AVERAGE REWARD REINFORCEMENT LEARNING WITH MONOTONIC POLICY IMPROVEMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

In continuing control tasks, an agent's average reward per time step is a more natural performance measure compared to the commonly used discounting framework as it can better capture an agent's long-term behavior. We derive a novel lower bound on the difference of the average rewards for two policies, where the lower bound depends on the average divergence between the policies. We show that previous work based on the discounted return (Schulman et al., 2015; Achiam et al., 2017) result in a trivial lower bound in the average reward setting. We develop an iterative procedure based on our lower bound which produces a sequence of monotonically improved policies for the average reward criterion. When combined with deep reinforcement learning methods, the procedure leads to scalable and efficient algorithms aimed at maximizing an agent's average reward performance. Empirically, we demonstrate the efficacy of our algorithms through a series of high-dimensional control tasks with long time horizons and show that discounting can lead to unsatisfactory performance on continuing control tasks.

1 INTRODUCTION

The goal of Reinforcement Learning (RL) is to build agents that can learn high-performing behaviors through trial-and-error interactions with the environment. Broadly speaking, modern RL tackles two kinds of problems: *episodic tasks* and *continuing tasks*. In episodic tasks, the agent-environment interaction can be broken into separate distinct episodes. The long term behavior of the agent can be easily measured in terms of the total sum of some predefined reward function. Examples of episodic tasks include training an agent to learn to play Go (Silver et al., 2016; 2018) or Atari video games (Mnih et al., 2013), where the episode terminates when the game ends. In continuing tasks, such as controlling robots with long operating lifespans (Peters & Schaal, 2008; Schulman et al., 2015; Haarnoja et al., 2018), there is no natural separation of episodes and the agent-environment interaction continues indefinitely. The performance of an agent in a continuing task is more difficult to quantify since even for bounded reward functions, the total sum of rewards is typically infinite.

One way of making the long-term reward objective meaningful for continuing tasks is to apply *discounting*, i.e., we maximize the discounted sum of rewards $r_0 + \gamma r_1 + \gamma^2 r_2 + \cdots$ for some discount factor $\gamma \in (0, 1)$, which is guaranteed to be finite for any bounded reward function. However the discounted objective gives higher importance to more recent experience of the agent. Such an objective —while useful in certain applications —cannot give an accurate description of an agent's long-term behavior. As argued in Chapter 10 of Sutton & Barto (2018) and in Naik et al. (2019), a more natural objective is to use the average reward received by an agent over every time-step. While the average reward setting has been extensively studied in the classical Markov Decision Process literature (Howard, 1960; Blackwell, 1962; Veinott, 1966; Bertsekas et al., 1995; Puterman, 1994), it is much less commonly used in reinforcement learning. An important open question is whether recent advances in RL for discounted reward can be naturally generalized to the average reward setting.

One major source of difficulty with modern RL algorithms lies in controlling the step-size for policy updates. In order to have better control over step-sizes, Schulman et al. (2015) constructed a lower bound on the difference between the expected discounted return for two arbitrary policies π and π'. The bound is based on the divergence between these two policies. Schulman et al. (2015) showed that iteratively maximizing this lower bound generates a sequence of monotonically improved policies in terms of their discounted return.

In this paper, we will show that the policy improvement theorem from Schulman et al. (2015) results in a trivial bound in the average reward case. We derive a novel result which lower bounds the difference of the average rewards based on the divergence of the policies. We show that iteratively maximizing this lower bound guarantees monotonic average reward policy improvement. Similar to the discounted case, the problem of maximizing the lower bound can be approximated with scalable algorithms which can be optimized using samples collected in the environment. We describe in detail two such algorithms: Average Reward TRPO (ATRPO) and Average Cost CPO (ACPO) which are average reward versions of algorithms based on their respective discounted variants (Schulman et al., 2015; Achiam et al., 2017). We further show that these algorithms are more effective than their discounted counterparts for continuing control tasks.

2 PRELIMINARIES

Consider a Markov Decision Process (MDP) (Sutton & Barto, 2018) $(\mathcal{S}, \mathcal{A}, P, r, \mu)$ where the state space $\mathcal{S}$ and action space $\mathcal{A}$ are assumed to be finite. The transition probability is denoted by $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$, the bounded reward function $r : \mathcal{S} \times \mathcal{A} \to [r_{\min}, r_{\max}]$, and $\mu : \mathcal{S} \to [0, 1]$ is the initial state distribution. Let $\pi = \{\pi(a|s) : s \in \mathcal{S}, a \in \mathcal{A}\}$ be a stationary policy, and Π is the set of all stationary policies. Here we discuss the two objective formulations for continuing control tasks: the average reward approach and discounted reward approach.

Average Reward Approach
In this paper, we will focus exclusively on *unichain* MDPs, which is when the Markov chain corresponding to every policy contains only one recurrent class and a finite but possibly empty set of transient states. The average reward objective is defined as:

$$\rho(\pi) := \lim_{N\to\infty} \frac{1}{N} \mathop{\mathbb{E}}_{\tau\sim\pi} \left[\sum_{t=0}^{N-1} r(s_t, a_t) \right] = \mathop{\mathbb{E}}_{\substack{s\sim d_\pi \\ a\sim\pi}} [r(s, a)]. \tag{1}$$

