Provably Convergent Policy Optimization via Metric-aware Trust Region Methods

Anonymous authors
Paper under double-blind review

Abstract

Trust-region methods based on Kullback-Leibler divergence are pervasively used to stabilize policy optimization in reinforcement learning. In this paper, we exploit more flexible metrics and examine two natural extensions of policy optimization with Wasserstein and Sinkhorn trust regions, namely *Wasserstein policy optimization (WPO)* and *Sinkhorn policy optimization (SPO)*. Instead of restricting the policy to a parametric distribution class, we directly optimize the policy distribution and derive their close-form policy updates based on the Lagrangian duality. Theoretically, we show that WPO guarantees a monotonic performance improvement, and SPO provably converges to WPO as the entropic regularizer diminishes. Moreover, we prove that with a decaying Lagrangian multiplier to the trust region constraint, both methods converge to global optimality. Experiments across tabular domains, robotic locomotion, and continuous control tasks further demonstrate the performance improvement of both approaches, more robustness of WPO to sample insufficiency, and faster convergence of SPO, over state-of-art policy gradient methods.

1 Introduction

Policy-based reinforcement learning (RL) approaches have received remarkable success in many domains, including video games (Mnih et al., 2013; Mnih et al., 2015), board games (Silver et al., 2016; Heinrich & Silver, 2016), robotics (Grudic et al., 2003; Gu et al., 2017), and continuous control tasks (Duan et al., 2016; Schulman et al., 2016). One prominent example is policy gradient method (Grudic et al., 2003; Peters & Schaal, 2006; Lillicrap et al., 2016; Sutton et al., 1999; Williams, 1992; Mnih et al., 2016; Silver et al., 2014). The core idea is to represent the policy with a probability distribution $\pi_\theta(a|s) = P[a|s;\theta]$, such that the action a in state s is chosen stochastically following the policy π_θ controlled by parameter θ. Determining the right step size to update the policy is crucial for maintaining the stability of policy gradient methods: too conservative choice of stepsizes result in slow convergence, while too large stepsizes may lead to catastrophically bad updates.

To control the size of policy updates, Kullback-Leibler (KL) divergence is commonly adopted to measure the difference between two policies. For example, the seminal work on trust region policy optimization (TRPO) by Schulman et al. (2015) introduced KL divergence based constraints (trust region constraints) to restrict the size of the policy update; see also Peng et al. (2019); Abdolmaleki et al. (2018). Kakade (2001) and Schulman et al. (2017) introduced a KL-based penalty term to the objective to prevent excessive policy shift.

Though KL-based policy optimization has achieved promising results, it remains interesting whether using other metrics to gauge the similarity between policies could bring additional benefits. Recently, few work (Richemond & Maginnis, 2017; Zhang et al., 2018; Moskovitz et al., 2021; Pacchiano et al., 2020) has explored the Wasserstein metric to restrict the deviation between consecutive policies. Compared with KL divergence, the Wasserstein metric has several desirable properties. Firstly, it is a true symmetric distance measure. Secondly, it allows flexible user-defined costs between actions and is less sensitive to ill-posed likelihood ratios. Thirdly but most importantly, the Wasserstein metric takes into account the geometry of the metric space (Panaretos & Zemel, 2019) and allows distributions to have different or even non-overlapping supports.

Motivating Example: Below we provide an example of a grid world (see Figure 1) that illustrates the advantages of using the Wasserstein metric over the KL divergence to construct trust regions and policy updates. The grid world consists of 5 regular grids and 2 goal grids, and there are three possible actions: left, right, and pickup. The player always starts from the middle grid, and making a left or right move results in a reward of -1. Picking up yields a reward of -3 at regular grids, $+5$ at the blue goal grid, and $+10$ at the red goal grid. An episode terminates either at the maximum length of 10 or immediately after picking up. We define the geometric distance between left and right actions to be 1, and 4 between other actions.

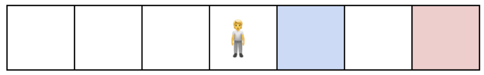
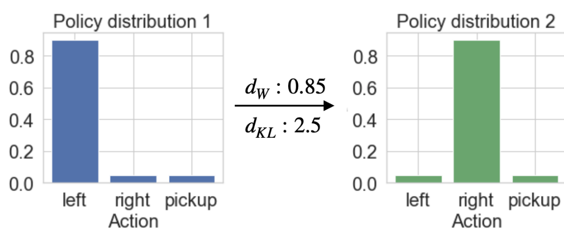
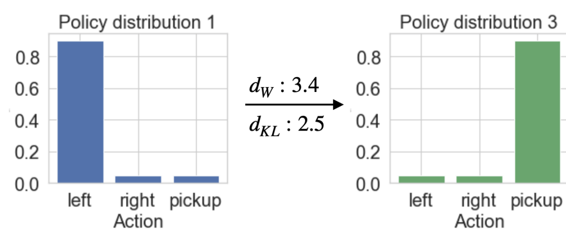


Figure 1: Motivating grid world example

Figure 2 shows the Wasserstein distance and KL divergence for different policy shifts of this grid world example. We can see that Wasserstein metric utilizes the geometric distance between actions to distinguish the shift of policy distribution to a close action (policy distribution 1 $\to$ 2 in figure 2a) from the shift to a far action (policy distribution 1 $\to$ 3 in figure 2b), while KL divergence does not. Figure 3 demonstrates the constrained policy updates based on Wasserstein distance and KL divergence respectively with a fixed trust region size 1. We can see that Wasserstein-based policy update finds the optimal policy faster than KL-based policy update. This is because KL distance is larger than Waserstein when considering policy shifts of close actions (see figure 2a). Therefore, Wasserstein policy update is able to shift action (from left to right) in multiple states, while KL update is only allowed to shift action in a single state. Besides, KL policy update keeps using a suboptimal short-sighted solution between the 2nd and 4th iteration, which further slows down the convergence.



(a) Policy shift of close action



(b) Policy shift of far action

Figure 2: Wasserstein utilizes geometric feature of action space

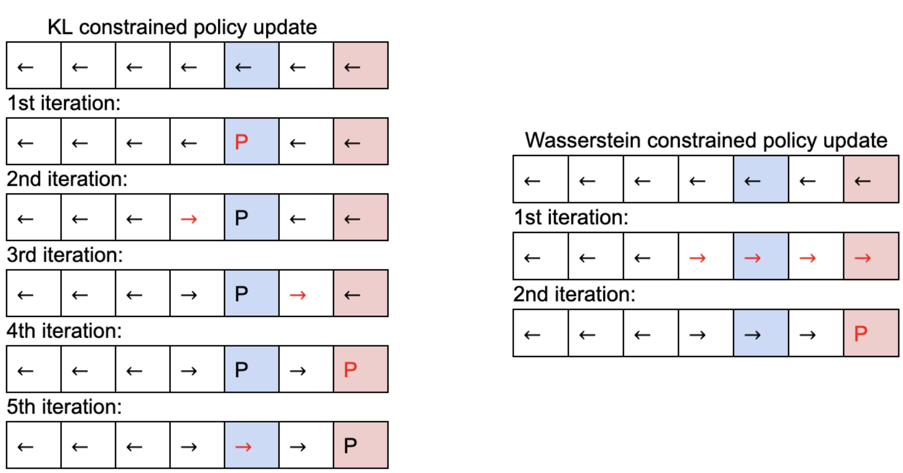


Figure 3: Demonstration of policy updates under different trust regions

However, the challenge of applying the Wasserstein metric for policy optimization is also evident: evaluating the Wasserstein distance requires solving an optimal transport problem, which could be computationally expensive. To avoid this computation hurdle, existing work resorts to different techniques to *approximate the policy update* under Wasserstein regularization. For example, Richemond & Maginnis (2017) solved the resulting RL problem using Fokker-Planck equations; Zhang et al. (2018) introduced particle approximation method to estimate the Wasserstein gradient flow. Recently, Moskovitz et al. (2021) instead considered the second-order Taylor expansion of Wasserstein distance based on Wasserstein information matrix to characterize the local behavioral structure of policies. Pacchiano et al. (2020) tackled behavior-guided policy optimization with smooth Wasserstein regularization by solving an approximate dual reformulation defined on reproducing kernel Hilbert spaces. Aside from such approximation, some of these work also limits the policy representation to a particular parametric distribution class, As indicated in Tessler et al. (2019), since parametric distributions are not convex in the distribution space, optimizing over such distributions results in local movements in the action space and thus leads to convergence to a sub-optimal solution. Henceforth, the theoretical performance of policy optimization under the Wasserstein metric remains elusive in light of these approximation errors.

