$.

Assumption 3.3. (Embedding feature regularity (Dai et al., 2020)) There exist finite constants C_μ, C_ϕ such that $\|\mu\|_2 \leq C_\mu, \|\phi\|_2 \leq C_\phi$. Further, $\phi(s,a)$ is L_ϕ Lipschitz continuous.

The previous two assumptions yield the following implications: *(i)* $\|\mu^T\phi\|_\infty \leq \|\mu\|_2 C_\mu$; *(ii)* $\|\phi\|_2 \leq C_\mu C_\phi$; and *(iii)* $\mu^T\phi(s,a)$ is Lipschitz continuous. Additionally, let us define $\mathcal{F}_\mu = \{\mu \mid \|\mu\|_2 \leq C_\mu\}$ as the function class of μ.

Lemma 3.4. *(Lipschitz continuity) Under Assumptions 3.1, 3.2, and 3.3, l is bounded (i.e.* $\|l(x;\tau,\mu,g)\|_\infty \leq M$*) and Lipschitz in* (τ,μ,g) *with constant* C_l.

Proof. The compactness of the state and action spaces, along with the boundedness of τ, imply the boundedness of the Wasserstein component of the loss functional. The boundedness of the cost value component follows from Lemma 9 in Dai et al. (2020). We define $M := (C_\tau+1)(1-\gamma)C_\beta C_\phi + C_\tau C_{\max} + C_g$, where $C_{\max}$ is the maximal cost from the data set and C_g is the Wasserstein bound. Lipschitz continuity follows from the proof of Lipschitz continuity of the loss functional from Dai et al. (2020) and the fact that the product of two Lipschitz functions (τ and g) is locally Lipschitz and the sum of Lipschitz functions is Lipschitz (τg, g, and the loss functional from Dai et al. (2020)). □

Under these assumptions, we can now derive non-asymptotic high-probability statements similar to the ones in Dai et al. (2020). Since the relevant arguments follow the same structure and reasoning to those in the Appendix E.2 of Dai et al. (2020), here we provide only the necessary adjustments and final statement. In particular, the probabilistic statements in our derivation of the finite sample confidence bounds will be made with probability at least $1 - 6\mathcal{N}_\infty(\mathcal{F}_\tau,\epsilon,2n)\mathcal{N}_\infty(\mathcal{F}_\mu,\epsilon,2n)\mathcal{N}_\infty(\mathcal{F}_g,\epsilon,2n)e^{-\frac{\xi}{18}}$. This is due to the addition of the function g in the loss functional and the fact that we only care about the one sided cost (upper) bound. Here, $\mathcal{N}_\infty(\mathcal{F},\epsilon,2n)$ is the l-∞ covering number of the functional class $\mathcal{F}$ with ϵ-net and $2n$ samples. The final finite sample statement is thus:

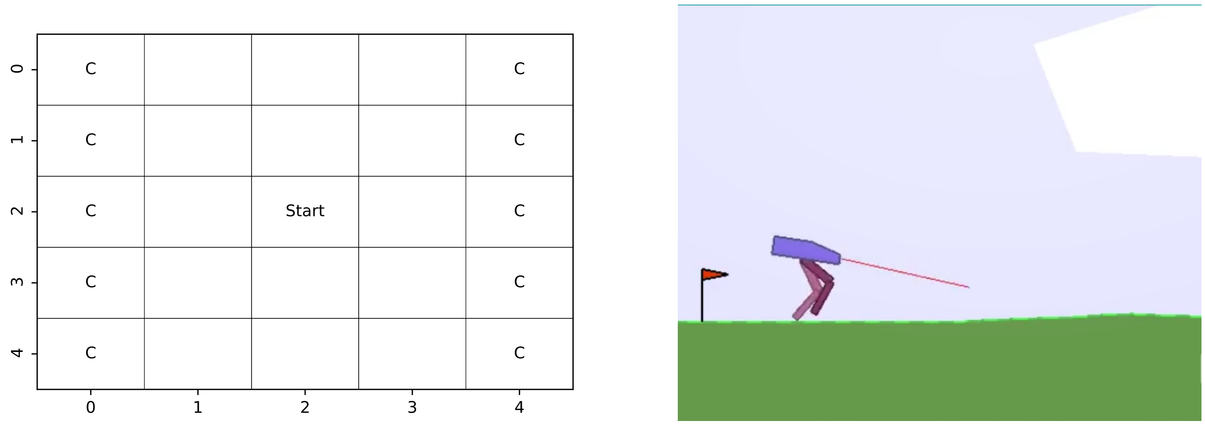
$$\mathbb{P}\left(\rho_\pi \leq u_n + \kappa\right) \geq \\ 1 - 6\exp\left(c_1 + 2(d_{\mathcal{F}_\tau} + d_{\mathcal{F}_\mu} + d_{\mathcal{F}_g} - 1)\log n - \frac{\xi}{18}\right)$$