Here $d_\pi(s) := \lim_{N\to\infty} \frac{1}{N} \sum_{t=0}^{N-1} P(s_t = s|\pi) = \lim_{t\to\infty} P(s_t = s|\pi)$ is the *stationary state distribution under policy* π, $\tau = (s_0, a_0, \dots,)$ is a sample trajectory. We use $\tau \sim \pi$ to indicate that the trajectory is sampled from policy π, i.e. $s_0 \sim \mu$, $a_t \sim \pi(\cdot|s_t)$, and $s_{t+1} \sim P(\cdot|s_t, a_t)$. In the unichain case, the average reward $\rho(\pi)$ is state-independent for any policy π (Bertsekas et al., 1995). We express the *average-reward value function* as $V^\pi(s) := \mathbb{E}_{\tau\sim\pi} \left[\sum_{t=0}^{\infty} (r(s_t, a_t) - \rho(\pi)) \middle| s_0 = s \right]$ and *action-value function* as $Q^\pi(s, a) := \mathbb{E}_{\tau\sim\pi} \left[\sum_{t=0}^{\infty} (r(s_t, a_t) - \rho(\pi)) \middle| s_0 = s, a_0 = a \right]$. We define the *average reward advantage function* as $A^\pi(s, a) := Q^\pi(s, a) - V^\pi(s)$.

Discounted Reward Approach
For some discount factor $\gamma \in (0, 1)$, the discounted reward objective is defined as

$$\rho_\gamma(\pi) := \mathop{\mathbb{E}}_{\tau\sim\pi} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] = \frac{1}{1-\gamma} \mathop{\mathbb{E}}_{\substack{s\sim d_{\pi,\gamma} \\ a\sim\pi}} [r(s, a)]. \tag{2}$$

where $d_{\pi,\gamma}(s) := (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t P(s_t = s|\pi)$ is known as the *future discounted state visitation distribution under policy* π. Note that unlike the average reward objective, *the discounted objective depends on the initial state distribution* μ. It can be easily shown that $d_{\pi,\gamma}(s) \to d_\pi(s)$ for all s as $\gamma \to 1$. The *discounted value function* is defined as $V^\pi_\gamma(s) := \mathbb{E}_{\tau\sim\pi} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \middle| s_0 = s \right]$ and *discounted action-value function* $Q^\pi_\gamma(s, a) := \mathbb{E}_{\tau\sim\pi} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \middle| s_0 = s, a_0 = a \right]$. Finally, the *discounted advantage function* is defined as $A^\pi_\gamma(s, a) := Q^\pi_\gamma(s, a) - V^\pi_\gamma(s)$.

It is well-known that $\lim_{\gamma\to 1}(1 - \gamma)\rho_\gamma(\pi) = \rho(\pi)$, implying that the discounted and average reward objectives are equivalent in the limit as γ approaches 1 (Blackwell, 1962). We will further discuss the relationship between the discounted and average reward value functions in the supplementary materials and prove that $\lim_{\gamma\to 1} A^\pi_\gamma(s, a) = A^\pi(s, a)$ (see Corollary A.1).

3 MONTONICALLY IMPROVEMENT GUARANTEES FOR DISCOUNTED RL

In many modern RL literature (Schulman et al., 2015; 2017; Abdolmaleki et al., 2018; Vuong et al., 2019), algorithms iteratively update policies within a local region, i.e., at iteration k we find policy π_{k+1} by maximizing $\rho_\gamma(\pi)$ within some region $D(\pi, \pi_k) \leq \delta$ for some divergence measure D. This approach allows us to control the step-size of each update using different choices of D and δ which can lead to better sample efficiency (Peters & Schaal, 2008). Schulman et al. (2015) derived a policy improvement bound based on a specific choice of D:

$$\rho_\gamma(\pi_{k+1}) - \rho_\gamma(\pi_k) \geq \frac{1}{1-\gamma} \mathop{\mathbb{E}}_{\substack{s\sim d_{\pi_k,\gamma} \\ a\sim\pi_{k+1}}} [A_\gamma^{\pi_k}(s,a)] - C \cdot \max_s [D_{\text{TV}}(\pi_{k+1} \parallel \pi_k)[s]] \tag{3}$$

where $D_{\text{TV}}(\pi' \parallel \pi)[s] := \frac{1}{2}\sum_a |\pi'(a|s) - \pi(a|s)|$ is the *total variation divergence* for policies π and π', and C is some constant which does not depend on the divergence term D_{TV}. Schulman et al. (2015) showed that by choosing π_{k+1} such that the right hand side of (3) is maximized, we are guaranteed to have $\rho_\gamma(\pi_{k+1}) \geq \rho_\gamma(\pi_k)$. This provided the theoretical foundation for an entire class of scalable policy optimization algorithms based on efficiently maximizing the right-hand-side of (3) (Schulman et al., 2015; 2017; Wu et al., 2017; Abdolmaleki et al., 2018; Vuong et al., 2019).

A natural question arises here is whether the iterative procedure described by Schulman et al. (2015) also guarantees improvement w.r.t. the average reward. Since the discounted and average reward objectives are equivalent when $\gamma \to 1$, one may assume that we can also lower bound the policy performance difference of the average reward objective by letting $\gamma \to 1$ for the bounds in Schulman et al. (2015). Unfortunately this results in a trivial bound. We will demonstrate this through a similar policy improvement bound from Achiam et al. (2017) based on the average divergence but a similar argument can be made for the original bound from Schulman et al. (2015) (see supplementary material for proof and discussion).

