Optimistic Policy Optimization via Multiple Importance Sampling

Anonymous Authors[1]

Abstract

Policy Search (PS) is an effective approach to Reinforcement Learning (RL) for solving control tasks with continuous state-action spaces. In this paper, we address the exploration-exploitation trade-off in PS by proposing an approach based on Optimism in Face of Uncertainty. We cast the PS problem as a suitable Multi Armed Bandit (MAB) problem, defined over the policy parameter space, and we propose a class of algorithms that effectively exploit the problem structure, by leveraging Multiple Importance Sampling to perform an off-policy estimation of expected return. We show that the regret of the proposed approach is bounded by $\widetilde{\mathcal{O}}(\sqrt{T})$ for both discrete and continuous parameter spaces. Finally, we evaluate our algorithms on tasks of varying difficulty, comparing them with existing MAB and RL algorithms.

1. Introduction

Reinforcement Learning (RL, Sutton & Barto, 2018) allows an agent to learn a control task by repeated interaction with the environment in the presence of a reward signal. One of the current challenges of RL is to master tasks, such as robotic locomotion, in which states and actions are naturally modeled as real numbers. Policy optimization (PO, Deisenroth et al., 2013) is a family of RL algorithms that are particularly suited to this class of problems. In PO, the behavior of the agent, or *policy*, is explicitly modeled, typically as a parametric mapping from states to actions. Learning corresponds to the optimization of a performance measure w.r.t. the agent's parameters.

The literature on PO focused mainly on the problem of *finding* the optimal policy with the minimum amount of interaction (Sutton et al., 2000; Sehnke et al., 2008; Silver et al., 2014; Schulman et al., 2015; Mnih et al., 2016; Espeholt et al., 2018). This is well motivated, as interacting with some environments can be very expensive. However, in many cases, we are also interested in the performance of the agent *during* the learning process. We call this *online policy optimization*. This goal is particularly relevant in applications where an agent must be deployed in the real world to perfect its behavior (e.g., robot learning) or to learn at all (e.g., recommender systems). In such cases, the *exploration-exploitation* dilemma arises naturally, as the agent must continually find the right trade-off between complying with its current expertise or widening it by trial and error. Equivalently, it must minimize its total *regret* w.r.t. the optimal behavior. This problem has been thoroughly studied in the field of Multi Armed Bandits (MAB, Auer et al., 2002; Lattimore & Szepesvári, 2019). In this simple framework, an agent has to repeatedly select an action, called an *arm* in this context, in order to maximize an unknown, stochastic reward. This can be seen as RL without states. However, we can also see PO as a MAB-like problem where the set of available actions is the parameter space of the agent. Hopefully, this allows to apply some of the theoretical and algorithmic ideas developed in the MAB literature to the problem of exploration in continuous-action RL, whose solutions proposed so far have been largely heuristic (Houthooft et al., 2016; Haarnoja et al., 2017; 2018). In particular, the Optimism in the Face of Uncertainty (OFU) principle at the heart of the Upper Confidence Bound (UCB, Lai & Robbins, 1985; Agrawal, 1995; Auer, 2002) family of MAB algorithms lends itself to relatively straightforward application to PO. The core idea is simply to overestimate the expected reward of arms, which, in our scenario, are the policies the agent can play. The overestimation is larger for those arms the agent knows less about.

To apply the OFU principle to policy optimization, we need to exploit some structure in the way arms (policy parameters) concur to generate rewards. This is both necessary, as the parameter space is typically continuous, and desirable, as there exists an evident correlation between arms that we can hope to exploit (different policies can lead to similar performances). Both features are absent in the classic MAB formulation, but have been studied before (e.g., Pandey et al., 2007; Kleinberg, 2005, see Section 7 for a brief overview of the related literature, including applications to RL).

In this work, we use Multiple Importance Sampling (MIS, Veach & Guibas, 1995) to capture the information shared by

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

different policies and we employ robust estimators inspired by Bubeck et al. (2013) to overcome the heavy-tail behavior typical of importance sampling. We adapt techniques from Metelli et al. (2018) to build confidence intervals of the expected performance of policy parameters via robust MIS. We employ these tools to design UCB-like algorithms for PO. The proposed algorithms apply to both the policy optimization paradigms:[1] action-based PO, in which we learn the policy parameters (Sutton et al., 2000), and parameter-based PO where optimization is over parametric policy *distributions* (Sehnke et al., 2008). Furthermore, we show how these algorithms can be used both in finite and continuous parameter spaces and we prove that all these algorithms attain a regret bound of $\widetilde{\mathcal{O}}(\sqrt{T})$. Since the optimization problem can be challenging in the continuous case, we propose a general discretization method that allows to trade computational complexity with regret, preserving the sub-linearity.

We start by providing the essential background in Section 2. In Section 3, we develop robust MIS estimators that will play an essential role in the algorithms. In Section 4 we provide a formalization of the online policy optimization problem. The algorithms are presented in Section 5 and analyzed in Section 6. Section 7 relates our work to the existing literature. Finally, in Section 8, we empirically evaluate the proposed methods on continuous control tasks.

2. Preliminaries

In this section, we provide an essential background on policy optimization and multiple importance sampling.

2.1. Policy optimization

In Policy Optimization (Deisenroth et al., 2013) we look for the policy that maximizes the agent's performance on a given RL task. The task is modeled as a discrete-time continuous Markov Decision Process (MDP, Puterman, 2014) $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma, \mu \rangle$, where $\mathcal{S} \in \mathbb{R}^{d_{\mathcal{S}}}$ is the state space; $\mathcal{A} \in \mathbb{R}^{d_{\mathcal{A}}}$ is the action space; $\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is a Markovian transition kernel, such that, for each time h, the next state is drawn as $s_{h+1} \sim \mathcal{P}(\cdot|s_h, a_h)$ that depends only on the current state and action; $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to [-R_{\max}, R_{\max}]$ is a bounded reward signal, such that the next reward $r_{h+1} = \mathcal{R}(s_h, a_h)$ is a function of the current state and action, and $R_{\max} > 0$ is the maximum reward; $\gamma \in (0, 1]$ is a discount factor; and $\mu \in \Delta(\mathcal{S})$ is the initial state distribution, such that the initial state is drawn as $s_0 \sim \mu$. The agent's behavior is modeled as a parametric policy $\pi_{\boldsymbol{\theta}} : \mathcal{S} \to \Delta(\mathcal{A})$, such that the current action is drawn as $a_h \sim \pi_{\boldsymbol{\theta}}(\cdot|s_h)$ depending on the current state, where $\boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^m$ are the policy parameters. Deterministic policies represent a special case where $\pi_{\boldsymbol{\theta}}$ is a Dirac delta function. With abuse of notation, we write $a_h = \pi_{\boldsymbol{\theta}}(s_h)$ in this case. In practice, we consider finite trajectories of length H, the effective horizon of the task. A trajectory is a sequence of states and actions $\tau = [s_0, a_0, s_1, a_1, \ldots, s_{H-1}, a_{H-1}]$. Every policy $\pi_{\boldsymbol{\theta}}$ induces a distribution over trajectories, whose density is denoted as $p_{\boldsymbol{\theta}}$. Our basic measure of performance is the sum of discounted rewards over the trajectory:

$$\mathcal{R}(\tau) = \sum_{h=0}^{H-1} \gamma^h r_{h+1}. \quad (1)$$

Let $J(\boldsymbol{\theta}) = \mathbb{E}_{\tau \sim p_{\boldsymbol{\theta}}}[\mathcal{R}(\tau)]$ be the expected performance under policy $\pi_{\boldsymbol{\theta}}$. In an *online learning* scenario, we aim to maximize the sum of expected performances over a sequence of episodes $t = 0, \ldots, T$. In the *action-based* policy optimization paradigm (Peters & Schaal, 2008), the problem we want to solve is simply:

$$\max_{\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_T \in \Theta} \sum_{t=0}^{T} \mathop{\mathbb{E}}_{\tau_t \sim p_{\boldsymbol{\theta}_t}} [\mathcal{R}(\tau_t)] = \max_{\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_T \in \Theta} \sum_{t=0}^{T} J(\boldsymbol{\theta}_t), \quad (2)$$

where $\pi_{\boldsymbol{\theta}_t}$ is the policy used for episode t. In the action-based paradigm, stochastic policies are typically employed in order to ensure exploration, although deterministic policies have also been used with the addition of exogenous noise (Silver et al., 2014). Instead, in the *parameter-based* policy optimization paradigm (Sehnke et al., 2008), we define a distribution over policy parameters, $\nu_{\boldsymbol{\xi}} \in \Delta(\Theta)$, called *hyperpolicy*, where $\boldsymbol{\xi} \in \Xi \subseteq \mathbb{R}^d$ are the hyperpolicy parameters, or *hyperparameters*. For each episode t, policy parameters are drawn as $\boldsymbol{\theta}_t \sim \nu_{\boldsymbol{\xi}_t}$ and the whole trajectory is executed with $\pi_{\boldsymbol{\theta}_t}$. In this case, the optimization problem becomes:

$$\max_{\boldsymbol{\xi}_0, \ldots, \boldsymbol{\xi}_T \in \Xi} \sum_{t=0}^{T} \mathop{\mathbb{E}}_{\boldsymbol{\theta}_t \sim \nu_{\boldsymbol{\xi}_t}} [J(\boldsymbol{\theta}_t)] = \max_{\boldsymbol{\xi}_0, \ldots, \boldsymbol{\xi}_T \in \Xi} \sum_{t=0}^{T} J(\boldsymbol{\xi}_t). \quad (3)$$

In the parameter-based paradigm, deterministic policies are typically employed, paired with stochastic hyperpolicies in order to ensure exploration.

2.2. Multiple importance sampling

Importance sampling (Cochran, 2007; Owen, 2013) is a technique that allows estimating the expectation of a function under some *target* or *proposal* distribution with samples drawn from a different distribution, called *behavioral*.

Let P and Q be probability measures on a measurable space $(\mathcal{Z}, \mathcal{F})$, such that $P \ll Q$ (i.e., P is absolutely continuous w.r.t. Q). The importance weight $w_{P/Q}$ is the Radon-Nikodym derivative of P w.r.t. Q, i.e., $w_{P/Q} \equiv \frac{\mathrm{d}P}{\mathrm{d}Q}$. Let p and q be the densities of P and Q, respectively, w.r.t. a reference measure. From the chain rule, $w_{P/Q} = \frac{p}{q}$. In the continuous case, p and q are probability density functions (pdf's) of absolutely continuous random variables having laws P and Q, respectively, and $w_{P/Q}$ is a likelihood ra-

[1] We follow the taxonomy of Metelli et al. (2018).

tio. Given a bounded function $f : \mathcal{Z} \rightarrow \mathbb{R}$, and a set of i.i.d. outcomes $z_1, \dots, z_N$ sampled from Q, the importance sampling estimator of $\mu := \mathbb{E}_{z \sim P}\left[f(z)\right]$ is:

$$\widehat{\mu}_{\text{IS}} = \frac{1}{N} \sum_{i=1}^{N} f(z_i) w_{P/Q}(z_i), \tag{4}$$

which is an unbiased estimator (Owen, 2013), i.e., $\mathbb{E}_{z_i \overset{\text{iid}}{\sim} Q}\left[\widehat{\mu}_{IS}\right] = \mu$.

Multiple importance sampling (Veach & Guibas, 1995) is a generalization of the importance sampling technique which allows samples drawn from several different behavioral distributions to be used for the same estimate. Let $Q_1, \dots, Q_K$ be all probability measures over the same probability space as P, and $P \ll Q_k$ for $k = 1, \dots, K$. Let $\beta_1(z), \dots, \beta_K(z)$ be mixture weights, i.e., for all $z \in \mathcal{Z}$, $\beta_1(z) + \dots + \beta_K(z) = 1$ and $\beta_k(z) \geqslant 0$ for $k = 1, \dots, K$. Let z_{ik} denote the i-th sample drawn from Q_k. Given N_k i.i.d. samples from each Q_k, the Multiple Importance Sampling estimator (MIS) is:

$$\widehat{\mu}_{\text{MIS}} = \sum_{k=1}^{K} \frac{1}{N_k} \sum_{i=1}^{N_k} \beta_k(z_{ik}) w_{P/Q_k}(z_{ik}) f(z_{ik}), \tag{5}$$

which is also an unbiased estimator of μ for any valid choice of the mixture weights. A common choice of the mixture weights having desirable variance properties is the balance heuristic (Veach & Guibas, 1995):

$$\beta_k(z) = \frac{N_k q_k(z)}{\sum_{j=1}^{K} N_j q_j(z)}, \tag{6}$$

which yields the Balance Heuristic estimator (BH):

$$\widehat{\mu}_{\text{BH}} = \sum_{k=1}^{K} \sum_{i=1}^{N_k} \frac{p(z_{ik})}{\sum_{j=1}^{K} N_j q_j(z_{ik})} f(z_{ik}). \tag{7}$$

Since (6) are valid mixture weights, $\widehat{\mu}_{\text{BH}}$ is an unbiased estimator of μ. Moreover, its variance is not significantly larger than any other choice of the mixture weights (Veach & Guibas, 1995, Theorem 1). An appealing interpretation of the balance heuristic is that we can look at $\widehat{\mu}_{\text{BH}}$ as an importance sampling estimation in which samples are drawn from the mixture $\Phi = \frac{p(z_{ik})}{\sum_{j=1}^{K} \frac{N_j}{N} q_j(z_{ik})}$.

To further characterize the variance of this estimator, we introduce the concept of Rényi divergence. Given probability measures P and Q on $(\mathcal{Z}, \mathcal{F})$, where $P \ll Q$ and Q is σ-finite, the α-Rényi divergence is defined as (Rényi, 1961):

$$D_\alpha(P \| Q) = \frac{1}{\alpha - 1} \log \int_{\mathcal{Z}} \left(w_{P/Q}\right)^\alpha \mathrm{d}Q, \tag{8}$$

for $\alpha \in [0, \infty]$[2]. We denote with $d_\alpha(P \| Q) = \exp\{D_\alpha(P \| Q)\}$ the exponentiated α-Rényi divergence. Of particular interest is d_2, as the variance of the importance weight is $\mathbb{V}\mathrm{ar}_{z \sim Q}\left[w_{P/Q}(z)\right] = d_2(P \| Q) - 1$, which is a divergence itself (Cortes et al., 2010). For this reason, we always mean the 2-Rényi divergence when omitting the order α. The Rényi divergence was used by Metelli et al. (2018, Lemma 4.1) to upper bound the variance of the importance sampling estimator as $\mathbb{V}\mathrm{ar}_{z_i \overset{\text{iid}}{\sim} q}\left[\widehat{\mu}_{\text{IS}}\right] \leqslant \|f\|_\infty^2 \, d_2(P \| Q)/N$. A similar result can be derived for the BH estimator:

Lemma 1. *Let P and $\{Q_k\}_{k=1}^{K}$ be probability measures on the measurable space $(\mathcal{Z}, \mathcal{F})$ such that $P \ll Q_k$ and $d_2(P \| Q_k) < \infty$ for $k = 1, \dots, K$. Let $f : \mathcal{Z} \rightarrow \mathbb{R}$ be a bounded function, i.e., $\|f\|_\infty < \infty$. Let $\widehat{\mu}_{BH}$ be the balance heuristic estimator of f, as defined in (7), using N_k i.i.d. samples from each Q_k. Then, the variance of $\widehat{\mu}_{BH}$ can be upper bounded as:*

$$\mathop{\mathbb{V}\mathrm{ar}}_{z_{ik} \overset{iid}{\sim} Q_k} \left[\widehat{\mu}_{BH}\right] \leqslant \|f\|_\infty^2 \frac{d_2(P \| \Phi)}{N},$$

where $N = \sum_{k=1}^{K} N_k$ is the total number of samples and $\Phi = \sum_{k=1}^{K} \frac{N_k}{N} Q_k$ is a finite mixture.