where $d_{\mathcal{F}} = \mathcal{VC}(\mathcal{F})$ denotes the VC-dimension of the function class $\mathcal{F}$. In addition, we have $c_1 = 3c + \log d_{\mathcal{F}_\tau} + \log d_{\mathcal{F}_\mu} + \log d_{\mathcal{F}_g} + (d_{\mathcal{F}_\tau} + d_{\mathcal{F}_\mu} + d_{\mathcal{F}_g} - 1)$ and $\kappa = \frac{11M\xi}{6n} + 2\frac{C_l M}{n}\left(1 + 2\sqrt{\frac{\xi}{9n}}\right)$. Lastly, u_n is the solution to the optimization problem in Equation 5.

3.3. Algorithm

In Algorithm 1, we provide a sketch of the complete COPO algorithm, based on the derivations presented in the previous sections. This sketch reflects the main steps required so that one may use the COPO algorithm in practice. As shown in Algorithm 1, COPO can be deployed in conjunction with any reward maximizing offline RL algorithms, denoted there by $\mathcal{A}$.

In our experiments (Section 4), we use the AlgaeDICE algorithm (Nachum et al., 2019b) for algorithm $\mathcal{A}$, but other algorithms such as OptiDICE (Lee et al., 2021), CQL (Kumar et al., 2020), or Fisher-BRC (Kostrikov et al., 2021) can be used, alternatively. If the policy optimization algorithm $\mathcal{A}$ does not produce the visitation density of the reward optimal policy, then this quantity can be estimated using a distribution correction estimation (DICE) algorithm such as DualDICE (Nachum et al., 2019a), denoted in Algorithm 1



	0	1	2	3	4
0	C				C
1	C				C
2	C		Start		C
3	C				C
4	C				C

Figure 1. (Left) The Walk-Around-Grid Environment. States with costs are marked with a ”C”. (Right) The cost-constrained BipedalWalker domain.

Algorithm 1 COPO Algorithm Sketch

```
Input Dataset $\mathcal{D} = \{s_i, a_i, r_i, s'_i\}_{i=0}^{n}$
      Offline policy optimization algorithm, $\mathcal{A}$
      Offline DICE algorithm, $\mathcal{P}$
if $\mathcal{D}$ is collected by a reward optimal policy then
    $\pi_C \leftarrow$ COPO($\mathcal{D}$)
else
    Approximate reward optimal policy $\pi_R$ by running $\mathcal{A}(\mathcal{D})$
    Approximate reward optimal policy visitation density $d^{\pi_R}$ by running $\mathcal{P}(\mathcal{D}, \pi_R)$
    $\pi_C \leftarrow$ COPO($d^{\pi_R}$)
end if
return $\pi_C$
```

as $\mathcal{P}$. We note that deploying such an algorithm is not necessary if the data distribution used by COPO corresponds to the reward-optimal visitation density. Lastly, the COPO$(\cdot)$ function in Algorithm 1 represents running the optimization routine associated with the projection step in Equation (2).