In this paper, we study policy optimization with trust regions based on Wasserstein distance and Sinkhorn divergence. The latter is a smooth variant of Waserstein distance by imposing an entropic regularization to the optimal transport problem (Cuturi, 2013). We call them, *Wasserstein Policy Optimization (WPO)* and *Sinkhorn Policy Optimization (SPO)*, respectively. Instead of confining the distribution of policy to a particular distribution class, we work on the space of policy distribution directly, and consider all admissible policies that are within the trust regions with the goal of avoiding approximation errors. Unlike existing work, we focus on *exact characterization* of the policy updates. We highlight our contributions as follows:

1. **Algorithms:** We develop closed-form expressions of the policy updates for both WPO and SPO based on the corresponding optimal Lagrangian multipliers of the trust region constraints. To the best of our knowledge, this is the first explicit closed-form updates for policy optimization based on Wasserstein and Sinkhorn trust regions. In particular, the optimal Lagrangian multiplier of SPO admits a simple form and can be computed efficiently. A practical on-policy actor-critic algorithm is proposed based on the derived expressions of policy updates and advantage value function estimation.
2. **Theory:** We theoretically show that WPO guarantees a *monotonic performance improvement* through the iterations, *even with non-optimal Lagrangian multipliers*. We also prove that SPO converges to WPO as the entropic regularizer diminishes. Moreover, we prove that with a decaying schedule of the multiplier, SPO and WPO converge to *global optimality*, and with a constant multiplier, both methods converge *linearly* up to a neighborhood of the optimal value. To our best knowledge, this appears to be the first convergence rate analysis of policy optimization based on Wasserstein-type metrics.
3. **Experiments:** We provide comprehensive evaluation on the efficiency of WPO and SPO under several types of testing environments including tabular domains, robotic locomotion tasks, and further extend it to continuous control tasks. Compared to state-of-art policy gradients approaches that use KL divergence such as TRPO and PPO and those use Wasserstein metric such as Wasserstein Natural Policy Gradient (WNPG) (Moskovitz et al., 2021) and Behavior Guided Policy Gradients (BGPG) (Pacchiano et al., 2020), our methods achieve better sample efficiency, faster convergence, and improved final performance. Numerical study indicates that by properly choosing the weight of entropic regularizer, SPO achieves a better trade-off between convergence and final performance than WPO.

Related work: Wasserstein-like metrics have been explored in a number of works in the context reinforcement learning. Ferns et al. (2004) first introduced bisimulation metrics based on Wasserstein distance to quantify behavioral similarity between states for the purpose of state aggregation. Such bisimulation metrics were recently utilized for representation learning of RL; see e.g., Castro (2020); Agarwal et al. (2021b). In addition, few recent work has also exploited Wasserstein distance for imitation learning (see e.g., Xiao et al. (2019); Dadashi et al. (2021)) and unsupervised RL (see e.g., He et al. (2022)). The most related work to ours are Richemond & Maginnis (2017); Zhang et al. (2018); Moskovitz et al. (2021); Pacchiano et al. (2020). These work directly use Wasserstein-like distance to measure proximity of policies instead of states. Unlike ours, these work apply Wasserstein distance as an explicit penalty function instead of trust-region constraints.

Besides, they use different strategies to approximate the Wasserstein distance. Moreover, none of these work has shown the monotonic performance improvement with their policy update. As for the use of Sinkhorn divergence in RL, the only related work, to our best knowledge, is Pacchiano et al. (2020), where the entropy regularization is merely used to mitigate the computational burden of computing Wasserstein metric, yet no explicit form of policy update is provided.

Wasserstsein-like metrics are also pervasively studied in distributionally robust optimization (DRO); see e.g., Esfahani & Kuhn (2018); Gao & Kleywegt (2016); Zhao & Guan (2018); Blanchet & Murthy (2019). We also point out that a recent concurrent work by Wang et al. (2021a) studied DRO using the Sinkhorn distance. Our duality formulations are largely inspired from existing work in DRO. However, we note that constrained policy optimization is conceptually different from DRO. Constrained policy optimization focuses on finding the optimistic policy that falls in a trust region, whereas DRO (e.g., the KL DRO) aims to optimize some worst-case loss given by the adversarial distribution of unknown parameters within some ambiguity set.

2 Background and Notations

Markov Decision Process (MDP): We consider an infinite-horizon discounted MDP, defined by the tuple $(\mathcal{S}, \mathcal{A}, P, r, \upsilon, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the transition probability, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, $\upsilon : \mathcal{S} \to \mathbb{R}$ is the distribution of the initial state s_0, and $\gamma \in (0, 1)$ is the discount factor. We define the return of timestep t as the accumulated discounted reward from t, $R_t = \sum_{k=0}^{\infty} \gamma^k r(s_{t+k}, a_{t+k})$, and the value function as $V^\pi(s) = \mathbb{E}[R_t|s_t = s; \pi]$. The performance of a stochastic policy π is defined as $J(\pi) = \mathbb{E}_{s_0,a_0,s_1...}[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)]$ where $a_t \sim \pi(a_t|s_t)$, $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$. As shown in Kakade & Langford (2002), the expected return of a new policy π' can be expressed in terms of the advantage over the old policy π: $J(\pi') = J(\pi) + \mathbb{E}_{s\sim\rho_\upsilon^{\pi'},a\sim\pi'}[A^\pi(s, a)]$, where $A^\pi(s, a) = \mathbb{E}[R_t|s_t = s, a_t = a; \pi] - \mathbb{E}[R_t|s_t = s; \pi]$ represents the advantage function and ρ_υ^π represents the unnormalized discounted visitation frequencies with initial state distribution υ, i.e., $\rho_\upsilon^\pi(s) = \mathbb{E}_{s_0\sim\upsilon}[\sum_{t=0}^{\infty} \gamma^t P(s_t = s|s_0)]$.

Trust Region Policy Optimization (TRPO): In TRPO (Schulman et al., 2015), the policy π is parameterized as π_θ with parameter vector θ. For notation brevity, we use θ to represent the policy π_θ. Then, the new policy θ' is found in each iteration to maximize the expected improvement $J(\pi') - J(\pi)$, or equivalently, the expected value of the advantage function:

$$\begin{aligned} \max_{\theta'} \quad & \mathbb{E}_{s\sim\rho_\upsilon^\theta,a\sim\theta'}[A^\theta(s, a)] \\ \text{s.t.} \quad & \mathbb{E}_{s\sim\rho_\upsilon^\theta}[d_{\mathrm{KL}}(\theta', \theta)] \le \delta, \end{aligned} \tag{1}$$

where d_{KL} represents the KL divergence and δ is the threshold of the distance between the new and the old policies.