3. Robust Importance Sampling Estimation

In this section, we discuss how to perform a robust importance weighting estimation. Recently it has been observed that, in many cases of interest, the plain estimator (4) presents problematic tail behaviors (Metelli et al., 2018), preventing the use of exponential concentration inequalities.[3] A common heuristic to address this problem consists in truncating the weight (Ionides, 2008):

$$\check{\mu}_{\text{IS}} = \frac{1}{N} \sum_{i=1}^{N} \min\left\{M, w_{P/Q}(z_i)\right\} f(z_i), \tag{9}$$

where M is a threshold to limit the magnitude of the importance weight. Similarly, for the multiple importance sampling case, restricting to the BH, we have:

$$\check{\mu}_{\text{BH}} = \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N_k} \min\left\{M, \frac{p(z_{ik})}{\sum_{j=1}^{K} \frac{N_j}{N} q_j(z_{ik})}\right\} f(z_{ik}). \tag{10}$$

Clearly, since we are changing the importance weights, we introduce a bias term, but, by reducing the range of the estimation, we get a benefit in terms of variance. Below, we present the bias-variance analysis of the estimator $\check{\mu}_{\text{BH}}$ and we conclude by showing that we are able, using an adaptive truncation, to guarantee an exponential concentration (differently from the non-truncated case).

Lemma 2. *Let P and $\{Q_k\}_{k=1}^{N}$ be probability measures on the measurable space $(\mathcal{Z}, \mathcal{F})$ such that $P \ll Q_k$ and there exists $\epsilon \in (0, 1]$ s.t. $d_{1+\epsilon}(P \| Q_k) < \infty$ for $k = 1, \dots, K$. Let $f : \mathcal{Z} \rightarrow \mathbb{R}_+$ be a bounded non-negative function, i.e., $\|f\|_\infty < \infty$. Let $\check{\mu}_{BH}$ be the truncated balance heuristic estimator of f, as defined in (10), using N_k i.i.d. samples*

[2] The special cases $\alpha = 0, 1$ and ∞ are defined as limits.

[3] Unless we require that $d_\infty(P \| \Phi)$ is finite, i.e., that the importance weight have finite supremum, there always exists a value $\alpha > 1$ such that $d_\alpha(P \| \Phi) = +\infty$.

from each Q_k. Then, the bias of $\check{\mu}_{BH}$ can be bounded as:

$$0 \leqslant \mu - \mathop{\mathbb{E}}_{z_{ik} \overset{iid}{\sim} Q_k} [\check{\mu}_{BH}] \leqslant \|f\|_\infty M^{-\epsilon} d_{1+\epsilon}\left(P\|\Phi\right)^\epsilon, \quad (11)$$

and the variance of $\check{\mu}_{BH}$ can be bounded as:

$$\mathop{\mathbb{V}\mathrm{ar}}_{z_{ik} \overset{iid}{\sim} Q_k} [\check{\mu}_{BH}] \leqslant \|f\|_\infty^2 M^{1-\epsilon} \frac{d_{1+\epsilon}\left(P\|\Phi\right)^\epsilon}{N}, \quad (12)$$

where $N = \sum_{k=1}^K N_k$ is the total number of samples and $\Phi = \sum_{k=1}^K \frac{N_k}{N} Q_k$ is a finite mixture.

It is worth noting that, by selecting $\epsilon = 1$, equation (12) reduces to Lemma 1, as the truncation operation can only reduce the variance. Clearly, the smaller we choose M, the larger the bias. Overall, we are interested in mining the joint contribution of bias and variance. Keeping P and Φ fixed we observe that the bias depends only on M, whereas the variance depends on M and on the number of samples N. Intuitively, we can allow larger truncation thresholds M as the number of samples N increases. The following result states that, when using an *adaptive threshold* depending on N, we are able to reach exponential concentration.

Theorem 1. *Let P and $\{Q_k\}_{k=1}^N$ be probability measures on the measurable space $(\mathcal{Z}, \mathcal{F})$ such that $P \ll Q_k$ and there exists $\epsilon \in (0,1]$ s.t. $d_{1+\epsilon}(P\|Q_k) < \infty$ for $k = 1, \dots, K$. Let $f : \mathcal{Z} \to \mathbb{R}_+$ be a bounded non-negative function, i.e., $\|f\|_\infty < \infty$. Let $\check{\mu}_{BH}$ be the truncated balance heuristic estimator of f, as defined in (10), using N_k i.i.d. samples from each Q_k. Let $M_N = \left(\frac{N d_{1+\epsilon}(P\|\Phi)^\epsilon}{\log \frac{1}{\delta}}\right)^{\frac{1}{1+\epsilon}}$, then with probability at least $1-\delta$:*

$$\check{\mu}_{BH} \leqslant \mu + \|f\|_\infty \left(\sqrt{2} + \frac{1}{3}\right) \left(\frac{d_{1+\epsilon}\left(P\|\Phi\right) \log \frac{1}{\delta}}{N}\right)^{\frac{\epsilon}{1+\epsilon}}, \quad (13)$$

and also, with probability at least $1-\delta$:

$$\check{\mu}_{BH} \geqslant \mu - \|f\|_\infty \left(\sqrt{2} + \frac{4}{3}\right) \left(\frac{d_{1+\epsilon}\left(P\|\Phi\right) \log \frac{1}{\delta}}{N}\right)^{\frac{\epsilon}{1+\epsilon}}. \quad (14)$$

Our adaptive truncation approach and the consequent concentration results resemble the ones proposed in Bubeck et al. (2013). However, unlike Bubeck et al. (2013) we do not remove samples with too high value, but we exploit the nature of the importance weighted estimator only to limit the weight magnitude. Indeed, this form of truncation turned out to be very effective in practice (Ionides, 2008).

4. Problem Formalization

The online learning problem that we aim to solve does *not* fall within the traditional MAB framework (not in its basic version, anyway) and can benefit from an ad-hoc formalization, provided in this section.