4. Experiments

We now empirically demonstrate that COPO achieves state-of-the-art performance on discrete and continuous control problems, while offering stronger and more robust theoretical guarantees[2].

4.1. Walk-Around-Grid

To test the efficacy of our novel COPO algorithm, we designed a simple simulated robotic navigation problem. This domain, titled Walk-Around-Grid, is an infinite horizon, 5x5 grid world in which the agent, starting in the middle state, receives a maximal reward of 1.0 at each step for walking counterclockwise around the outermost edges of the grid. The agent receives a cost of 1.0 upon transitioning into a state in either the rightmost or leftmost columns. A reward of 0.5 is given for walking counterclockwise along states inside the outermost edges. Intuitively, a reward-optimal policy for this domain would cause the agent to to walk counter-clockwise around the outside edges of the grid, while a reward-optimal policy that satisfies all cost constraints would encode a behavior corresponding to the agent walking counter-clockwise around the rectangle but *inside* the columns with constraints. A depiction of the Walk-Around-Grid is presented in the top of Figure 1.

We now compare the performance of our developed COPO algorithm against BPLC—a state-of-the-art offline constrained policy optimization method (Le et al., 2019). The offline data set on which all algorithms are trained consists of 1000 trajectories (each cutoff after 250 steps) collected from a uniform random policy.

Table 1 shows the performance of COPO and BPLC when trained on uniformly sampled random data and for various settings of the cost budget, β. We selected a range of budget values for which feasible policies are likely to exist, so that both algorithms have a space on which to search for solutions that maximize return and that do not break constraints. In our experiments, for each budget setting, both COPO and BPLC were run for 100 trials. The first column of the table indicates which methods satisfy the desired cost constraint for different budgets. Notice that both COPO and BPLC are capable of ensuring that the cost of the returned policy is under the desired threshold. The second column of the table presents the *cost spread* of each algorithm; defined as twice the standard error of the per-step cost. It reflects how consistently an algorithm achieves a particular desired cost level,

[2] All experiments were executed on a server with a single GPU and 24 CPUs using different seeds for each independent run.

	Cost constraint satisfied		Cost spread (lower is better)		Mean per-step reward (higher is better)	
Budget	COPO	BPLC	**COPO**	BPLC	**COPO**	BPLC
0.4	Yes	Yes	$\mathbf{7.396e^{-3}}$	$3.714e^{-2}$	$\mathbf{0.761}$	0.742
0.5	Yes	Yes	$\mathbf{8.253e^{-3}}$	$4.518e^{-2}$	$\mathbf{0.840}$	0.779
0.6	Yes	Yes	$6.990e^{-2}$	$\mathbf{4.001e^{-2}}$	$\mathbf{0.901}$	0.812
0.7	Yes	Yes	$\mathbf{9.921e^{-4}}$	$3.276e^{-2}$	$\mathbf{0.992}$	0.866
0.8	Yes	Yes	$\mathbf{9.856e^{-4}}$	$3.230e^{-2}$	$\mathbf{0.992}$	0.901
Average	—	—	$\boldsymbol{1.750e^{-2}}$	$3.750e^{-2}$	$\boldsymbol{0.897}$	0.821

Table 1. Performance of different offline policy optimization algorithms in the Walk-Around-Grid environment for various cost budgets. Results are computed over 100 trials. The *cost spread* is defined as twice the standard error of the per-step cost and reflects how consistently an algorithm achieves a particular desired cost level (lower is better).

thus a lower value is preferred. The range of cost values that are achieved by COPO are tighter than those achieved by BPLC, indicating that COPO's policies are more robust with respect to the particular finite data set given to the algorithm. This is particularly important in real-world applications where it is paramount that algorithms ensure safety with high probability even when trained using (potentially small) finite training sets. Finally, the third column of the table indicates the mean per-step reward achieved by each algorithm. COPO consistently outperforms BPLC across different budget values.