Proposition 1. *Consider the following bound from Achiam et al. (2017)*

$$D^-_{\pi,\gamma}(\pi') \leq \rho_\gamma(\pi') - \rho_\gamma(\pi) \leq D^+_{\pi,\gamma}(\pi') \tag{4}$$

where

$$D^\pm_{\pi,\gamma}(\pi') = \frac{1}{1-\gamma} \mathop{\mathbb{E}}_{\substack{s\sim d_\pi \\ a\sim\pi}} \left[\frac{\pi'(a|s)}{\pi(a|s)} A_\gamma^\pi(s,a)\right] \pm \frac{2\gamma\epsilon_\gamma}{(1-\gamma)^2} \mathop{\mathbb{E}}_{s\sim d_\pi} [D_{TV}(\pi' \parallel \pi)[s]]$$

and $\epsilon_\gamma = \max_s \left|\mathbb{E}_{a\sim\pi'}[A_\gamma^\pi(s,a)]\right|$. *We have:*

$$\lim_{\gamma\to 1} (1-\gamma) D^\pm_{\pi,\gamma}(\pi') = \pm\infty \tag{5}$$

Since $\lim_{\gamma\to 1}(1-\gamma)(\rho_\gamma(\pi') - \rho_\gamma(\pi)) = \rho(\pi') - \rho(\pi)$, Proposition 1 says (4) becomes trivial when used on the average reward. This result is discouraging as it shows that the policy improvement guarantee from Schulman et al. (2015) does not appear to generalize to the average reward setting. In the next section, we will derive an alternative policy improvement bound for the average reward objective which can be used to generate monotonically improved policies w.r.t. the average reward.

4 MAIN RESULTS

4.1 AVERAGE REWARD POLICY IMPROVEMENT THEOREM

Let $d_\pi \in \mathbb{R}^{|\mathcal{S}|}$ be the probability column vector whose components are $d_\pi(s)$, $P_\pi \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$ be the transition matrix under policy π whose (s, s') component is $P_\pi(s'|s) = \sum_a P(s'|s,a)\pi(a|s)$, and $P^*_\pi = \lim_{t\to\infty} P^t_\pi$ be the limiting distribution for the transition matrix. We use $\|\cdot\|_p$ to denote the operator norm of a matrix. In particular $\|\cdot\|_1$ and $\|\cdot\|_\infty$ are the maximum absolute column sum and maximum absolute row sum of a matrix respectively (Horn & Johnson, 2012).

Suppose we have a new policy π' obtained via some update rule from the current policy π. Similar to the discounted case, we would like to measure their performance difference $\rho(\pi') - \rho(\pi)$ using an expression which depends on π and some divergence metric between the two policies. The following identity shows that $\rho(\pi') - \rho(\pi)$ can be expressed using the advantange function of π.

Lemma 1. *For any two stochastic policies π and π':*

$$\rho(\pi') - \rho(\pi) = \lim_{N\to\infty} \frac{1}{N} \mathop{\mathbb{E}}_{\tau\sim\pi'} \left[\sum_{t=0}^{N-1} A^{\pi}(s_t, a_t) \right] = \mathop{\mathbb{E}}_{\substack{s\sim d_{\pi'} \\ a\sim\pi'}} [A^{\pi}(s,a)] \tag{6}$$

Proof. See Supplementary Materials. □

Lemma 1 is the an extension of the well-known policy difference lemma from Kakade & Langford (2002) to the average reward case. Note that this expression depends on samples drawn from π'. However we can show through the following lemma that when d_π and $d_{\pi'}$ are "close," we can evaluate the expression in (6) using samples from d_π (see supplementary material for proof).

Lemma 2. *For any two stochastic policies π and π', the following bound holds:*

$$\mathop{\mathbb{E}}_{\substack{s\sim d_{\pi} \\ a\sim\pi'}} [A^{\pi}(s,a)] - 2\epsilon D_{TV}(d_{\pi'} \parallel d_\pi) \leq \rho(\pi') - \rho(\pi) \leq \mathop{\mathbb{E}}_{\substack{s\sim d_{\pi} \\ a\sim\pi'}} [A^{\pi}(s,a)] + 2\epsilon D_{TV}(d_{\pi'} \parallel d_\pi) \tag{7}$$

where $\epsilon = \max_s \left| \mathbb{E}_{a\sim\pi'(a|s)}[A^{\pi}(s,a)] \right|$.

Since our goal is to relate policy performance to the divergence between policies, we use the following result to connect the divergence between stationary distributions $d_{\pi'}$ and d_π and their corresponding policies π' and π (see supplementary materials for proof).

Lemma 3. *The divergence between the stationary distributions d_π and $d_{\pi'}$ can be upper bounded by the average divergence between policies π and π':*

$$D_{TV}(d_{\pi'} \parallel d_\pi) \leq \left\| (I - P_{\pi'} + P^*_{\pi'})^{-1} \right\|_\infty \mathop{\mathbb{E}}_{s\sim d_\pi} [D_{TV}(\pi' \parallel \pi)[s]] \tag{8}$$

For a transition matrix P, the matrix $(I - P + P^*)^{-1}$ is known as the *fundamental matrix of the Markov chain*. This matrix is guaranteed to exist for regular Markov chains (Kemeny & Snell, 1960). By extension, its ∞-norm is guaranteed to be finite.

The following theorem connects the average reward performance of two policies and their average divergence.