Let $\mathcal{X} \subseteq \mathbb{R}^d$ be our decision set, or *arm set* in MAB jargon. Let $(\Omega, \mathcal{F}, P)$ be a probability space. Let $\{Z_{\boldsymbol{x}} : \Omega \to \mathcal{Z} \mid \boldsymbol{x} \in \mathcal{X}\}$ be a set of continuous random vectors parametrized by $\mathcal{X}$, with common sample space $\mathcal{Z} \subseteq \mathbb{R}^m$. We denote with $p_{\boldsymbol{x}}$ the probability density function of $Z_{\boldsymbol{x}}$. Finally, let $f : \mathcal{Z} \to \mathbb{R}$ be a bounded *payoff function*, and $\mu(\boldsymbol{x}) = \mathbb{E}_{z \sim p_{\boldsymbol{x}}}[f(z)]$ its expectation under $p_{\boldsymbol{x}}$. For each iteration $t = 0, \dots, T$, we select an arm $\boldsymbol{x}_t$, draw a sample z_t from $p_{\boldsymbol{x}_t}$, and observe payoff $f(z_t)$, up to horizon T. The goal is to maximize the expected total payoff:

$$\max_{\boldsymbol{x}_0, \dots, \boldsymbol{x}_T \in \mathcal{X}} \sum_{t=0}^T \mathop{\mathbb{E}}_{z_t \sim p_{\boldsymbol{x}_t}} [f(z_t)] = \max_{\boldsymbol{x}_0, \dots, \boldsymbol{x}_T \in \mathcal{X}} \sum_{t=0}^T \mu(\boldsymbol{x}_t). \quad (15)$$

Although we can evaluate $p_{\boldsymbol{x}}$ for each $\boldsymbol{x} \in \mathcal{X}$, we can only observe $f(z_t)$ for the z_t that are actually sampled. This models precisely the online, episodic policy optimization problem. In action-based policy optimization, $\mathcal{X}$ corresponds to the parameter space Θ of a class of stochastic policies $\{\pi_{\boldsymbol{\theta}} \mid \boldsymbol{\theta} \in \Theta\}$, $\mathcal{Z}$ to the set of possible trajectories, $p_{\boldsymbol{x}}$ to the density $p_{\boldsymbol{\theta}}$ over trajectories induced by policy $\pi_{\boldsymbol{\theta}}$, and $f(z)$ to cumulated reward $\mathcal{R}(\tau)$. In parameter-based policy optimization, $\mathcal{X}$ corresponds to the hyperparameter space Ξ of a class of stochastic hyperpolicies $\{\nu_{\boldsymbol{\xi}} \mid \boldsymbol{\xi} \in \Xi\}$, $\mathcal{Z}$ to the policy parameter space Θ, $p_{\boldsymbol{x}}$ to hyperpolicy $\nu_{\boldsymbol{\xi}}$, and $f(z)$ to performance $J(\boldsymbol{\theta})$. In both cases, each iteration corresponds to a single episode, and horizon T is the total number of episodes (not to be confused with the trajectory horizon H). From now on, we will refer to (15) simply as the policy optimization problem.[4]

The peculiarity of this framework, compared to the classic MAB one, is the special structure existing over the arms. In particular, the expected payoff μ of different arms is correlated thanks to the stochasticity of the $p_{\boldsymbol{x}}$'s on a common sample space $\mathcal{Z}$. We *could*, of course, frame policy optimization as a MAB problem, at the cost of ignoring some structure. It would be enough to regard $\mu(\boldsymbol{x})$ as the expectation of a totally unknown, stochastic reward function. This would put us in the continuous MAB framework (Kleinberg et al., 2013), but would ignore the special arm correlation. In the following, we will show how this correlation can be exploited to guarantee efficient exploration.

5. Algorithms

In this section, we use the mathematical tools presented so far to design a policy search algorithm that efficiently explores the space of solutions. The proposed algorithm, called OPTIMIST (Optimistic Policy opTImization via Multiple Importance Sampling with Truncation), is based on the

[4] In abstract terms, (15) is a sequential decision problem over a functional space of random variables, and may have applications beyond policy optimization.

Optimism in the Face of Uncertainty (OFU) principle and follows the Upper Confidence Bound (UCB) strategy (Lai & Robbins, 1985; Agrawal, 1995; Auer et al., 2002) commonly used in Multi Armed Bandit (MAB) problems (Robbins, 1985; Bubeck et al., 2012; Lattimore & Szepesvári, 2019).

To apply the UCB strategy to the policy optimization problem, we need an estimate of the objective $\mu(\boldsymbol{x})$ and a confidence region. We use importance sampling to capture the correlation among the arms. In particular, to better use all the data that we collect, we would like to use a multiple importance sampling estimator like the one from (5). Unfortunately, the heavy-tailed behavior of this estimator would result in an inefficient exploration. Instead, we use the robust balance heuristic estimator $\check{\mu}_{\mathrm{BH}}$ from (10), which has better tail behavior. To simplify the notation, , we treat each sample $\boldsymbol{x}$ as a distinct one. This is w.l.o.g. (as each sample is always multiplied by its number of occurrences anyway) and corresponds to the case $K = t-1$ and $N_k \equiv 1$. Hence, at each iteration t:

$$\check{\mu}_t(\boldsymbol{x}) = \sum_{k=0}^{t-1} \min\left\{ M_{t-1}, \frac{p_{\boldsymbol{x}}(z_k)}{\sum_{j=1}^{t-1} p_{\boldsymbol{x}_j}(z_k)} \right\} f(z_k), \qquad (16)$$

where $M_t = \left(\frac{t d_{1+\epsilon}(p_{\boldsymbol{x}} \| \Phi_t)^{\epsilon}}{\log \frac{1}{\delta_t}} \right)^{\frac{1}{1+\epsilon}}$ and $\Phi_t = \frac{1}{t}\sum_{k=0}^{t-1} p_{\boldsymbol{x}_k}$. According to Theorem 1, the following *index*:

$$B_t^{\epsilon}(\boldsymbol{x}, \delta_t) := \check{\mu}_t(\boldsymbol{x}) + \|f\|_{\infty} \left(\sqrt{2} + \frac{4}{3} \right) \left(\frac{d_{1+\epsilon}(p_{\boldsymbol{x}_t} \| \Phi_t) \log \frac{1}{\delta_t}}{t} \right)^{\frac{\epsilon}{1+\epsilon}}, \qquad (17)$$

is an upper bound on $\mu(\boldsymbol{x})$ with probability at least $1-\delta_t$, i.e., an upper confidence bound. The OPTIMIST algorithm simply selects, at each iteration t, the arm with the largest value of the index $B_t^{\epsilon}(\boldsymbol{x})$, breaking ties deterministically. The pseudocode is provided in Algorithm 1. The initial arm $\boldsymbol{x}_0$ is arbitrary, as no prior information is available. The regret analysis of Section 6 will provide a confidence schedule $(\delta_t)_{t=1}^{T}$. Knowledge of the actual horizon T is not needed. Although we can use any $\epsilon \in (0,1]$, we suggest to use $\epsilon = 1$ in practice, as it yields the more common 2-Rényi divergence. To be able to compute the indexes (or to perform any kind of index maximization), the algorithm needs to store all the $\boldsymbol{x}_t$ together with the observed payoffs $f(z_t)$, hence $\mathcal{O}(Td)$ space is required, where d is the dimensionality of the arm space $\mathcal{X}$ (not to be confused with cardinality $|\mathcal{X}|$, which may be infinite).

The optimization step (line 4) may be very difficult when $\mathcal{X}$ is not discrete (cf. Srinivas et al., 2010), as the index $B_t^{\epsilon}(\boldsymbol{x}, \delta_t)$ is non-convex and non-differentiable. Global optimization methods could be applied at the cost of giving up theoretical guarantees. In practice, this direction may be beneficial, but we leave it to future, more application-oriented work. Instead, we propose a general discretization method. The key intuition, common in the continuous MAB literature, is to make the discretization progressively finer. The pseudocode for this variant, called OPTIMIST 2, is reported in Algorithm 2. Note that the arm space $\mathcal{X}$ itself is fixed (and infinite), as adaptive discretization is performed for optimization purposes only. Implementing any variant of OPTIMIST to solve a policy optimization problem, whether in the action-based or in the parameter-based formulation, requires some additional caveats, discussed in Appendix A.