4.2. BipedalWalker Environment

To show the applicability of our COPO algorithm to more complex domains, we evaluate its performance in a modified version the BipedalWalker domain, adapted from OpenAI (Brockman et al., 2016). A depiction of this environment is shown in the bottom of Figure 1. The modified BipedalWalker environment models a robotic control task where the goal is to control a two-legged robot so that it walks as far as possible, but under costs that penalize it for exceeding a particular maximum velocity. This is a continuous control problem in which the 14-dimensional state vector modeling the current pose and velocity of the robot. The 4-dimensional actions of the robot control motor torques. The reward signal is proportional to the distance the agent has traveled from the initial state. To test the efficacy of the COPO algorithm, we extend the original BipedalWalker environment with a binary cost function reflecting situations where the walker's linear velocity exceeds a maximum allowed velocity threshold. A policy attaining zero costs would, therefore, keep the walker below a given speed limit while allowing it to move as far as possible.

To test the projection step of COPO, we first trained an online AlgaeDICE agent to identify a purely reward-optimal successful policy[3]. We constructed 10 statistically-independent data sets by sampling from such a reward-optimal policy and recording both the trajectories and the corresponding incurred costs. We subsequently ran our novel projection algorithm on each data set and evaluated the projected policy (in terms of per-step cost and per-step reward) over the course of the algorithm's execution. The results for this experiment are shown in Figure 2. Here, each curve depicts the average performance of a given algorithm (COPO or BPLC[4]) over 20 environment episodes. Each point in these curves was computed over 10 trials, and error curves represent one standard error.

As shown in Figure 2, both COPO and BPLC approach a similar per-step reward after 7,000 timesteps. Importantly, however, COPO is always capable of identifying lower-cost policies than BPLC at the end of the training procedure. This is consistent with the observation that, given a fixed-sized data set—and a sufficient number of iterations; i.e., processing time—COPO is guaranteed to produce more robust high probability confidence bounds on the true cost of a policy, and these bounds hold even under finite data. BPLC, by contrast, is *not* guaranteed to return policies that are safe in the non-asymptotic case, and in fact identifies policies whose cost is 10.4% higher than COPO's. The above results highlight COPO's advancement of the state-of-the-art in critical applications where processing time is cheap but where the deployment of unsafe policies may be catastrophic.

[3] Success is defined as achieving average episode return (sum of rewards) above 300.

[4] The original formulation of BPLC was constructed upon the Fitted-Q Iteration algorithm and could only be applied to discrete-action problems. To address this, we extended BPLC by substituting Fitted Q-Iteration with the AlgaeDICE algorithm. This also allows BPLC to reap the benefits of DICE estimation.

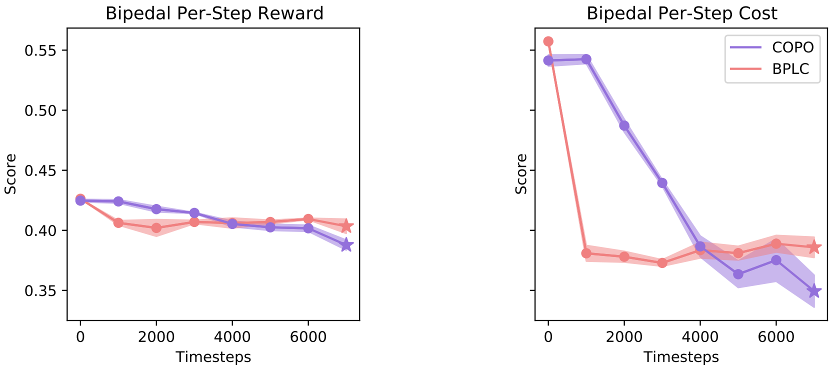


Figure 2. Performance of COPO and BPLC in the cost-constrained BipedalWalker environment with a budget of 0.35. Axes are the same scale on both plots. Timesteps refers to the number of steps run for each optimization routine.