Theorem 1. *For any two stochastic policies π and π', the following bounds hold:*

$$\rho(\pi') - \rho(\pi) \leq \mathop{\mathbb{E}}_{\substack{s\sim d_{\pi} \\ a\sim\pi}} \left[\frac{\pi'(a|s)}{\pi(a|s)} A^{\pi}(s,a) \right] + 2\xi \mathop{\mathbb{E}}_{\mathbf{s}\sim d_\pi} [D_{TV}(\pi' \parallel \pi)[s]] \tag{9}$$

$$\rho(\pi') - \rho(\pi) \geq \mathop{\mathbb{E}}_{\substack{s\sim d_{\pi} \\ a\sim\pi}} \left[\frac{\pi'(a|s)}{\pi(a|s)} A^{\pi}(s,a) \right] - 2\xi \mathop{\mathbb{E}}_{\mathbf{s}\sim d_\pi} [D_{TV}(\pi' \parallel \pi)[s]] \tag{10}$$

where $\xi = \max_s \mathbb{E}_{a\sim\pi'} \left| A^{\pi}(s,a) \right| \left\| (I - P_{\pi'} + P^*_{\pi'})^{-1} \right\|_\infty$.

Proof. Combine the bounds from Lemma 2 and Lemma 3. Then rewrite the expectation for $A^{\pi}(s,a)$ as an expectation w.r.t. π using importance sampling gives us the desired bound. □

The right-hand-side of the bounds in Theorem 1 are guaranteed to be finite. The bound in Theorem 1 is given in terms of the TV divergence, however the KL divergence is more commonly used in practice. Vuong et al. (2019) also showed by comparing various divergence measures that the KL has superior empirical performance. The relationship between the total variation divergence and KL divergence is given by Pinsker's inequality (Tsybakov, 2008), which says that for any two distributions p and q: $D_{\text{TV}}(p \parallel q) \leq \sqrt{D_{\text{KL}}(p\|q)/2}$. We can then show that

$$\mathop{\mathbb{E}}_{s\sim d_\pi} [D_{\text{TV}}(\pi' \parallel \pi)[s]] \leq \mathop{\mathbb{E}}_{s\sim d_\pi} [\sqrt{D_{\text{KL}}(\pi'\|\pi)[s]/2}] \leq \sqrt{\mathop{\mathbb{E}}_{s\sim d_\pi} [D_{\text{KL}}(\pi'\|\pi)][s]]/2} \tag{11}$$

where the second inequality comes from Jensen's inequality. The inequality in (11) shows that the bounds in Theorem 1 still hold when $\mathbb{E}_{s\sim d_\pi}[D_{\text{TV}}(\pi' \parallel \pi)[s]]$ is substituted with $\sqrt{\mathbb{E}_{s\sim d_\pi}[D_{\text{KL}}(\pi'\|\pi)][s]/2}$.

4.2 APPROXIMATE POLICY ITERATION

One direct consequence of Theorem 1 is that iteratively maximizing the right-hand-side of (10) generates a monotonically improving sequence of policies w.r.t. the average reward objective. Algorithm 1 gives an approximate policy iteration algorithm that produces such a sequence of policies.

Algorithm 1 Approximate Policy Iteration for Average Reward Objective

Initialize: π_0

1: **for** $k = 0, 1, 2, \ldots$ **do**

2: Policy evaluation step: evaluate $A^{\pi_k}(s, a)$ for all s, a.

3: Policy improvement step:

$$\pi_{k+1} = \operatorname*{argmax}_{\pi} \left(\mathop{\mathbb{E}}_{\substack{s\sim d_{\pi_k} \\ a\sim\pi}} [A^{\pi_k}(s,a)] - \xi \sqrt{2 \mathop{\mathbb{E}}_{\mathbf{s}\sim d_{\pi_k}} [D_{\mathrm{KL}}(\pi \| \pi_k)[s]]} \right) \tag{12}$$

where $\xi = \max_s \mathbb{E}_{a\sim\pi} |A^{\pi_k}(s,a)| \left\| (I - P_\pi + P_\pi^*)^{-1} \right\|_\infty$

Proposition 2. *Given an initial policy π_0, Algorithm 1 is guaranteed to generate a sequence of policies $\pi_1, \pi_2, \ldots$ such that $\rho(\pi_0) \le \rho(\pi_1) \le \rho(\pi_2) \le \cdots$.*

Proof. At iteration k, $\mathbb{E}_{s\sim d_{\pi_k}, a\sim\pi}[A^{\pi_k}(s,a)] = 0$, $\mathbb{E}_{s\sim d_{\pi_k}}[D_{\mathrm{KL}}(\pi\|\pi_k)[s]] = 0$ for $\pi = \pi_k$. By Equation (12) and Theorem 1, $\rho(\pi_{k+1}) - \rho(\pi_k) \ge 0$. □

However, Algorithm 1 is difficult to implement in practice since it requires exact knowledge of the advantage function and transition matrix. Furthermore, calculating the term ξ is impractical when the state space is large. In the next section, we will introduce a sample-based algorithm which approximates the update rule given in Equation (12).