Algorithm 1 OPTIMIST

```
1: Input: initial arm $\boldsymbol{x}_0$, confidence schedule $(\delta_t)_{t=1}^{T}$, order $\epsilon \in (0,1]$
2: Draw sample $z_0 \sim p_{\boldsymbol{x}_0}$ and observe payoff $f(z_0)$
3: for $t = 1, \dots, T$ do
4:     Select arm $\boldsymbol{x}_t = \arg\max_{\boldsymbol{x} \in \mathcal{X}} B_t^{\epsilon}(\boldsymbol{x}, \delta_t)$
5:     Draw sample $z_t \sim p_{\boldsymbol{x}_t}$ and observe payoff $f(z_t)$
6: end for
```

6. Regret Analysis

In this section, we provide high-probability guarantees on the quality of the solution provided by Algorithm 1. First, we rephrase the optimization problem (1) in terms of *regret minimization*. The instantaneous regret is defined as:

$$\Delta_t = \mu(\boldsymbol{x}^*) - \mu(\boldsymbol{x}_t), \qquad (18)$$

where $\boldsymbol{x}^* = \arg\max_{\boldsymbol{x} \in \mathcal{X}} \mu(\boldsymbol{x})$. Let $\mathit{Regret}(T) = \sum_{t=0}^{T} \Delta_t$ be the total regret. As $\mu(\boldsymbol{x}^*)$ is a constant, problem (15) is trivially equivalent to:

$$\min_{\boldsymbol{x}_0, \dots, \boldsymbol{x}_T \in \mathcal{X}} \mathit{Regret}(T). \qquad (19)$$

In the following, we will show that Algorithm 1 yields sublinear regret under some mild assumptions. The proofs combine techniques from Srinivas et al. (2010) and Bubeck et al. (2013) and are reported in Appendix B. First, we need the following assumption on the Rényi divergence:

Assumption 1. *For all $t = 1, \dots, T$, the $(1+\epsilon)$-Rényi divergence is uniformly bounded as:*

$$\sup_{\boldsymbol{x}_0, \boldsymbol{x}_1, \dots, \boldsymbol{x}_t \in \mathcal{X}} d_{1+\epsilon}(p_{\boldsymbol{x}_t} \| \Phi_t) = v_{\epsilon} < \infty,$$

where $\Phi_t = \frac{1}{t}\sum_{k=0}^{t-1} p_{\boldsymbol{x}_k}$,

which can be easily enforced through careful policy (or hyperpolicy) design (See Appendix TODO).

6.1. Discrete arm set

We start from the discrete case, where $|\mathcal{X}| = K \in \mathbb{N}_+$. This setting is particularly convenient, as the optimization step can be trivially solved in time $\mathcal{O}(Kt)$ per iteration,[5] where t is from evaluation of (16) via clever caching. This sums up to total time $\mathcal{O}(KT^2)$. It is also of practical interest: even in applications where $\mathcal{X}$ is naturally continuous (e.g., robotics), the set of solutions that can be actually tried in practice may sometimes be constrained to a discrete, reasonably small set. In this simple setting, OPTIMIST achieves $\widetilde{\mathcal{O}}(T^{\frac{1}{1+\epsilon}})$ regret:

Theorem 2. *Let $\mathcal{X}$ be a discrete arm set with $|\mathcal{X}| = K \in \mathbb{N}_+$. Under Assumption 1, Algorithm 1 with confidence schedule $\delta_t = \frac{3\delta}{t^2\pi^2 K}$ guarantees, with probability at least $1-\delta$:*

$$\textit{Regret}(T) \leqslant \Delta_0 + CT^{\frac{1}{1+\epsilon}}\left[v_\epsilon\left(2\log T + \log\frac{\pi^2 K}{3\delta}\right)\right]^{\frac{\epsilon}{1+\epsilon}},$$

where $C = (1+\epsilon)\left(2\sqrt{2}+\frac{5}{3}\right)\|f\|_\infty$, and Δ_0 is the instantaneous regret of the initial arm $\boldsymbol{x}_0$.

This yields a $\widetilde{\mathcal{O}}(\sqrt{T})$ regret when $\epsilon = 1$.

6.2. Compact arm set

Now, we consider the more general case of a compact arm set $\mathcal{X} \in \mathbb{R}^d$. This case is also more interesting as it allows to tackle virtually any RL task. We can assume, w.l.o.g., that $\mathcal{X}$ is entirely contained in a box $[-D, D]^d$, with $D \in \mathbb{R}_+$. We also need the following assumption on the expected payoff:

Assumption 2. *The expected payoff μ is Lipschitz continuous, i.e., there exists a constant $L > 0$ such that, for every $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$:*

$$|\mu(\boldsymbol{x}') - \mu(\boldsymbol{x})| \leqslant L\left\|\boldsymbol{x}-\boldsymbol{x}'\right\|_1.$$

This assumption is easily satisfied for policy optimization, as shown in the following:

Lemma 3. *In the policy optimization problem, Assumption 2 can be replaced by:*

$$\sup_{s\in\mathcal{S},\boldsymbol{\theta}\in\Theta}\mathop{\mathbb{E}}_{a\sim\pi_{\boldsymbol{\theta}}}\left[|\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a|s)|\right] \leqslant \boldsymbol{u}_1, \qquad (20)$$

in the action-based paradigm, and by:

$$\sup_{\boldsymbol{\xi}\in\Xi}\mathop{\mathbb{E}}_{\boldsymbol{\theta}\sim\rho_{\boldsymbol{\xi}}}\left[|\nabla_{\boldsymbol{\xi}}\log\rho_{\boldsymbol{\xi}}(\boldsymbol{\theta})|\right] \leqslant \boldsymbol{u}_2, \qquad (21)$$

in the parameter-based paradigm, where $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are d-dimensional vectors and the inequalities are component-wise.

In the proof, we show how to derive the corresponding Lipschitz constants, and show how (20) and (21) are satisfied by the commonly-used Gaussian policy and hyperpolicy, respectively. This is enough for OPTIMIST to achieve $\widetilde{\mathcal{O}}(d^{\frac{\epsilon}{1+\epsilon}}T^{\frac{1}{1+\epsilon}})$ regret:

Algorithm 2 OPTIMIST 2

```
Input: initial arm $\boldsymbol{x}_0$, confidence schedule $(\delta_t)_{t=1}^T$, discretization schedule $(\tau_t)_{t=1}^T$, order $\epsilon \in (0,1]$
Draw sample $z_0 \sim p_{\boldsymbol{x}_0}$ and observe payoff $f(z_0)$
for $t = 1, \dots, T$ do
    Discretize $\mathcal{X}$ with a uniform grid $\widetilde{\mathcal{X}}_t$ of $\tau_t^d$ points
    Select arm $\boldsymbol{x}_t = \arg\max_{\boldsymbol{x}\in\widetilde{\mathcal{X}}_t} B_t^\epsilon(\boldsymbol{x}, \delta_t)$
    Draw sample $z_t \sim p_{\boldsymbol{x}_t}$ and observe payoff $f(z_t)$
end for
```

Theorem 3. *Let $\mathcal{X}$ be a d-dimensional compact arm set with $\mathcal{X} \subseteq [-D, D]^d$. Under Assumptions 1 and 2, Algorithm 1 with confidence schedule $\delta_t = \frac{6\delta}{\pi^2 t^2(1+d^d t^{2d})}$ guarantees, with probability at least $1-\delta$:*

$$\textit{Regret}(T) \leqslant \Delta_0 + CT^{\frac{1}{1+\epsilon}}\left[v_\epsilon\left(2(d+1)\log T + d\log d + \log\frac{\pi^2}{3\delta}\right)\right]^{\frac{\epsilon}{1+\epsilon}} + \frac{\pi^2 LD}{6},$$

where $C = (1+\epsilon)\left(2\sqrt{2}+\frac{5}{3}\right)\|f\|_\infty$, and Δ_0 is the instantaneous regret of the initial arm $\boldsymbol{x}_0$.