Wasserstein Distance: Given two probability distributions of policies π and π' on the discrete action space $\mathcal{A} = \{a_1, a_2, \ldots, a_N\}$, the Wasserstein distance between the policies is defined as:

$$d_{\mathrm{W}}(\pi', \pi) = \inf_{Q\in\Pi(\pi',\pi)} \langle Q, M\rangle, \tag{2}$$

where $\langle\cdot, \cdot\rangle$ denotes the Frobenius inner product. The infimum is taken over all joint distributions Q with marginals π' and π, and M is the cost matrix with $M_{ij} = d(a_i, a_j)$, where $d(a_i, a_j)$ is defined as the distance between actions a_i and a_j. Its largest entry in magnitude is denoted by $\|M\|_\infty$. In implementation, we use 1-Wasserstein distance, i.e., ℓ_1 norm $d(a_i, a_j) = \|a_i - a_j\|_1$.

Sinkhorn Divergence: Sinkhorn divergence (Cuturi, 2013) provides a smooth approximation of the Wasserstein distance by adding an entropic regularizer. The Sinkhorn divergence is defined as:

$$d_{\mathrm{S}}(\pi', \pi|\lambda) = \inf_{Q\in\Pi(\pi',\pi)} \left\{ \langle Q, M\rangle - \frac{1}{\lambda} h(Q) \right\}, \tag{3}$$

where $h(Q) = -\sum_{i=1}^{N}\sum_{j=1}^{N} Q_{ij}\log Q_{ij}$ represents the entropy term, and $\lambda > 0$ is a regularization parameter. The intuition of adding the entropic regularization is: since most elements of the optimal joint distribution Q will be 0 with a high probability, by trading the sparsity with entropy, a smoother and denser coupling between distributions can be achieved (Courty et al., 2014; 2016). Therefore, when the weight of the entropic regularization decreases (i.e., λ increases), the sparsity of the divergence increases, and the Sinkhorn divergence converges to the Wasserstein metric, i.e., $\lim_{\lambda\to\infty} d_{\mathrm{S}}(\pi', \pi|\lambda) = d_{\mathrm{W}}(\pi', \pi)$. More critically, Sinkhorn divergence is useful to mitigate the computational burden of computing Wasserstein distance. In fact, the efficiency improvement that Sinkhorn divergence and the related algorithms brought paves the way to utilize Wasserstein-like metrics in many machine learning domains, including online learning (Cesa-Bianchi & Lugosi, 2006), model selection (Juditsky et al., 2008; Rigollet & Tsybakov, 2011), generative modeling (Genevay et al., 2018; Petzka et al., 2017; Patrini et al., 2019), dimensionality reduction (Huang et al., 2021; Lin et al., 2020; Wang et al., 2021b).

3 Wasserstein Policy Optimization

Motivated by TRPO, here we consider a trust region based on the Wasserstein metric. Moreover, we lift the restrictive assumption that a policy has to follow a parametric distribution class by allowing all admissible policies. Then, the new policy π' is found in each iteration to maximize the estimated expected value of the advantage function. Therefore, the *Wasserstein Policy Optimization* (WPO) framework is shown as follows:

$$\begin{aligned} \max_{\pi'\in\mathcal{D}} \quad & \mathbb{E}_{s\sim\rho_\upsilon^\pi, a\sim\pi'(\cdot|s)}[A^\pi(s,a)] \\ \text{where} \quad & \mathcal{D} = \{\pi' | \mathbb{E}_{s\sim\rho_\upsilon^\pi}[d_{\mathrm{W}}(\pi'(\cdot|s), \pi(\cdot|s))] \le \delta\}, \end{aligned} \tag{4}$$

where the Wasserstein distance $d_{\mathrm{W}}(\cdot,\cdot)$ is defined in (2).

In most practical cases, the reward r is bounded and correspondingly, the accumulated discounted reward R_t is bounded. So without loss of generality, we make the following assumption:

Assumption 1. *Assume $A^\pi(s,a)$ is bounded, i.e., $\sup_{a\in\mathcal{A}, s\in\mathcal{S}} |A^\pi(s,a)| \le A^{max}$ for some $A^{max} > 0$.*

With Wasserstein metric based trust region constraint, we are able to derive the closed-form of the policy update shown in Theorem 1. The main idea is to form the Lagrangian dual of the constrained optimization problem presented above, which is inspired by the way to obtain the extremal distribution in Wasserstein DRO literature, see e.g., Kuhn et al. (2019); Blanchet & Murthy (2019); Zhao & Guan (2018). The detailed proof can be found in Appendix B.

Theorem 1. ***(Closed-form policy update)*** *Let $\kappa_s^\pi(\beta, j) = argmax_{k=1\ldots N}\{A^\pi(s,a_k) - \beta M_{kj}\}$, where M denotes the cost matrix. If Assumption 1 holds, then an optimal solution to (4) is:*

$$\pi^*(a_i|s) = \sum\nolimits_{j=1}^{N} \pi(a_j|s) f_s^*(i,j), \tag{5}$$

where $f_s^*(i,j) = 1$ if $i = \kappa_s^\pi(\beta^*, j)$ and $f_s^*(i,j) = 0$ otherwise, and β^* is an optimal Lagrangian multiplier corresponds to the following dual formulation:

$$\min_{\beta\ge 0} F(\beta) = \min_{\beta\ge 0}\{\beta\delta + \mathbb{E}_{s\sim\rho_\upsilon^\pi}\sum\nolimits_{j=1}^{N}\pi(a_j|s)\max_{i=1\ldots N}(A^\pi(s,a_i) - \beta M_{ij})\}. \tag{6}$$

Moreover, we have $\beta^* \le \bar{\beta}$, where $\bar{\beta} := \max_{s\in\mathcal{S},k,j=1\ldots N,k\ne j}(M_{kj})^{-1}(A^\pi(s,a_k) - A^\pi(s,a_j))$.

Remark 1. *For ease of notation and simplicity, we assume the uniqueness of $\kappa_s^\pi(\beta, j)$ in order to form the simple expression of f_s^* in Theorem 1. When it is not unique, a necessary condition for the optimality of π^* in (5) is $\sum_{i\in\mathcal{K}_s^\pi(\beta,j)} f_s^*(i,j) = 1$, and $f_s^*(i,j) = 0$ for $i \notin \mathcal{K}_s^\pi(\beta,j)$, where $\mathcal{K}_s^\pi(\beta,j) = argmax_{k=1\ldots N} A^\pi(s,a_k) - \beta M_{kj}$. The weight $f_s^*(i,j)$ for $i \in \mathcal{K}_s^\pi(\beta,j)$ could be determined through linear programming (see details in (17) in Appendix B).*

The exact policy update for WPO in (5) requires computing the optimal Lagrangian multiplier β^* by solving the one-dimensional subproblem (6). A closed-form of β^* is not easy to obtain in general, except for special cases of the distance $d(x, y)$ or cost matrix M. In Appendix C, we provide the closed form of β^* for the case when $d(x, y) = 0$ if $x = y$ and 1 otherwise.

WPO Policy Update: Based on Theorem 1, we introduce the following WPO policy updating rule:

$$\pi_{k+1}(a_i|s) = \mathbb{F}^{\text{WPO}}(\pi_k) := \sum_{j=1}^{N} \pi_k(a_j|s) f_s^k(i,j), \tag{WPO}$$

where $f_s^k(i,j) = 1$ if $i = \kappa_s^{\pi_k}(\beta_k, j)$ and 0 otherwise. Note that different from (5), we allow β_k to be chosen arbitrarily and time dependently. We show that such policy update always leads to a monotonic improvement of the performance even when β_k is not the optimal Lagrangian multiplier. In particular, we propose two strategies to update multiplier β_k:

(i) Approximation of optimal β_k: To improve the convergence, we can approximately solve the optimal Lagrangian multiplier based on Sinkhorn divergence. More details in Section 4.