5 PRACTICAL APPLICATIONS

5.1 AVERAGE REWARD TRUST REGION POLICY OPTIMIZATION

For large RL problems, it is common to consider some parameterized policy class $\Pi_\theta \subseteq \Pi$. Our goal is to devise a computationally tractable version of Algorithm 1 for policies in Π_θ, i.e. given a policy π_{θ_k} at iteration k, how do we obtain the best possible $\pi_{\theta_{k+1}}$? We can rewrite the unconstrained optimization problem in (12) as a constrained problem:

$$\operatorname*{maximize}_{\pi_\theta \in \Pi_\theta} \quad \mathop{\mathbb{E}}_{\substack{s\sim d_{\pi_{\theta_k}} \\ a\sim\pi_\theta}} [A^{\pi_{\theta_k}}(s,a)] \quad \text{s.t.} \quad \bar{D}_{\mathrm{KL}}(\pi_\theta \parallel \pi_{\theta_k}) \le \delta \tag{13}$$

where $\bar{D}_{\mathrm{KL}}(\pi_\theta \parallel \pi_{\theta_k}) := \mathbb{E}_{s\sim d_{\pi_{\theta_k}}}[D_{\mathrm{KL}}(\pi_\theta\|\pi_{\theta_k})[s]]$. The constraint set $\{\pi_\theta \in \Pi_\theta : \bar{D}_{\mathrm{KL}}(\pi_\theta \parallel \pi_{\theta_k}) \le \delta\}$ is called the *trust region set*. This problem can be regarded as an average reward variant of TRPO from Schulman et al. (2015). Note there that the advantage function in (13) is the *average reward advantage function*. When we set $\pi_{\theta_{k+1}}$ to be the optimal solution to (13), $\pi_{\theta_{k+1}}$ can be shown to have the following performance guarantee:

Proposition 3. *Let $\pi_{\theta_{k+1}}$ be the optimal solution to* (13) *for some $\pi_{\theta_k} \in \Pi_\theta$. The policy performance difference between $\pi_{\theta_{k+1}}$ and π_{θ_k} can be lower bounded by*

$$\rho(\pi_{\theta_{k+1}}) - \rho(\pi_{\theta_k}) \ge -\xi^{\pi_{\theta_{k+1}}} \sqrt{2\delta} \tag{14}$$

where $\xi^{\pi_{\theta_{k+1}}} = \max_s \mathbb{E}_{a\sim\pi_{\theta_{k+1}}} |A^{\pi_{\theta_k}}(s,a)| \left\| (I - P_{\pi_{\theta_{k+1}}} + P^*_{\pi_{\theta_{k+1}}})^{-1} \right\|_\infty$.

Proof. Since $\bar{D}_{\mathrm{KL}}(\pi_{\theta_k} \parallel \pi_{\theta_k}) = 0$, π_{θ_k} is a feasible solution. The objective value is 0 for $\pi_\theta = \pi_{\theta_k}$. The bound follows from (10) and (11) where the average KL is bounded by δ. □

We note that the discounted TRPO problem from Schulman et al. (2015) is equivalent to (13) as $\gamma \to 1$. This seems to imply that the discounted TRPO problem for large discount factors could also learn to optimize average reward policies. We will further explore this in Section 7 and demonstrate empirically that discounted TRPO gives unsatisfactory results when optimizing for the average reward. However solving (13) for policies with high-dimensional parameter spaces such as deep neural networks can be highly challenging. Several algorithms have been proposed for efficiently solving the discounted version of (13): Schulman et al. (2015) and Wu et al. (2017) converts (13) into a convex problem via Taylor approximations, another approach is to first solve (13) in the nonparametric policy space then project the result back into the parameter space (Abdolmaleki et al., 2018; Vuong et al., 2019). These algorithms can easily be adapted for the average reward case and are theoretically justified via Theorem 1 and Proposition 3. One notable difference compared to the discounted case is the estimation of the critic (see Appendix C), we provide as an example an overview for the average reward version of the TRPO (ATRPO) algorithm in Appendix D.1.

5.2 Reinforcement Learning with Average Cost Constraints

In addition to learning to improve its long-term performance, many real-life applications of RL also require the agent to satisfy certain safety constraints. A mathematically principled framework for incorporating safety constraints into RL is using Constraint Markov Decision Processes (CMDP). A CMDP (Kallenberg, 1983; Ross, 1985; Altman, 1999) is an MDP equipped with a constraint set Π_c, a CMDP problems finds a policy π that maximizes an agent's long-run reward given that $\pi \in \Pi_c$. We consider two forms of constraint sets: the average cost constraint set $\{\pi \in \Pi : \rho_c(\pi) \leq b\}$ and the discounted cost constraint set $\{\pi \in \Pi : \rho_{c,\gamma}(\pi) \leq b\}$. Here b is some given constraint bound, $\rho_c(\pi) = \lim_{N\to\infty} \frac{1}{N} \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{N-1} c(s_t, a_t)\right]$, and $\rho_{c,\gamma}(\pi) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t)\right]$ for some bounded cost function $c : \mathcal{S} \times \mathcal{A} \to [c_{\min}, c_{\max}]$. However, directly adding cost constraints to any iterative policy improvement algorithms can be sample inefficient since the cost constraint needs to be evaluated using samples from the new policy after every policy update. Instead, Achiam et al. (2017) proposed updating π_{θ_k} via the following optimization problem:

$$\underset{\pi_\theta \in \Pi_\theta}{\text{maximize}} \quad \underset{\substack{s\sim d_{\pi_{\theta_k},\gamma} \\ a\sim\pi_\theta}}{\mathbb{E}} [A_\gamma^{\pi_{\theta_k}}(s,a)] \quad \text{s.t.} \quad \tilde{\rho}_{c,\gamma}(\pi_\theta) \leq b,\ \bar{D}_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_k}) \leq \delta. \tag{15}$$