This yields a $\widetilde{\mathcal{O}}(\sqrt{dT})$ regret when $\epsilon = 1$. Unfortunately, the optimization step may be very time-consuming. In some applications, we can assume the time required to draw samples to dominate the computational time. In fact, drawing a sample (Algorithm 1, line 2) corresponds to generating a whole trajectory of experience, which may take a long time, especially in real-world applications.

6.3. Discretization

When optimization over the infinite arm space $\mathcal{X}$ is not feasible, Algorithm 2 can be used instead. This variant restricts the optimization to a progressively finer grid $\widetilde{\mathcal{X}}_t$ of $(\tau_t)^d$ vertices. A reasonably coarse discretization schedule can be used at the price of a worse (but still sublinear) regret:

Theorem 4. *Let $\mathcal{X}$ be a d-dimensional compact arm set with $\mathcal{X} \subseteq [-D, D]^d$. For any $\kappa \geqslant 2$, under Assumptions 1 and 2, Algorithm 2 with confidence schedule $\delta_t = \frac{6\delta}{\pi^2 t^2\left(1+\lceil t^{1/\kappa}\rceil^d\right)}$ and discretization schedule $\tau_t = \lceil t^{\frac{1}{\kappa}}\rceil$ guarantees, with probability at least $1-\delta$:*

$$\textit{Regret}(T) \leqslant \Delta_0 + C_1 T^{\left(1-\frac{1}{\kappa}\right)}d + C_2 T^{\frac{1}{1+\epsilon}} \cdot \left[v_\epsilon\left((2+{}^{d}\!/\!_{\kappa})\log T + d\log 2 + \log\frac{\pi^2}{3\delta}\right)\right]^{\frac{\epsilon}{1+\epsilon}},$$

where $C_1 = \frac{\kappa}{\kappa-1}LD$, $C_2 = (1+\epsilon)\left(2\sqrt{2}+\frac{5}{3}\right)\|f\|_\infty$, and Δ_0 is the instantaneous regret of the initial arm $\boldsymbol{x}_0$.

Let us focus on the case $\epsilon = 1$, which is the only one of practical interest in the scope of this paper. For $\kappa = 2$, we

[5] We consider the evaluation of pdf's, payoffs and Rényi divergences in (16) atomic, as it is heavily problem-dependent.

obtain regret $\widetilde{\mathcal{O}}(d\sqrt{T})$. Unfortunately, the time required for optimization is exponential in arm space dimensionality d. For $d \geqslant 2$, we can break the curse of dimensionality by taking $\kappa = d$. In this case, the regret is $\widetilde{\mathcal{O}}\left(dT^{\left(1-\frac{1}{d}\right)}\right)$. On the other hand, the time per iteration is only $O(t^2)$. Note that the regret is sublinear for any choice of κ. Going further: for any $\zeta > 0$, $\kappa = \frac{d}{\zeta}$ grants $O(t^{1+\zeta})$ time per iteration at the cost of $\widetilde{\mathcal{O}}\left(dT^{\left(1-\frac{\zeta}{d}\right)}\right)$ regret.[6]

7. Related Works

In this section, we briefly survey related works from the literature.

Finite-Arms Bandits Exploiting particular arm structures is a common trend in the MAB literature. Correlated bandit methods assume dependencies among arms, either through a subdivision of the arms in clusters (Pandey et al., 2007; Wang et al., 2018) or through the dependency of the expected payoffs on a global latent variable (Mersereau et al., 2009; Atan et al., 2015). The arm correlation we model in Section 4, instead, is based on the effects the arms have on a shared stochastic process. This is closer in spirit to the work of Kallus (2018), in which the selection influences, but does not completely determine, the arm that is actually pulled. Also related is the concept of probabilistically triggered arms in combinatorial bandits (Cesa-Bianchi & Lugosi, 2012; Saritaç & Tekin, 2017; Chen et al., 2016).

Continuous Bandits In continuous bandits, it is necessary to exploit some sort of structure. In linear bandits (Auer, 2002), the expected payoff is a linear function of the selected arm. The OFU principle can be applied to the unknown linear coefficients, estimated with ridge regression (Abbasi-Yadkori et al., 2011). Unfortunately, the linearity assumption is too stringent for most applications. More general frameworks make Lipschitz or Hölder continuity assumptions and often resolve to clever discretization schedules combined with UCB-like strategies (Kleinberg, 2005; Auer et al., 2007; Kleinberg et al., 2008; Bubeck et al., 2009), obtaining $\widetilde{\mathcal{O}}(\sqrt{T})$ regret in some cases. Srinivas et al. (2010) make the assumption that the payoff function has low RKHS complexity, and use Gaussian processes to model uncertainty, achieving $\widetilde{\mathcal{O}}(\sqrt{dT})$ regret. The main advantage of our framework is that the necessary technical assumptions are easily met in the context of policy optimization.

Reinforcement Learning Although there is a long history of rigorously applying the OFU principle to tabular RL (Kearns & Singh, 2002; Brafman & Tennenholtz, 2002; Strehl et al., 2009; Jaksch et al., 2010; Lattimore & Hutter, 2014; Dann & Brunskill, 2015; Dann et al., 2017; Jin et al., 2018; Ok et al., 2018), with extensions to continuous states (Ortner & Ryabko, 2012; Lakshmanan et al., 2015; Bellemare et al., 2016), optimistic approaches to continuous-action MDPs[7] remain largely heuristic (Houthooft et al., 2016; Haarnoja et al., 2017; 2018). Developing ideas from Bubeck & Munos (2010), Weinstein & Littman (2012) apply continuous bandit techniques to open-loop iterative planning, a model-based approach to RL. In the model-free setting, Chowdhury & Gopalan (2018) prove $\widetilde{\mathcal{O}}(\sqrt{T})$ regret for kernelized MDPs, where rewards and transitions are assumed to have low RKHS complexity, leaving some computational problems open. Our proposed algorithms are model-free and do not make assumptions on the MDP, besides boundedness of the reward. Moreover, Algorithm 2 applied to parameter-based policy optimization allows a straightforward and efficient implementation. Thompson sampling (TS, Thompson, 1933) is a different approach to MABs, not based on optimism, which enjoys the same theoretical guarantees of UCB (Kaufmann et al., 2012) with better performance in many applications (Chapelle & Li, 2011). TS was applied to value-based RL (e.g., Osband et al., 2013), and its application to policy optimization could also be fruitful.

8. Numerical Simulations

In this section, we present the results of the numerical simulation of OPTIMIST on RL tasks on both discrete and continuous parameter spaces. We restrict our experiments to the *parameter-based* PS and Gaussian hyperpolicies. This setting is particularly convenient as the Rényi divergence between Gaussian distributions admits closed form (Gil et al., 2013). On the contrary, in the action-based scenario, we would need to compute the divergences between trajectory distributions, which is intractable. The usual approach consists in estimating the Rényi divergence from the samples. However, we would lose our theoretical guarantees on the regret. Furthermore, the known estimators for the Rényi divergence tend empirically to be unstable (Metelli et al., 2018). It is worth noting that at each iteration OPTIMIST needs to compute the Rényi divergence between a candidate hyperpolicy $\nu_{\boldsymbol{\xi}_t}$ and the mixture of hyperpolicies visited so far $\nu_{\boldsymbol{\xi}_0}, ..., \nu_{\boldsymbol{\xi}_{t-1}}$. We prove in Appendix A that this quantity can be upper bounded by the harmonic mean of the divergences between the candidate hyperpolicy $\nu_{\boldsymbol{\xi}_t}$ and each component of the mixture $\nu_{\boldsymbol{\xi}_k}$ for $k = 1, ..., t-1$.