5. Connections to Related Work

Many techniques exist that tackle the offline RL problem and the constrained RL setting. The Distribution Correction Estimation (DICE) family of RL algorithms (Lee et al., 2021; Kostrikov et al., 2020; Nachum et al., 2019a;b; Zhang et al., 2020a;b; Yang et al., 2020a; Dai et al., 2020) is a gamut of offline policy optimization and evaluation algorithms that rely upon explicitly estimating the distributional shift between the target policy and the offline data distribution. Our algorithm may be viewed as an application of policy optimization DICE methods to problems formulated as CMDPs (Altman, 1999).

Online policy optimization in the constrained setting has been studied most notably in the Constrained Policy Optimization (CPO) framework (Achiam et al., 2017). This is a trust region-based policy optimization algorithm that aims to satisfy constraints on accrued costs at each policy iteration (Schulman et al., 2015). Additionally, projection-based algorithms (Yang, Rosca, Narasimhan, and Ramadge, 2020b; Zhang, Vuong, and Ross, 2020c) for policy optimization in CMDPs attempt to remedy approximation errors in the CPO algorithm by employing a policy-space projection step. Our work can be seen as an offline RL counterpart to these methods which employs DICE estimation and convex duality (Boyd & Vandenberghe, 2004). Batch Policy Learning under Constraints (BPLC) (Le et al., 2019) is the closest prior work to ours, and the baseline algorithm with which we compare empirical performance. BPLC tackles the problem of offline RL with constraints via an adversarial game-theoretic approach. Our work differs from BPLC in that our COPO algorithm explicitly accounts for distributional shift, while BPLC does not. This implies that the safety guarantees provided by BPLC are not necessarily robust to variability inherent to finite training sets. COPO, by contrast, produces high probability confidence bounds that allow for a more robust estimation the true cost of a policy given a finite amount of training data.

Offline RL algorithms such as Conservative Q-Learning (CQL) do not consider scenarios with costs and constraints (Kumar et al., 2020). They do, however, learn conservative estimates of Q-functions based on measures of data uncertainty. CQL and other related algorithms (Bharadhwaj et al., 2021) may be viewed as offline RL algorithms with explicit safety considerations and are loosely related to ours. The objective of COPO's projection step can be viewed as a behavior regularized offline RL objective (Wu, Tucker, and Nachum, 2019; Kostrikov, Tompson, Fergus, and Nachum, 2021).

6. Conclusion

We introduced a novel Constrained Offline Policy Optimization algorithm (COPO) for efficiently learning cost-constrained policies in a fully offline manner. COPO is based on a novel constrained policy projection technique for identifying cost-feasible policies. It improves upon the state-of-the-art in offline constrained policy optimization by explicitly accounting for distributional shift and by offering *non-asymptotic high confidence confidence bounds* on the true cost of a policy. These formal properties are superior to those in the existing literature, which only guarantee convergence of a point estimate on a single sample of the data generating distribution. Our experiments demonstrate that COPO improves upon the state-of-the-art by achieving lower-cost policies and by producing policies with a tighter range of cost values. This indicates that COPO's policies are more robust with respect to the particular finite data set given to the algorithm, which is particularly important in real-world applications where it is paramount that algorithms ensure safety with high probability even when trained using potentially small training sets. Future work will address some of the theoretical limitations of COPO. We emphasize, first, the importance of formally characterizing how different distance functions—used in the projection step—might affect performance in different families of constrained MDPs. We would also like to perform empirical analyses of COPO in physical, non-simulated systems, in order to empirically evaluate its performance when operating under severe limitations on the amount of available training data.

Acknowledgements

ACKNOWLEDGMENT OF SUPPORT AND DISCLAIMER: (a) Contractor acknowledges Government's support in the publication of this paper. This material is based upon work supported by the US Navy Contract No. N6833520C0964. (b) Any opinions, findings and conclusions or recommendation expressed in this material are those of the author(s) and do not necessarily reflect the views of the US Navy.

References

Achiam, J., Held, D., Tamar, A., and Abbeel, P. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning*. PMLR, 2017.