(ii) Time-dependent β_k: To improve the computational efficiency, we can simply treat β_k as a time-dependent parameter, e.g., we can set β_k as a diminishing sequence. In this setting, (WPO) produces the solution to the following penalty version of problem (4) (with $d = d_{\text{W}}$):

$$\max_{\pi_{k+1}} \mathbb{E}_{s\sim\rho_\upsilon^{\pi_k}, a\sim\pi_{k+1}(\cdot|s)}[A^{\pi_k}(s,a)] - \beta_k \mathbb{E}_{s\sim\rho_\upsilon^{\pi_k}}[d(\pi_{k+1}(\cdot|s), \pi_k(\cdot|s))]. \tag{7}$$

4 Sinkhorn Policy Optimization

In this section, we introduce Sinkhorn policy optimization (SPO) by constructing trust region with Sinkhorn divergence. In the following theorem, we derive the optimal policy update in each step when using Sinkhorn divergence based trust region. Detailed proofs are provided in Appendix D.

Theorem 2. *If Assumption 1 holds, then the optimal solution to (4) with Sinkhorn divergence is:*

$$\pi_\lambda^*(a_i|s) = \sum_{j=1}^{N} \frac{\exp\left(\frac{\lambda}{\beta_\lambda^*} A^\pi(s,a_i) - \lambda M_{ij}\right)}{\sum_{k=1}^{N} \exp\left(\frac{\lambda}{\beta_\lambda^*} A^\pi(s,a_k) - \lambda M_{kj}\right)} \pi(a_j|s), \tag{8}$$

where M denotes the cost matrix and β_λ^* is an optimal solution to the following dual formulation:

$$\begin{aligned}\min_{\beta\geq 0} F_\lambda(\beta) = \min_{\beta\geq 0} \Big\{ &\beta\delta - \mathbb{E}_{s\sim\rho_\upsilon^\pi} \sum_{j=1}^{N} \pi(a_j|s)\Big(\frac{\beta}{\lambda} + \frac{\beta}{\lambda}\ln(\pi(a_j|s)) - \frac{\beta}{\lambda}\ln\Big[\sum_{i=1}^{N}\exp\Big(\frac{\lambda}{\beta}A^\pi(s,a_i) - \lambda M_{ij}\Big)\Big]\Big) \\ &\mathbb{E}_{s\sim\rho_\upsilon^\pi} \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{\beta}{\lambda} \frac{\exp\left(\frac{\lambda}{\beta}A^\pi(s,a_i) - \lambda M_{ij}\right)\cdot \pi(a_j|s)}{\sum_{k=1}^{N}\exp\left(\frac{\lambda}{\beta}A^\pi(s,a_k) - \lambda M_{kj}\right)} \Big\}.\end{aligned} \tag{9}$$

Moreover, we have $\beta_\lambda^* \leq \frac{2A^{max}}{\delta}$.

In contrast to the Wasserstein dual formulation (6), the objective in the Sinkhorn dual formulation (9) is differentiable in β and admits closed-form gradients (shown in Appendix F). With this gradient information, we can use gradient-based global optimization algorithms (Wales & Doye, 1998; Zhan et al., 2006; Leary, 2000) to find a global optimal solution β_λ^* to (9).

Next, we show that if the entropic regularization parameter λ is large enough, then the optimal solution β_λ^* is a close approximation to the optimal solution β^* to the Wasserstein dual formulation. Proof is provided in Appendix G.

Theorem 3. *Define $\beta_{UB} = \max\{\frac{2A^{max}}{\delta}, \bar{\beta}\}$. We have:*

1. *$F_\lambda(\beta)$ converges to $F(\beta)$ uniformly on $[0, \beta_{UB}]$ with a convergence rate $\frac{1}{\lambda}$,*

2. $\lim_{\lambda \to \infty} argmin_{0 \leq \beta \leq \beta_{UB}} F_\lambda(\beta) \subseteq argmin_{0 \leq \beta \leq \beta_{UB}} F(\beta)$.

Although it is difficult to obtain the exact value of the optimal solution β^* to the Wasserstein dual formulation (6), the above theorem suggests that we can approximate β^* via β^*_λ by setting up a relative large λ. In practice, we can also adopt a smooth homotopy approach by setting an increasing sequence λ_k for each iteration and letting $\lambda_k \to \infty$.

SPO Policy Update: Based on Theorem 2, we introduce the following SPO policy updating rule:

$$\pi_{k+1}(a_i|s) = \mathbb{F}^{\text{SPO}}(\pi_k) = \sum_{j=1}^{N} \frac{\exp\left(\frac{\lambda_k}{\beta_k} A^{\pi_k}(s, a_i) - \lambda_k M_{ij}\right)}{\sum_{l=1}^{N} \exp\left(\frac{\lambda_k}{\beta_k} A^{\pi_k}(s, a_l) - \lambda_k M_{lj}\right)} \pi_k(a_j|s). \tag{SPO}$$

Here $\lambda_k \geq 0$ and $\beta_k \geq 0$ are some control parameters. The parameter β_k can be either computed via solving the one-dimensional subproblem (9) or simply set as a diminishing sequence. The proper setup of λ_k can effectively adjust the trade-off between convergence speed and final performance. More details are provided in the ablation study in Section 7.

5 Theoretical Analysis

We first show that SPO policy update converges to WPO policy update as the regularization parameter increases (i.e., $\lambda \to \infty$). The detailed proof is provided in Appendix H.

Lemma 1. *As $\lambda_k \to \infty$, SPO update converges to WPO update:* $\lim_{\lambda_k \to \infty} \mathbb{F}^{SPO}(\pi_k) \in \mathbb{F}^{WPO}(\pi_k)$.

We then provide a theoretical justification that WPO policy update (and SPO with $\lambda \to \infty$) are always guaranteed to improve the true performance J monotonically if we have access to the true advantage function. If the advantage function can only be evaluated inexactly with limited samples, then an extra estimation error (measured by the largest absolute entry $\|\cdot\|_\infty$) will occur. Proof can be found in Appendix I.

Theorem 4. ***(Performance improvement)*** *For any initial state distribution μ and any $\beta_k \geq 0$, if $||\hat{A}^\pi - A^\pi||_\infty \leq \epsilon$ for some $\epsilon > 0$, let $\hat{\mathcal{K}}_s^{\pi_k}(\beta_k, j) = argmax_{i=1,\dots,N}\{\hat{A}^{\pi_k}(s, a_i) - \beta_k M_{ij}\}$, WPO policy update (and SPO with $\lambda \to \infty$) guarantee the following performance improvement bound when the inaccurate advantage function $\hat{A}^\pi$ is used,*

$$J(\pi_{k+1}) \geq J(\pi_k) + \beta_k \mathbb{E}_{s \sim \rho_\mu^{\pi_{k+1}}} \sum_{j=1}^{N} \pi_k(a_j|s) \sum_{i \in \hat{\mathcal{K}}_s^{\pi_k}(\beta_k, j)} f_s^k(i, j) M_{ij} - \frac{2\epsilon}{1-\gamma}. \tag{10}$$

In the following, we show that with a decreasing schedule of the multiplier β_k, both WPO and SPO policy updates have their values $J(\pi_k)$ converging to the optimal $J^\star = \max_\pi J(\pi)$ on the tabular domain. To start, for k-th iteration, we consider (WPO) and (SPO) (with arbitrary $\lambda > 0$) whose updates π_{k+1} are optimal solutions to (7) with d being d_{W} and d_{S} respectively.

Assumption 2. *The state space and the action space are both finite, the reward function r is non-negative, and the initial distribution covers all state.*

Note that under the assumption that both state and action spaces are finite, the reward can be assumed non-negative without loss of generality, as we can always add $\max_{s,a}|r(s,a)|$ to the reward function to make it non-negative without changing the optimal policy and the order of the policies. Defining the optimal value function $V^\star(s) = \max_\pi \mathbb{E}[R_t|s_t = s]$, we have the following theorem. The proof is provided in Appendix J.