Here, $\tilde{\rho}_{c,\gamma}(\pi_\theta) := \rho_{c,\gamma}(\pi_{\theta_k}) + \frac{1}{1-\gamma} \mathbb{E}_{s\sim d_{\pi_{\theta_k},\gamma}, a\sim\pi_\theta}\left[A_{c,\gamma}^{\pi_{\theta_k}}(s,a)\right]$ and $A_{c,\gamma}^{\pi_{\theta_k}}(s,a)$ is the discounted cost advantage function where we replace the reward with the cost. Note that in (15) the original cost constraint was replaced by a discounted surrogate cost constraint $\tilde{\rho}_{c,\gamma}(\pi_\theta)$ which can be evaluated using samples from π_{θ_k}. The bound (4) still hold when the reward function is replaced with the cost (see also Corollary 2 from Achiam et al. (2017)). Therefore by (4) and (11):

$$|\rho_{c,\gamma}(\pi_\theta) - \tilde{\rho}_{c,\gamma}(\pi_\theta)| \leq \frac{\gamma\epsilon_{c,\gamma}}{(1-\gamma)^2}\sqrt{2\bar{D}_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_k})} \tag{16}$$

where $\epsilon_{c,\gamma} = \max_s \left|\mathbb{E}_{a\sim\pi'}[A_{c,\gamma}^{\pi}(s,a)]\right|$. This shows that the surrogate cost is a good approximation to $\rho_{c,\gamma}(\pi_\theta)$ when π_θ and π_{θ_k} are close. Using (16) and the trust region constraint, Achiam et al. (2017) upper bounded the worst-case constraint violation for when $\pi_{\theta_{k+1}}$ is the solution to (15).

The framework is problematic when the cost constraint is undiscounted. Define the average surrogate cost as $\tilde{\rho}_c(\pi_\theta) := \rho_c(\pi_{\theta_k}) + \mathbb{E}_{s\sim d_{\pi_{\theta_k}}, a\sim\pi_{\theta_k}}[A_c^{\pi_\theta}(s,a)]$. We can easily show that

$$\lim_{\gamma\to 1}(1-\gamma)(\rho_{c,\gamma}(\pi_\theta) - \tilde{\rho}_{c,\gamma}(\pi_\theta)) = \rho_c(\pi_\theta) - \tilde{\rho}_c(\pi_\theta) \quad \text{and} \quad \lim_{\gamma\to 1}\frac{\gamma\epsilon_{c,\gamma}}{1-\gamma}\sqrt{2\bar{D}_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_k})} = \infty$$

However, by Theorem 1 and (11):

$$|\rho_c(\pi_\theta) - \tilde{\rho}_c(\pi_\theta)| \leq \xi_c^{\pi_\theta}\sqrt{2\bar{D}_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_k})} \tag{17}$$

where $\xi_c^{\pi_\theta} = \max_s \mathbb{E}_{a\sim\pi_\theta}\left|A_c^{\pi_{\theta_k}}(s,a)\right| \left\|(I - P_{\pi_\theta} + P_{\pi_\theta}^*)^{-1}\right\|_\infty$. We then have the following result:

Proposition 4. *Suppose π_θ and π_{θ_k} satisfy the constraints $\tilde{\rho}_c(\pi_\theta) < b$ and $\bar{D}_{KL}(\pi_\theta \parallel \pi_{\theta_k}) \leq \delta$, then*

$$\rho_C(\pi_\theta) \leq b + \xi_c^{\pi_\theta}\sqrt{2\delta} \tag{18}$$

The upper-bound in Proposition 4 provides a worst-case constraint violation guarantee when π_θ is the solution to the average-cost variant of (15). It is an undiscounted parallel to Proposition 2 in Achiam et al. (2017) which provides a similar guarantee for the discounted case. It shows that contrary to what was previously believed (Tessler et al., 2019), (15) can easily be modified to accommodate for average cost constraints and still satisfy an upper bound for worst-case constraint violation. Scalable algorithms have been proposed for approximately solving (15) (Achiam et al., 2017; Zhang et al., 2020). Proposition 4 shows that these algorithms can be generalized to average cost constraints with only minor modifications. In Appendix D.2, we will show how the CPO algorithm (Achiam et al., 2017) can be modified for average cost constraints.

6 Related Work

Dynamic programming algorithms for finding the optimal average reward policies have been well-studied (Howard, 1960; Blackwell, 1962; Veinott, 1966). Several Q-learning-like algorithms for problems with unknown dynamics have been proposed, such as R-Learning (Schwartz, 1993), RVI Q-Learning (Abounadi et al., 2001), and CSV-Learning (Yang et al., 2016). Mahadevan (1996) conducted a thorough empirical analysis of the R-Learning algorithm. More recently, POLITEX updates policies using a Boltzmann distribution over the sum of action-value function estimates of the previous policies (Abbasi-Yadkori et al., 2019) and Wei et al. (2020) introduced a model-free algorithm for optimizing the average reward of weakly-communicating MDPs. For policy gradient methods, Baxter & Bartlett (2001) showed that if $1/(1-\gamma)$ is large compared to the mixing time of the Markov Chain induced by the MDP, then the gradient of $\rho_\gamma(\pi)$ can accurately approximate the gradient of $\rho(\pi)$. Kakade (2001) extended upon this result and provided an error bound on using an optimal discounted policy to maximize the average reward.

Policy improvement bounds have been extensively explored in the discounted case. The results from Schulman et al. (2015) is an extension of Kakade & Langford (2002) which restricted the policy class to a mixture of policies. Pirotta et al. (2013) also proposed an alternative generalization to Kakade & Langford (2002). Achiam et al. (2017) improved upon Schulman et al. (2015) by replacing the maximum divergence with the average divergence.

7 Experiments

In our experiments, we aim to answer the following:

1. Are the average reward versions of trust-region-based algorithms effective at controlling agents in long/infinite-horizon tasks?
2. Can discounting be used in place of the average reward objective?
3. Which performance criterion is more suitable for continuing control tasks?