8.1. Linear Quadratic Gaussian Regulator

The Linear Quadratic Gaussian Regulator (LQG, Dorato et al., 1995) is a benchmark problem for continuous con-

[6] The worse dependency $\widetilde{\mathcal{O}}(d)$ of the regret on the arm space dimensionality (w.r.t. $\widetilde{\mathcal{O}}(\sqrt{d})$ of Algorithm 1) is also necessary to prevent the time per iteration from being exponential in d.

[7] Note that, although we discretize the policy parameter space in some occasions, we always deal with continuous action spaces.

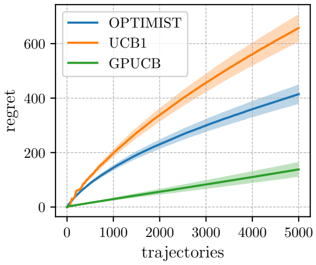

Figure 1. Cumulative regret in the LQG experiment, comparing OPTIMIST, UCB1 and GPUCB (30 runs, 95% c.i.)

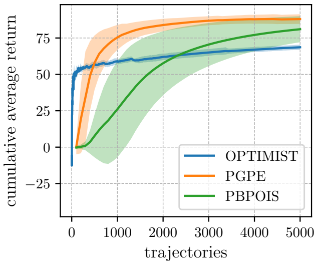

Figure 2. Cumulative average return for the Mountain Car, comparing OPTIMIST, PGPE and PB-POIS (5 runs, 95% c.i.)

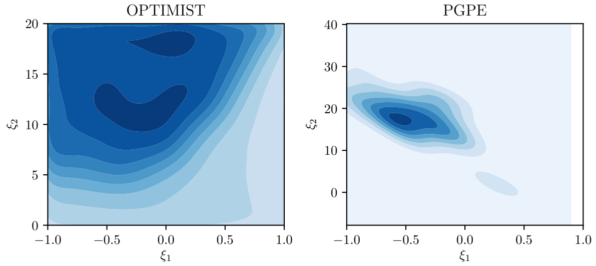

Figure 3. Heatmaps representing the region of the parameter space Ξ explored by OPTIMIST and PGPE in the Mountain Car experiment.

trol. We consider the monodimensional case in which the state space limited to $\mathcal{S} = [-4, 4]$, the action space is $\mathcal{A} = [-4, 4]$ and the horizon is limited to 20. At each timestep, the agent receives a penalization proportional to the magnitude of the state and the action applied, i.e., $R(s, a) = -as^2 - ba^2$. We employ a Gaussian hyperpolicy $\nu_\xi = \mathcal{N}(\xi, \sigma^2)$, where ξ is the mean parameter to be learned and $\sigma = 0.15$ fixed. The case in which we also learn the standard deviation is reported in Appendix C. The Linear Quadratic Gaussian Regulator (LQG, Dorato et al., 1995) is a benchmark problem for continuous control. We consider the monodimensional case in which the state space restricted to $\mathcal{S} = [-4, 4]$, the action space is $\mathcal{A} = [-4, 4]$ and the horizon is limited to 20. We employ a Gaussian hyperpolicy $\nu_\xi = \mathcal{N}(\xi, \sigma^2)$, where ξ is the mean parameter to be learned and $\sigma = 0.15$ fixed. The case in which we learn the standard deviation too is reported in Appendix C.

The goal of this experiment is to compare OPTIMIST with classical MAB algorithms, in particular UCB1 (Auer et al., 2002) and GPUCB (Srinivas et al., 2010) when the parameter space Ξ is discrete. For this purpose we consider a uniform discretization of the interval $[-1, 1]$ made of 100 arms. All algorithms are run with confidence level $\delta = 0.2$. In Figure 1, we show the cumulative regret averaged over 30 runs of OPTIMIST compared with UCB1 and GPUCB. We can see that our algorithm significantly outperforms UCB1. Indeed, OPTIMIST is able to exploit the structure of arms, i.e., hyperpolicies, by means of the MIS estimation, whereas UCB1 does not make any assumption on arm correlation. On the contrary, GPUCB shows a better performance w.r.t. to OPTIMIST. We point out that GPUCB requires to specify, at the beginning of learning, the kernel of the Gaussian Process (GP) from which the payoff function is sampled. We employed the default scikit-learn kernel (RBF) which invalidates all theoretical guarantees, as our payoff is not actually sampled from a GP.[8]

8.2. Mountain Car

The second experiment, shown in Figure 2, illustrates the behavior of OPTIMIST when the parameters of the hyperpolicy belong to a compact (continuous) space, on the Mountain Car tas (Brockman et al., 2016). We use a Gaussian hyperpolicy with a two-dimensional learnable mean within a box $[-1, -1] \times [0, 20]$ and a fixed covariance $\mathrm{diag}(0.15, 3)$. We compare OPTIMIST2 with $k = 3$ and $\delta = 0.2$ against parameter based policy optimization algorithms PGPE (Sehnke et al., 2008) and PB-POIS (Metelli et al., 2018). The best step size for PGPE was searched in the set $[3, 2, 1, 0.1, 0.01, 0.001]$. For PBPOIS, we used the suggested hyperparameters. We can notice that OPTIMIST2 is able to learn a good policy in a very short time thanks to its better exploration capabilities (Figure 3). However, the policy gradient methods outperform it on the long run.

9. Conclusion

We have studied the problem of exploration versus exploitation in policy optimization using MAB techniques. We have proposed OPTIMIST, an optimism-based approach for both the action-based and the parameter-based exploration frameworks, and for both discrete and continuous parameter spaces. We have proved sublinear regret bounds for OPTIMIST under assumptions that are easily met in practice. The empirica evaluation on continuous control tasks showed that the proposed algorithms are effectively able to leverage the structure of the PO problem, although the performances are not optimal when compared to methods with stronger assumptions or without guarantees, at least on these simple problems. Future work should focus on finding more efficient (but still effective) ways to perform optimization in the infinite-arm setting, and on applying OPTIMIST also to the action-based framework, which requires additional caveats in computing the exploration bonus.

[8] Indeed, GPUCB showed a significantly more exploitative behavior w.r.t. UCB1 and OPTIMIST in the experiment.

References

Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pp. 2312–2320, 2011.

Agrawal, R. Sample mean based index policies by o (log n) regret for the multi-armed bandit problem. *Advances in Applied Probability*, 27(4):1054–1078, 1995.

Atan, O., Tekin, C., and Schaar, M. Global multi-armed bandits with hölder continuity. In *Artificial Intelligence and Statistics*, pp. 28–36, 2015.

Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

Auer, P., Cesa-Bianchi, N., and Fischer, P. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

Auer, P., Ortner, R., and Szepesvári, C. Improved rates for the stochastic continuum-armed bandit problem. In *International Conference on Computational Learning Theory*, pp. 454–468. Springer, 2007.

Bellemare, M., Srinivasan, S., Ostrovski, G., Schaul, T., Saxton, D., and Munos, R. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*, pp. 1471–1479, 2016.

Boucheron, S., Lugosi, G., and Massart, P. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013.

Brafman, R. I. and Tennenholtz, M. R-max-a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3(Oct): 213–231, 2002.

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

Bubeck, S. and Munos, R. Open loop optimistic planning. In *COLT 2010 - The 23rd Conference on Learning Theory, Haifa, Israel, June 27-29, 2010*, pp. 477–489, 2010.

Bubeck, S., Stoltz, G., Szepesvári, C., and Munos, R. Online optimization in x-armed bandits. In *Advances in Neural Information Processing Systems*, pp. 201–208, 2009.

Bubeck, S., Cesa-Bianchi, N., et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5 (1):1–122, 2012.

Bubeck, S., Cesa-Bianchi, N., and Lugosi, G. Bandits with heavy tail. *IEEE Transactions on Information Theory*, 59 (11):7711–7717, 2013.