Altman, E. Constrained markov decision processes. *Chapman and Hall/CRC*, 1999.

Bastani, M. Model-free intelligent diabetes management using machine learning. *Ph.D. Thesis*, 2014.

Bharadhwaj, H., Kumar, A., Rhinehart, N., Levine, S., Shkurti, F., and Garg, A. Conservative safety critics for exploration. In *International Conference on Learning Representations*, 2021.

Boyd, S. and Vandenberghe, L. *Convex Optimization*. Cambridge University Press, 2004.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. Openai gym, 2016.

Dai, B., Nachum, O., Chow, Y., Li, L., Szepesvari, C., and Schuurmans, D. Coindice: Off-policy confidence interval estimation. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 9398–9411. Curran Associates, Inc., 2020.

Duchi, J. C., Glynn, P. W., and Namkoong, H. Statistics of robust optimization: A generalized empirical likelihood approach. *Mathematics of Operations Research*, 0(0): null, 2021.

Kostrikov, I., Nachum, O., and Tompson, J. Imitation learning via off-policy distribution matching. In *International Conference on Learning Representations*, 2020.

Kostrikov, I., Tompson, J., Fergus, R., and Nachum, O. Offline reinforcement learning with fisher divergence critic regularization, 2021.

Kumar, A., Zhou, A., Tucker, G., and Levine, S. Conservative q-learning for offline reinforcement learning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1179–1191. Curran Associates, Inc., 2020.

Lam, H. and Zhou, E. The empirical likelihood approach to quantifying uncertainty in sample average approximation. *Operations Research Letters*, 45(4):301–307, 2017. ISSN 0167-6377.

Le, H., Voloshin, C., and Yue, Y. Batch policy learning under constraints. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3703–3712. PMLR, 09–15 Jun 2019.

Lee, J., Jeon, W., Lee, B.-J., Pineau, J., and Kim, K.-E. Optidice: Offline policy optimization via stationary distribution correction estimation. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021.

Levine, S., Kumar, A., Tucker, G., and Fu, J. Offline reinforcement learning: Tutorial, review, and perspectives on open problems, 2020.

Nachum, O. and Dai, B. Reinforcement learning via fenchel-rockafellar duality, 2020.

Nachum, O., Chow, Y., Dai, B., and Li, L. Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019a.

Nachum, O., Dai, B., Kostrikov, I., Chow, Y., Li, L., and Schuurmans, D. Algaedice: Policy gradient from arbitrary experience, 2019b.

Rockafellar, R. T. *Convex Analysis*. Princeton University Press, 1970. ISBN 9780691015866.

Saria, S. Individualized sepsis treatment using reinforcement learning. *Nature Medicine*, 24:1641–1642, 2018.

Sascha Lange, Thomas Gabel, M. R. Batch reinforcement learning. In *Reinforcement Learning*, pp. 45–73. Springer, 2012.

Schulman, J., Levine, S., Moritz, P., Jordan, M. I., and Abbeel, P. Trust region policy optimization. *CoRR*, abs/1502.05477, 2015.

Wu, Y., Tucker, G., and Nachum, O. Behavior regularized offline reinforcement learning, 2019.

Yang, M., Dai, B., Nachum, O., Tucker, G., and Schuurmans, D. Offline policy selection under uncertainty, 2020a.

Yang, T.-Y., Rosca, J., Narasimhan, K., and Ramadge, P. J. Projection-based constrained policy optimization. In *International Conference on Learning Representations*, 2020b.

Zhang, R., Dai, B., Li, L., and Schuurmans, D. Gendice: Generalized offline estimation of stationary values. In *International Conference on Learning Representations*, 2020a.

Zhang, S., Liu, B., and Whiteson, S. GradientDICE: Rethinking generalized offline estimation of stationary values. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 11194–11203. PMLR, 13–18 Jul 2020b.

Zhang, Y., Vuong, Q., and Ross, K. First order constrained optimization in policy space. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 15338–15349. Curran Associates, Inc., 2020c.