Theorem 5. ***(Global convergence)*** *Under Assumption 2, we have for any $\beta_k \geq 0$, (WPO) satisfies that*

$$\|V^\star - V^{\pi_{k+1}}\|_\infty \leq \gamma \|V^\star - V^{\pi_k}\|_\infty + \beta_k \|M\|_\infty, \tag{11}$$

and (SPO) satisfies that

$$\|V^\star - V^{\pi_{k+1}}\|_\infty \leq \gamma \|V^\star - V^{\pi_k}\|_\infty + 2\frac{\beta_k}{1-\gamma}\left(\|M\|_\infty + 2\frac{\log N}{\lambda}\right). \tag{12}$$

If $\lim_{k\to\infty}\beta_k = 0$*, we further have* $\lim_{k\to\infty} J(\pi_k) = J^\star$*.*

Remark 2. *Note the convergence is* geometric*. If we keep* β_k *as a constant, then* $0 \leq J^\star - J(\pi^T) \leq \|V^\star - V^{\pi^T}\|_\infty \leq \gamma^T \|V^\star - V^{\pi_0}\|_\infty + \frac{\beta B}{1-\gamma}$*, where* $B = \|M\|_\infty$ *for (WPO) and* $B = 2\frac{\|M\|_\infty + 2\frac{\log N}{\lambda}}{1-\gamma}$ *for (SPO). To achieve an* ϵ *optimality gap, we only need to take* $\beta = \frac{(1-\gamma)\epsilon}{2B}$ *and let* $T \geq \frac{\log(\epsilon/2)}{\gamma}$*.*

Remark 3. *The study of global non-asymptotic convergence of nonconvex policy optimization algorithms has been an active research topic. Recent theoretical work has mostly centered on PG and natural policy gradient (NPG) - a close relative of TRPO; see e.g., Agarwal et al. (2021a); Cen et al. (2021); Lan (2022). To our best knowledge, few work has discussed the global convergence of TRPO. Neu et al. (2017) and Geist et al. (2019) established the connection of TRPO to Mirror Descent, but did not provide any non-asymptotic rate; Shani et al. (2020) showed that adaptive TRPO with decaying stepsize achieved* $O(1/\sqrt{T})$ *convergence rate for unregularized MDPs in the tabular setting (finite state and finite action). Our result seems to be the first non-asymptotic analysis of policy optimization based on Wasserstein and Sinkhorn divergence. It remains interesting to extend the convergence theory of TRPO/WPO/SPO to function approximation regime following recent advance Agarwal et al. (2021a). However, this is beyond the scope of our current work, as we focus on explicit close-form update of WPO/SPO, which can be a viable alternative to TRPO in practice.*

6 A Practical Algorithm

In practice, the advantage value functions are often estimated from sampled trajectories. In this section, we provide a practical on-policy actor-critic algorithm, described in Algorithm 1, that combines WPO/SPO with advantage function estimation.

At each iteration, the first step is to collect trajectories, which can be either complete or partial. If the trajectory is complete, the total return can be directly expressed as the accumulated discounted rewards $R_t = \sum_{k=0}^{T-t-1}\gamma^k r_{t+k}$. If the trajectory is partial, it can be estimated by applying the multi-step temporal difference (TD) methods (De Asis et al., 2017): $\hat{R}_{t:t+n} = \sum_{k=0}^{n-1}\gamma^k r_{t+k} + \gamma^n V(s_{t+n})$. Then for the advantage estimation, we can use Monte Carlo advantage estimation, i.e., $\hat{A}_t^{\pi_k} = R_t - V_{\psi_k}(s_t)$ or Generalized Advantage Estimation (GAE) (Schulman et al., 2016), which provides a more explicit control over the bias-variance trade-off. In the value update step, we use a neural net to represent the value function, where ψ is the parameter that specifies the value net $s \to V(s)$. Then, we can update ψ by using gradient descent, which significantly reduces the computational burden of computing advantage directly. The computational complexity of the algorithm is discussed in Appendix K.

Algorithm 1: On-policy WPO/SPO algorithm

Input: number of iterations K, learning rate α
Initialize policy π_0 and value network V_{ψ_0} with random parameter ψ_0
for $k = 0, 1, 2 \ldots K$ **do**
    Collect trajectory set $\mathcal{D}_k$ on policy π_k
    For each timestep t in each trajectory, compute total returns G_t and estimate advantages $\hat{A}_t^{\pi_k}$
    Update value:
    $\psi_{k+1} \leftarrow \psi_k - \alpha \nabla_{\psi_k} \sum (G_t - V_{\psi_k}(s_t))^2$
    Update policy:
    $\pi_{k+1} \leftarrow \mathbb{F}(\pi_k)$ via WPO/ SPO with $\hat{A}_t^{\pi_k}$
end

7 Experiments

In this section, we evaluate the proposed WPO and SPO approaches presented in Algorithm 1. We compare the performance of our methods with benchmarks including TRPO (Schulman et al., 2015), PPO (Schulman et al., 2017), A2C (Mnih et al., 2016); and with BGPG (Pacchiano et al., 2020), WNPG (Moskovitz et al., 2021) for continuous control. The code of our WPO/SPO can be found here[1]. We adopt the implementations of TRPO, PPO and A2C from OpenAI Baselines (Dhariwal et al., 2017) for MuJuCo tasks and Stable Baselines (Hill et al., 2018) for other tasks. For BGPG, we adopt the same implementation[2] as (Pacchiano et al., 2020).

Our experiments include (1) ablation study that focuses on sensitivity analysis of WPO and SPO; (2) tabular domain tasks with discrete state and action including the Taxi, Chain, and Cliff Walking environments; (3) locomotion tasks with continuous state and discrete action including the CartPole, Acrobot environments; (4) additional comparison of KL and Wasserstein trust regions under tabular domain and locomotion tasks; and (5) extension to continuous control tasks with continuous action including HalfCheetah, Hopper, Walker, and Ant environments from MuJuCo. See table 4 in Appendix A for a summary of performance. The setting of hyperparameters and network sizes of our algorithms and additional results are provided in Appendix A.

7.1 Ablation Study

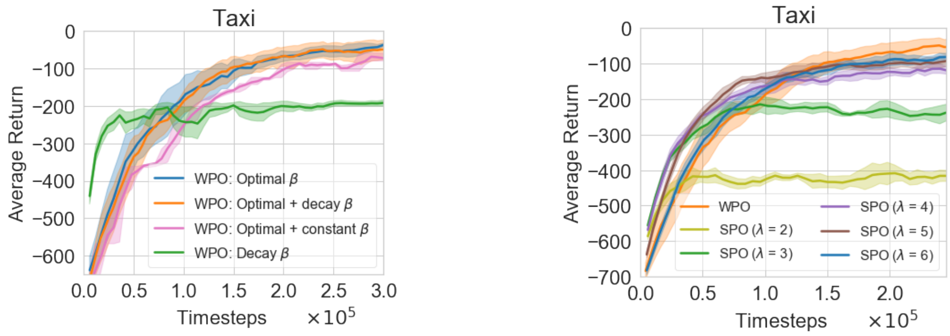
In this experiment, we first examine the sensitivity of WPO in terms of different strategies of β_k. We test four settings of β value for WPO policy update: (1) Setting 1: computing optimal β value for all policy update; (2) Setting 2: computing optimal β value for first 20% of policy updates and decaying β for the remaining; (3) Setting 3: computing optimal β value for first 20% of policy updates and fix β as its last updated value for the remaining; (4) Setting 4: decaying β for all policy updates (e.g., $\beta_k = \frac{1}{k^2}$). In particular, Setting 3 is rooted in the observation that β^* does not change significantly throughout all the policy updates, especially in the later stage in the experiments carried out in the paper. Small perturbations are added to the approximate values to avoid any stagnation in updating. Taxi task (Dietterich, 1998) from tabular domain is selected for this experiment.