Cesa-Bianchi, N. and Lugosi, G. Combinatorial bandits. *Journal of Computer and System Sciences*, 78(5):1404–1422, 2012.

Chapelle, O. and Li, L. An empirical evaluation of thompson sampling. In *Advances in neural information processing systems*, pp. 2249–2257, 2011.

Chen, W., Wang, Y., Yuan, Y., and Wang, Q. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *The Journal of Machine Learning Research*, 17(1):1746–1778, 2016.

Chowdhury, S. R. and Gopalan, A. Online learning in kernelized markov decision processes. *CoRR*, abs/1805.08052, 2018.

Cochran, W. G. *Sampling techniques*. John Wiley & Sons, 2007.

Cortes, C., Mansour, Y., and Mohri, M. Learning bounds for importance weighting. In Lafferty, J. D., Williams, C. K. I., Shawe-Taylor, J., Zemel, R. S., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems 23*, pp. 442–450. Curran Associates, Inc., 2010.

Dann, C. and Brunskill, E. Sample complexity of episodic fixed-horizon reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 2818–2826, 2015.

Dann, C., Lattimore, T., and Brunskill, E. Unifying pac and regret: Uniform pac bounds for episodic reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 5713–5723, 2017.

Deisenroth, M. P., Neumann, G., Peters, J., et al. A survey on policy search for robotics. *Foundations and Trends® in Robotics*, 2(1–2):1–142, 2013.

Dorato, P., Abdallah, C. T., Cerone, V., and Jacobson, D. H. *Linear-quadratic control: an introduction*. Prentice Hall Englewood Cliffs, NJ, 1995.

Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I., Legg, S., and Kavukcuoglu, K. IMPALA: scalable distributed deep-rl with importance weighted actor-learner architectures. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pp. 1406–1415, 2018.

Gil, M., Alajaji, F., and Linder, T. Rényi divergence measures for commonly used univariate continuous distributions. *Information Sciences*, 249:124–131, 2013.

Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. Reinforcement learning with deep energy-based policies. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pp. 1352–1361, 2017.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pp. 1856–1865, 2018.

Hershey, J. R. and Olsen, P. A. Approximating the kullback leibler divergence between gaussian mixture models. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pp. IV–317. IEEE, 2007.

Houthooft, R., Chen, X., Duan, Y., Schulman, J., De Turck, F., and Abbeel, P. Vime: Variational information maximizing exploration. In *Advances in Neural Information Processing Systems*, pp. 1109–1117, 2016.

Ionides, E. L. Truncated importance sampling. *Journal of Computational and Graphical Statistics*, 17(2):295–311, 2008.

Jaksch, T., Ortner, R., and Auer, P. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(Apr):1563–1600, 2010.

Jin, C., Allen-Zhu, Z., Bubeck, S., and Jordan, M. I. Is q-learning provably efficient? In *Advances in Neural Information Processing Systems*, pp. 4868–4878, 2018.

Kallus, N. Instrument-armed bandits. In *Algorithmic Learning Theory*, pp. 529–546, 2018.

Kaufmann, E., Korda, N., and Munos, R. Thompson sampling: An asymptotically optimal finite-time analysis. In *International Conference on Algorithmic Learning Theory*, pp. 199–213. Springer, 2012.

Kearns, M. and Singh, S. Near-optimal reinforcement learning in polynomial time. *Machine learning*, 49(2-3):209–232, 2002.

Kleinberg, R., Slivkins, A., and Upfal, E. Multi-armed bandits in metric spaces. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp. 681–690. ACM, 2008.

Kleinberg, R., Slivkins, A., and Upfal, E. Bandits and experts in metric spaces. *arXiv preprint arXiv:1312.1277*, 2013.

Kleinberg, R. D. Nearly tight bounds for the continuum-armed bandit problem. In *Advances in Neural Information Processing Systems*, pp. 697–704, 2005.

Lai, T. L. and Robbins, H. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1): 4–22, 1985.

Lakshmanan, K., Ortner, R., and Ryabko, D. Improved regret bounds for undiscounted continuous reinforcement learning. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pp. 524–532, 2015.

Lattimore, T. and Hutter, M. Near-optimal pac bounds for discounted mdps. *Theoretical Computer Science*, 558: 125–143, 2014.

Lattimore, T. and Szepesvári, C. *Bandit Algorithms*. Cambridge University Press (preprint), 2019.

Mersereau, A. J., Rusmevichientong, P., and Tsitsiklis, J. N. A structured multiarmed bandit problem and the greedy policy. *IEEE Transactions on Automatic Control*, 54(12): 2787–2802, 2009.

Metelli, A. M., Papini, M., Faccio, F., and Restelli, M. Policy optimization via importance sampling. In *Advances in Neural Information Processing Systems*, pp. 5447–5459, 2018.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T. P., Harley, T., Silver, D., and Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pp. 1928–1937, 2016.

Ok, J., Proutière, A., and Tranos, D. Exploration in structured reinforcement learning. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pp. 8888–8896, 2018.

Ortner, R. and Ryabko, D. Online regret bounds for undiscounted continuous reinforcement learning. In *Proceedings of the 25th International Conference on Neural Information Processing Systems-Volume 2*, pp. 1763–1771. Curran Associates Inc., 2012.

Osband, I., Russo, D., and Roy, B. V. (more) efficient reinforcement learning via posterior sampling. In *Advances in Neural Information Processing Systems 26:*

27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States., pp. 3003–3011, 2013.

Owen, A. B. *Monte Carlo theory, methods and examples*. 2013.

Pandey, S., Chakrabarti, D., and Agarwal, D. Multi-armed bandit problems with dependent arms. In *Proceedings of the 24th international conference on Machine learning*, pp. 721–728. ACM, 2007.

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

Puterman, M. L. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Rényi, A. On measures of entropy and information. Technical report, Hungarian Academy of Sciences Budapest Hungary, 1961.

Robbins, H. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pp. 169–177. Springer, 1985.

Saritaç, A. Ö. and Tekin, C. Combinatorial multi-armed bandit problem with probabilistically triggered arms: A case with bounded regret. In *Signal and Information Processing (GlobalSIP), 2017 IEEE Global Conference on*, pp. 111–115. IEEE, 2017.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *International Conference on Machine Learning*, pp. 1889–1897, 2015.

Sehnke, F., Osendorfer, C., Rückstieß, T., Graves, A., Peters, J., and Schmidhuber, J. Policy gradients with parameter-based exploration for control. In *International Conference on Artificial Neural Networks*, pp. 387–396. Springer, 2008.

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., and Riedmiller, M. Deterministic policy gradient algorithms. In *ICML*, 2014.

Srinivas, N., Krause, A., Kakade, S., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. In Fürnkranz, J. and Joachims, T. (eds.), *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 1015–1022, Haifa, Israel, June 2010. Omnipress.

Strehl, A. L., Li, L., and Littman, M. L. Reinforcement learning in finite mdps: Pac analysis. *Journal of Machine Learning Research*, 10(Nov):2413–2444, 2009.

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

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

Thompson, W. R. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

Veach, E. and Guibas, L. J. Optimally combining sampling techniques for Monte Carlo rendering. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques - SIGGRAPH '95*, pp. 419–428. ACM Press, 1995.

Wang, Z., Zhou, R., and Shen, C. Regional multi-armed bandits. In *International Conference on Artificial Intelligence and Statistics*, pp. 510–518, 2018.

Weinstein, A. and Littman, M. L. Bandit-based planning and learning in continuous-action markov decision processes. In *Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, ICAPS 2012, Atibaia, São Paulo, Brazil, June 25-19, 2012*, 2012.