Our design goal for the experiments is to simulate continuing control tasks where the agent can interact with the environment indefinitely. We consider three tasks (Ant, HalfCheetah, and Humanoid) from the MuJoCo physical simulator (Todorov et al., 2012) implemented in the OpenAI gym (Brockman et al., 2016). The natural goal is to train the agents to walk as fast as possible without falling. However the standard MuJoCo tasks are episodic with terminal states. We convert these tasks into continuing control tasks via the following: when the agent falls, the agent incurs a large cost for falling, but then continues the trajectory from a random start state. We use these tasks for both training and evaluation.

In this section we compare the discounted version of the TRPO algorithm with its average reward counterpart (ATRPO). Additional experimental results for the average cost version of CPO (ACPO) are presented in the supplementary material where we show that the average cost surrogate function presented in Section 5.2 can be used to accurately approximate average cost constraints. Detailed specifications and hyperparameter settings for the experiments can be found in the Appendix F.

Figure 1 shows that ATRPO improves performance by 33.3%, 9.6%, 21.2% on HalfCheetah, Ant and Humanoid respectively over the best discount rates 0.99. One point worth noting is that increasing the discount rate does not necessarily lead to better performance. This seems to contradict the equivalence relation stated in Section 5.1. However in practice, the optimization problem in (13) (and

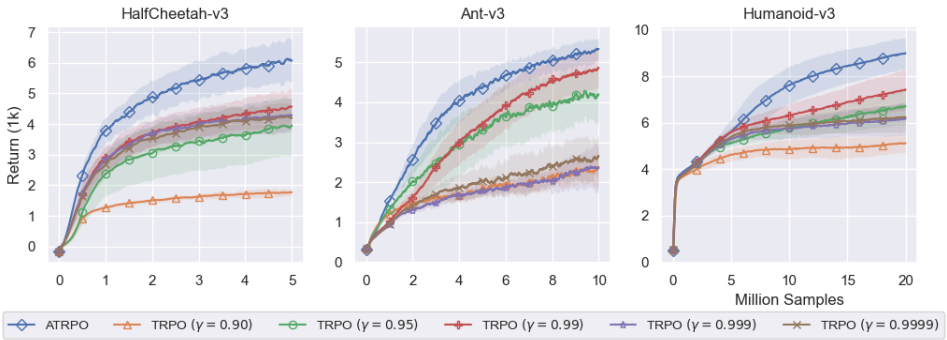

Figure 1: Learning curves comparing ATRPO and TRPO with different discount factors. The solid lines represent the total return of trajectories of length of 1000 averaged over the last 100 trajectories averaged over 10 random seeds, and the shaded region represent one standard deviation.

its discounted variant) are approximated by applying gradient updates via samples collected in the environment. It has been shown that choosing a smaller discount rate can reduce the variance of the gradient estimator (Zhao et al., 2011; Schulman et al., 2016) and improve generalization (Amit et al., 2020). Many algorithms are known to become unstable as $\gamma \to 1$ (Naik et al., 2019). In practice, the discount factor serves as a hyperparameter which can be tuned to improve performance. We have shown here that using the average reward criteria obviates the need to tune the discount factor while delivering superior performance.

We also evaluated the performance of the agents on extremely long trajectories. After every 500k samples of training (with learning curves corresponding to Figure 1), we run 10 separate test trajectories of fixed length 10,000[1] using the current policy with no exploration. The random seed used for evaluation is different from those used in training. The average of the then trajectories are reported in Figure 2. ATRPO also provides better performance on these test trajectories.

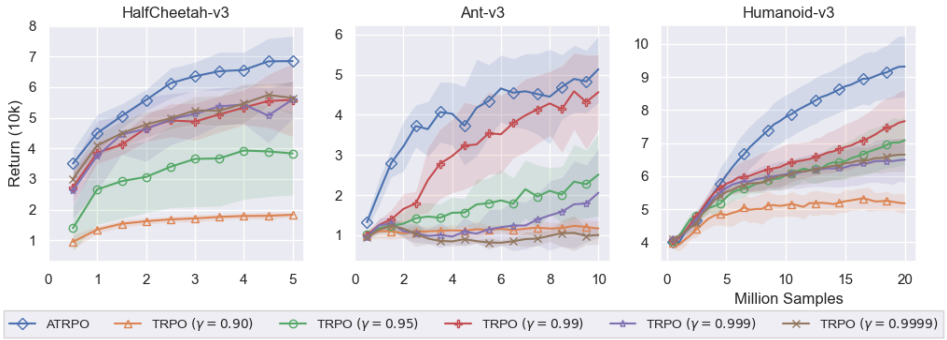

Figure 2: Comparing performance on evaluation trajectories of length 10,000. The solid line represents the total return averaged over 10 trajectories for separate test runs with no exploration and 10 different random seeds from those used during training. The shaded area is one standard deviation.

8 Discussion

We showed that the average reward performance difference can be lower bounded by the average divergence between the two policies. To the best of our knowledge, our work is the first to theoretically justify the use of trust-region algorithms for optimizing the average reward. Our results demonstrate both theoretically and practically that the average reward criterion is more suited than the discounted criterion for continuing control tasks and we hope our work may serve as a stepping stone for further research in the average reward criteria.

[1] We removed the maximum 1000 episode length cap from the original OpenAI gym version of MuJoCo.