Table 1: Run time for different β settings

Runtime	Taxi (s)	CartPole (s)
Setting 1 (optimal β)	1224	130
Setting 2 (optimal-then-decay)	648	63
Setting 3 (optimal-then-fix)	630	67
Setting 4 (decaying β)	522	44

The performance comparisons and average run times are shown in figure 4 and table 1 respectively. Figure 4a and table 1 clearly indicate a tradeoff between computation efficiency and accuracy in terms of different choices of β value. Setting 2 is the most effective way to balance the tradeoff between performance and run time. For the rest of experiments, we adopt this setting for both WPO and SPO. Figure 4b shows that as λ increases, the convergence becomes slower but the final performance of SPO improves and becomes closer to that of WPO, which verifies the convergence property of Sinkhorn to Wasserstein distance shown in Theorem 3. Therefore, the choice of λ can effectively adjust the trade-off between convergence and final performance. Similar results are observed when using time-varying λ on Taxi, Chain and CartPole tasks, presented in figure 9 in Appendix A.

[1] `https://github.com/efficientwpo/EfficientWPO`
[2] `https://github.com/behaviorguidedRL/BGRL`



(a) Different settings of β

(b) Different choices of λ

Figure 4: Episode rewards for Taxi with different β and λ settings, averaged across 5 runs with random initialization.

7.2 Tabular Domains

We evaluate WPO and SPO on tabular domain tasks and test the exploration ability of the algorithms on several environments including Taxi, Chain, and Cliff Walking. We use a table of size $|\mathcal{S}| \times |\mathcal{A}|$ to represent the policy $\pi(a|s)$. For the value function, we use a neural net to smoothly update the values. The performance of WPO and SPO are compared to the performance of TRPO, PPO and A2C under the same neural net structure. Results on Taxi, Cliff and Chain are reported in figure 5.

As shown in figure 5, the performances of WPO, SPO and TRPO are manifestly better than A2C and PPO. Among the trust region based methods, WPO and SPO outperform TRPO in Taxi (and also in Cliff Walking), whereas in Chain, the performances of these three methods are comparable.

As further shown in table 2, for the Taxi environment, WPO has a higher successful drop-off rate and a lower task completion time while the original TRPO reaches the time limit with a drop-off rate 0, suggesting that WPO finds a better policy than the original TRPO. In figure 7, we also compare the performance of WPO under Wasserstein and KL divergences given different number of samples N_A used to estimate the advantage function, and the result suggests that using Wasserstein metric is more robust than KL divergence under inaccurate advantage values.

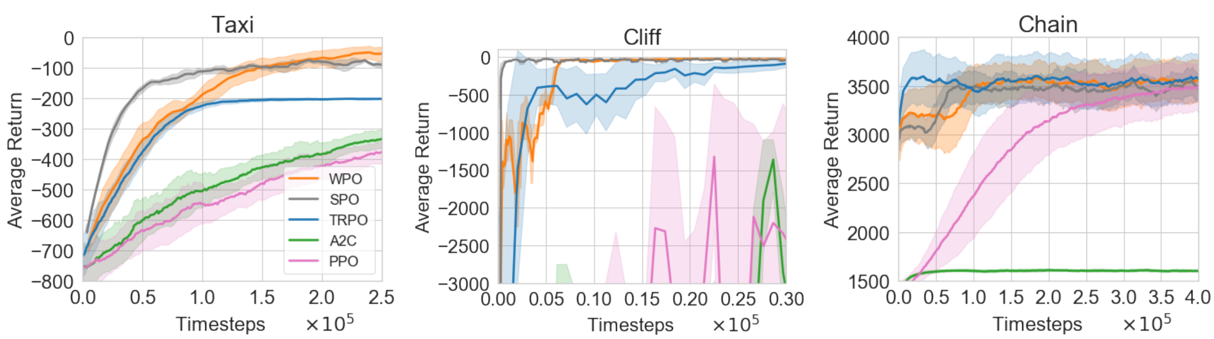


Figure 5: Episode rewards during training for tabular domain tasks, averaged across 5 runs with random initialization.

Table 2: Trained agents performance on Taxi

	Success (+20)	Fail (-10)	Steps (-1)	Return
WPO	0.753	0.232	70.891	-58.151
TRPO	0	0	200	-200

7.3 Robotic Locomotion Tasks

We now integrate deep neural network architecture into WPO and SPO and evaluate their performance on several locomotion tasks (with continuous state and discrete action), including CartPole (Barto et al., 1983) and Acrobot (Geramifard et al., 2015). We use two separate neural nets to represent the policy and the value. The policy neural net receives state s as an input and outputs the categorical distribution of $\pi(a|s)$. A random subset of states $\mathcal{S}_k \in \mathcal{S}$ is sampled at each iteration to perform policy updates.

Figure 6 shows that WPO and SPO outperform TRPO, PPO and A2C in most tasks in terms of final performance, except in Acrobot where PPO performs the best. In most cases, SPO converges faster but WPO has a better final performance. To train 10^5 timesteps in the discrete locomotion tasks, the training wall-clock time is around 63s for WPO, 65s for SPO, 59s for TRPO and 70s for PPO. Therefore, WPO has a similar computational efficiency as TRPO and PPO.

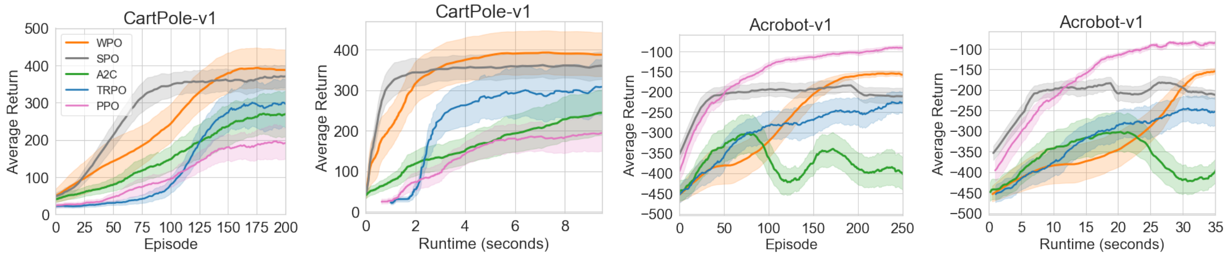
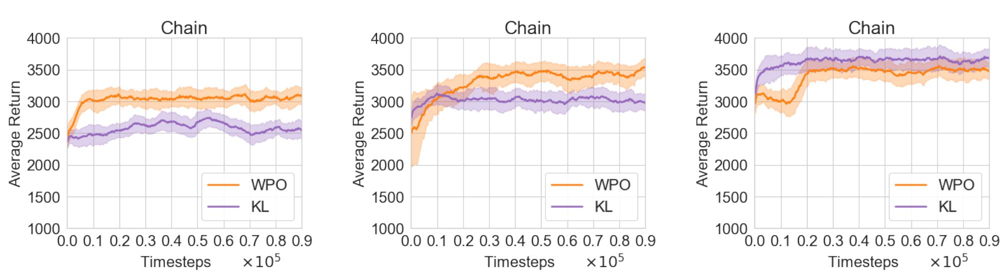


Figure 6: Episode rewards during the training process for the locomotion tasks, averaged across 5 runs with random initialization.