REFERENCES

Yasin Abbasi-Yadkori, Peter Bartlett, Kush Bhatia, Nevena Lazic, Csaba Szepesvari, and Gellért Weisz. Politex: Regret bounds for policy iteration using expert prediction. In *International Conference on Machine Learning*, pp. 3692–3702, 2019.

Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation. *International Conference on Learning Representation (ICLR)*, 2018.

Jinane Abounadi, D Bertsekas, and Vivek S Borkar. Learning algorithms for markov decision processes with average cost. *SIAM Journal on Control and Optimization*, 40(3):681–698, 2001.

Joshua Achiam. UC Berkeley CS 285 (Fall 2017), Advanced Policy Gradients, 2017. URL: `http://rail.eecs.berkeley.edu/deeprlcourse-fa17/f17docs/lecture_13_advanced_pg.pdf`. Last visited on 2020/05/24.

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 22–31. JMLR. org, 2017.

Eitan Altman. *Constrained Markov decision processes*, volume 7. CRC Press, 1999.

Ron Amit, Ron Meir, and Kamil Ciosek. Discount factor as a regularizer in reinforcement learning. In *International conference on machine learning*, 2020.

Jonathan Baxter and Peter L Bartlett. Infinite-horizon policy-gradient estimation. *Journal of Artificial Intelligence Research*, 15:319–350, 2001.

Dimitri P Bertsekas, Dimitri P Bertsekas, Dimitri P Bertsekas, and Dimitri P Bertsekas. *Dynamic programming and optimal control*, volume 1,2. Athena scientific Belmont, MA, 1995.

David Blackwell. Discrete dynamic programming. *The Annals of Mathematical Statistics*, pp. 719–726, 1962.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym, 2016.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *International Conference on Machine Learning (ICML)*, 2018.

Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

Ronald A Howard. *Dynamic programming and markov processes.* John Wiley, 1960.

Sham Kakade. Optimizing average reward using discounted rewards. In *International Conference on Computational Learning Theory*, pp. 605–615. Springer, 2001.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *International Conference on Machine Learning*, volume 2, pp. 267–274, 2002.

L.C.M. Kallenberg. *Linear Programming and Finite Markovian Control Problems*. Centrum Voor Wiskunde en Informatica, 1983.

J.G. Kemeny and I.J. Snell. *Finite Markov Chains*. Van Nostrand, New Jersey, 1960.

Erich L Lehmann and George Casella. *Theory of point estimation*. Springer Science & Business Media, 2006.

Sridhar Mahadevan. Average reward reinforcement learning: Foundations, algorithms, and empirical results. *Machine learning*, 22(1-3):159–195, 1996.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *NIPS Deep Learning Workshop*, 2013.

Abhishek Naik, Roshan Shariff, Niko Yasui, and Richard S Sutton. Discounted reinforcement learning is not an optimization problem. *NeurIPS Optimization Foundations for Reinforcement Learning Workshop*, 2019.

Jan Peters and Stefan Schaal. Reinforcement learning of motor skills with policy gradients. *Neural networks*, 21(4):682–697, 2008.

Matteo Pirotta, Marcello Restelli, Alessio Pecorino, and Daniele Calandriello. Safe policy iteration. In *International Conference on Machine Learning*, pp. 307–315, 2013.

Martin L Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.

Keith W Ross. Constrained markov decision processes with queueing applications. *Dissertation Abstracts International Part B: Science and Engineering[DISS. ABST. INT. PT. B- SCI. & ENG.],*, 46(4), 1985.

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

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *International Conference on Learning Representations (ICLR)*, 2016.

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

Anton Schwartz. A reinforcement learning method for maximizing undiscounted rewards. In *Proceedings of the tenth international conference on machine learning*, volume 298, pp. 298–305, 1993.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484, 2016.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419): 1140–1144, 2018.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pp. 1057–1063, 2000.

Chen Tessler, Daniel J Mankowitz, and Shie Mannor. Reward constrained policy optimization. *International Conference on Learning Representation (ICLR)*, 2019.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012.

Alexandre B Tsybakov. *Introduction to nonparametric estimation*. Springer Science & Business Media, 2008.

Arthur F Veinott. On finding optimal policies in discrete dynamic programming with no discounting. *The Annals of Mathematical Statistics*, 37(5):1284–1294, 1966.

Quan Vuong, Yiming Zhang, and Keith W Ross. Supervised policy update for deep reinforcement learning. In *International Conference on Learning Representation (ICLR)*, 2019.

Chen-Yu Wei, Mehdi Jafarnia-Jahromi, Haipeng Luo, Hiteshi Sharma, and Rahul Jain. Model-free reinforcement learning in infinite-horizon average-reward markov decision processes. In *International conference on machine learning*, 2020.

Yuhuai Wu, Elman Mansimov, Roger B Grosse, Shun Liao, and Jimmy Ba. Scalable trust-region method for deep reinforcement learning using kronecker-factored approximation. In *Advances in neural information processing systems (NIPS)*, pp. 5285–5294, 2017.

Shangdong Yang, Yang Gao, Bo An, Hao Wang, and Xingguo Chen. Efficient average reward reinforcement learning using constant shifting values. In *AAAI*, pp. 2258–2264, 2016.

Yiming Zhang, Quan Vuong, and Keith W Ross. First order optimization in policy space for constrained deep reinforcement learning. *arXiv preprint arXiv:2002.06506*, 2020.

Tingting Zhao, Hirotaka Hachiya, Gang Niu, and Masashi Sugiyama. Analysis and improvement of policy gradient estimation. In *Advances in Neural Information Processing Systems*, pp. 262–270, 2011.