7.4 Additional Comparison of Wasserstein and KL trust regions

We also show that compared with the KL divergence, the utilization of Wasserstein metric can cope with the inaccurate advantage function estimations caused by the lack of samples. We further evaluate the performance of WPO under Wasserstein and KL divergences (as derived in Peng et al. (2019)) on the Chain task, given different number of samples N_A used to estimate the advantage function. As shown in figure 7, when N_A is 1000, KL performs slightly better than WPO. However, when N_A decreases to 100 or 250, WPO outperforms KL. Similar results are also obtained for the locomotion tasks (figure 10 in Appendix A). These results indicate that WPO is more robust than KL under inaccurate advantage values. This finding is coherent with our observations on the policy update formulations of Wasserstein and KL. For the Wasserstein update in (5), policy will be updated only when the advantage difference between two actions is significant, i.e., $A^\pi(s, a_j) - \beta M_{ij} \geq A^\pi(s, a_i)$. However, for the KL update in Peng et al. (2019), policy will be updated as long as the current advantage function has a single non-zero value. Therefore, KL update is more sensitive; while Wasserstein update is more robust and more tolerant to advantage inaccuracies.



(a) $N_A = 100$ (b) $N_A = 250$ (c) $N_A = 1000$

Figure 7: Episode rewards during training for the Chain task, where advantage value function is estimated under different number of samples.

7.5 Extension to Continuous Control

To extend to environments with continuous action, we use Implicit Quantile Networks (IQN) (Will Dabney & Munos, 2018) actor that can represent an arbitrary complex non-parametric policy. Let $F_s^{-1}(p)$ represent the quantile function associated with policy $\pi(\cdot|s)$. The IQN actor takes state s and probability $p \in [0, 1]$ as input, and outputs the corresponding quantile value $a = F_s^{-1}(p)$. IQN actor can be trained to approach pre-defined target policy distributions through quantile regression (Will Dabney & Munos, 2018; Tessler et al., 2019).

Define the action support for state s in k-th iteration as $I^{\pi_k}(s) = \{a : A^{\pi_k}(s,a) - \beta_k d(a,a') \geq \min_{a' \in I^{\pi_{k-1}}(s)} A^{\pi_k}(s,a')\}$. Then, the WPO/SPO target policy distribution to guide IQN update in the k-th iteration is:

$$D_{I^{\pi_k}(s)}(a|s) = \sum\nolimits_{a' \in I^{\pi_{k-1}}(s)} \pi_k(a'|s) f_s(a,a'), \tag{13}$$

where for WPO update $f_s(a,a') = 1$ if $a = \operatorname{argmax}_{a \in I^{\pi_k}(s)}\{A^{\pi_k}(s,a) - \beta_k d(a,a')\}$ and $f_s(a,a') = 0$ otherwise; for SPO update, $f_s(a,a') = \frac{\exp(\frac{\lambda_k}{\beta_k} A^{\pi_k}(s,a) - \lambda_k d(a,a'))}{\sum_{a \in I^{\pi_k}(s)} \exp(\frac{\lambda_k}{\beta_k} A^{\pi_k}(s,a) - \lambda_k d(a,a'))}$. In implementation, we sample a batch of states $\mathcal{S}_k \in \mathcal{S}$ at each iteration to perform policy updates, and for each $s \in \mathcal{S}_k$, we sample $|\mathcal{A}_k|$ actions to approximate the support $I^{\pi_k}(s)$ and the target policy distribution $D_{I^{\pi_k}(s)}(\cdot|s)$.

We additionally compare WPO and SPO with WNPG (Moskovitz et al., 2021) and BGPG (Pacchiano et al., 2020) that are specially designed to address the continuous control with Wasserstein metric, for several MuJuCo tasks including HalfCheetah, Hopper, Walker, and Ant. Figure 8 shows that WPO and SPO have consistently better performances than other benchmarks.

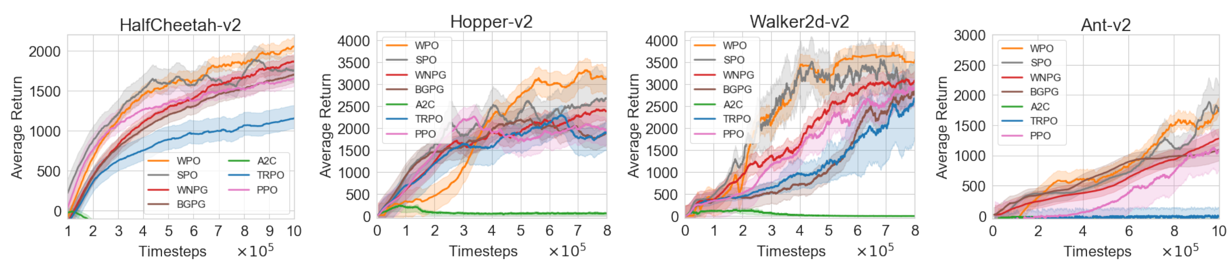


Figure 8: Episode rewards during training for MuJuCo continuous control tasks, averaged across 5 runs with random initialization.

8 Conclusion

In this paper, we present two policy optimization frameworks, WPO and SPO, which can exactly characterize the policy updates instead of confining their distributions to particular distribution class or requiring any approximation. Our methods outperform TRPO and PPO with better sample efficiency, faster convergence, and improved final performance. Our numerical results show that the Wasserstein metric is more robust to the ambiguity of advantage functions, compared with the KL divergence. Our strategy for adjusting β value for WPO can reduce the computational time and boost the convergence without noticeable performance degradation. SPO improves the convergence speed of WPO by properly choosing the weight of the entropic regularizer. Performance improvement and global convergence for WPO are discussed. For future work, it remains interesting to extend the idea to PPO and natural policy gradients, which penalize the policy update instead of imposing trust region constraint, and extend it to off-policy frameworks.

References

Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation. *ArXiv Preprint*, pp. arXiv:1806.06920, 2018.

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *J. Mach. Learn. Res.*, 22(98):1–76, 2021a.

Rishabh Agarwal, Marlos C Machado, Pablo Samuel Castro, and Marc G Bellemare. Contrastive behavioral similarity embeddings for generalization in reinforcement learning. In *Proceedings of the 9th International Conference on Learning Representations*, 2021b.

Andrew G. Barto, Richard S. Sutton, and Charles W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(5): 834–846, 1983.

Jose Blanchet and Karthyek Murthy. Quantifying distributional model risk via optimal transport. *Mathematics of Operations Research*, 44(2):565–600, 2019.

Pablo Samuel Castro. Scalable methods for computing state similarity in deterministic Markov decision processes. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, volume 34, pp. 10069–10076, 2020.

Shicong Cen, Chen Cheng, Yuxin Chen, Yuting Wei, and Yuejie Chi. Fast global convergence of natural policy gradient methods with entropy regularization. *Operations Research*, 2021.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games.* Cambridge University Press, 2006.

Nicolas Courty, Rémi Flamary, and Devis Tuia. Domain adaptation with regularized optimal transport. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 274–289. Springer, 2014.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1853–1865, 2016.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, pp. 2292–2300, 2013.

Robert Dadashi, Léonard Hussenot, Matthieu Geist, and Olivier Pietquin. Primal Wasserstein imitation learning. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

Kristopher De Asis, J. Fernando Hernandez-Garcia, G. Zacharias Holland, and Richard S. Sutton. Multi-step reinforcement learning: A unifying algorithm. *ArXiv Preprint*, pp. arXiv:1703.01327, 2017.

Prafulla Dhariwal, Christopher Hesse, Oleg Klimov, Alex Nichol, Matthias Plappert, Alec Radford, John Schulman, Szymon Sidor, Yuhuai Wu, and Peter Zhokhov. OpenAI baselines. `https://github.com/openai/baselines`, 2017.

Thomas G. Dietterich. The MAXQ method for hierarchical reinforcement learning. In *Proceedings of the 15th International Conference on Machine Learning*, pp. 118–126, 1998.

Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *Proceedings of the 33rd International Conference on Machine Learning*, pp. 1329–1338, 2016.

Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171 (1):115–166, 2018.

Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite Markov decision processes. In *Uncertainty in Artificial Intelligence*, volume 4, pp. 162–169, 2004.

Rui Gao and Anton J Kleywegt. Distributionally robust stochastic optimization with Wasserstein distance. *arXiv preprint arXiv:1604.02199*, 2016.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. In *International Conference on Machine Learning*, pp. 2160–2169. PMLR, 2019.

Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with Sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pp. 1608–1617, 2018.

Alborz Geramifard, Christoph Dann, Robert H. Klein, William Dabney, and Jonathan P. How. RLPy: A value-function-based reinforcement learning framework for education and research. *Journal of Machine Learning Research*, 16(46):1573–1578, 2015.

Gregory Z. Grudic, Vijay Kumar, and Lyle H. Ungar. Using policy gradient reinforcement learning on autonomous robot controllers. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 406–411, 2003.

Shixiang Gu, Ethan Holly, Timothy Lillicrap, and Sergey Levine. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *Proceedings of the 2017 IEEE International Conference on Robotics and Automation*, pp. 3389–3396, 2017.

Shuncheng He, Yuhang Jiang, Hongchang Zhang, Jianzhun Shao, and Xiangyang Ji. Wasserstein Unsupervised Reinforcement Learning. In *Proceedings of the Thirty-Sixth AAAI Conference on Artificial Intelligence*. AAAI Press, 2022.

Johannes Heinrich and David Silver. Deep reinforcement learning from self-play in imperfect-information games. *ArXiv Preprint*, pp. arXiv:1603.01121, 2016.

Ashley Hill, Antonin Raffin, Maximilian Ernestus, Adam Gleave, Anssi Kanervisto, Rene Traore, Prafulla Dhariwal, Christopher Hesse, Oleg Klimov, Alex Nichol, Matthias Plappert, Alec Radford, John Schulman, Szymon Sidor, and Yuhuai Wu. Stable baselines. `https://github.com/hill-a/stable-baselines`, 2018.

Minhui Huang, Shiqian Ma, and Lifeng Lai. A Riemannian block coordinate descent method for computing the projection robust Wasserstein distance. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 4446–4455, 2021.

Anatoli Juditsky, Philippe Rigollet, and Alexandre B Tsybakov. Learning by mirror averaging. *The Annals of Statistics*, 36(5):2183–2206, 2008.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the 19th International Conference on Machine Learning*, pp. 267–274, 2002.

Sham M Kakade. A natural policy gradient. In *Advances in Neural Information Processing Systems*, volume 14, 2001.

Daniel Kuhn, Peyman Mohajerin Esfahani, Viet Anh Nguyen, and Soroosh Shafieezadeh-Abadeh. Wasserstein distributionally robust optimization: Theory and applications in machine learning. In *Operations Research & Management Science in the Age of Analytics*, pp. 130–166. INFORMS, 2019.

Guanghui Lan. Policy mirror descent for reinforcement learning: Linear convergence, new sampling complexity, and generalized problem classes. *Mathematical programming*, pp. 1–48, 2022.

Robert Leary. Global optimization on funneling landscapes. *Journal of Global Optimization*, 18:367–383, 2000.

Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In *Proceedings of the 4th International Conference on Learning Representations*, 2016.

Tianyi Lin, Chenyou Fan, Nhat Ho, Marco Cuturi, and Michael I Jordan. Projection robust Wasserstein distance and Riemannian optimization. *ArXiv Preprint*, pp. arXiv:2006.07458, 2020.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with deep reinforcement learning. *ArXiv Preprint*, pp. arXiv:1312.5602, 2013.

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

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning*, pp. 1928–1937, 2016.

Ted Moskovitz, Michael Arbel, Ferenc Huszar, and Arthur Gretton. Efficient Wasserstein natural gradients for reinforcement learning. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

Gergely Neu, Anders Jonsson, and Vicenç Gómez. A unified view of entropy-regularized markov decision processes. *arXiv preprint arXiv:1705.07798*, 2017.

Aldo Pacchiano, Jack Parker-Holder, Yunhao Tang, Krzysztof Choromanski, Anna Choromanska, and Michael Jordan. Learning to score behaviors for guided policy optimization. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 7445–7454, 2020.

Victor M. Panaretos and Yoav Zemel. Statistical aspects of wasserstein distances. *Annual Review of Statistics and Its Application*, 6(1):405–431, 2019.

Giorgio Patrini, Rianne van den Berg, Patrick Forre, Marcello Carioni, Samarth Bhargav, Max Welling, Tim Genewein, and Frank Nielsen. Sinkhorn autoencoders. In *Uncertainty in Artificial Intelligence*, pp. 733–743, 2019.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *ArXiv Preprint*, pp. arXiv:1910.00177, 2019.

Jan Peters and Stefan Schaal. Policy gradient methods for robotics. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2219–2225, 2006.

Henning Petzka, Asja Fischer, and Denis Lukovnicov. On the regularization of Wasserstein GANs. *ArXiv Preprint*, pp. arXiv:1709.08894, 2017.

Pierre H. Richemond and Brendan Maginnis. On Wasserstein reinforcement learning and the Fokker-Planck equation. *ArXiv Preprint*, pp. arXiv:1712.07185, 2017.

Philippe Rigollet and Alexandre Tsybakov. Exponential screening and optimal rates of sparse estimation. *The Annals of Statistics*, 39(2):731–771, 2011.

R. Tyrrell Rockafellar and Roger J. B. Wets. *Variational Analysis*. Springer, 1998.

Johannes O. Royset. Approximations and solution estimates in optimization. *Mathematical Programming*, 170:479–506, 2018.

John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 1889–1897, 2015.

John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *Proceedings of the 4th International Conference on Learning Representations*, 2016.

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

Lior Shani, Yonathan Efroni, and Shie Mannor. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 5668–5675, 2020.

David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *Proceedings of the 31st International Conference on Machine Learning*, pp. 387–395, 2014.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Vedavyas Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P. Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

Maurice Sion. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171–176, 1958.

Richard S. Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, pp. 1057–1063, 1999.

Chen Tessler, Guy Tennenholtz, and Shie Mannor. Distributional policy optimization: An alternative approach for continuous control. In *Advances in Neural Information Processing Systems*, pp. 1350–1360, 2019.

David Wales and Jonathan Doye. Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. *The Journal of Physical Chemistry A*, 101(28): 5111–5116, 1998.

Jie Wang, Rui Gao, and Yao Xie. Sinkhorn distributionally robust optimization. *ArXiv Preprint*, pp. arXiv:2109.11926, 2021a.

Jie Wang, Rui Gao, and Yao Xie. Two-sample test using projected Wasserstein distance. In *Proceedings of IEEE International Symposium on Information Theory*, volume 21, 2021b.

David Silver Will Dabney, Georg Ostrovski and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International Conference on Machine Learning*, 2018.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, 1992.

Huang Xiao, Michael Herman, Joerg Wagner, Sebastian Ziesche, Jalal Etesami, and Thai Hong Linh. Wasserstein adversarial imitation learning. *ArXiv Preprint*, pp. arXiv:1906.08113, 2019.

Lixin Zhan, Jeff Chen, and Wing-Ki Liu. Monte Carlo basin paving: An improved global optimization method. *Physical Review E*, 73:015701, 2006.

Ruiyi Zhang, Changyou Chen, Chunyuan Li, and Lawrence Carin. Policy optimization as Wasserstein gradient flows. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 5737–5746, 2018.

Chaoyue Zhao and Yongpei Guan. Data-driven risk-averse stochastic optimization with Wasserstein metric. *Operations Research Letters*, 46(2):262–267, 2018.