LETTER Communicated by Maxim Raginsky

An Online Policy Gradient Algorithm for Markov Decision Processes with Continuous States and Actions

Yao Ma
mycw45@gmail.com
Tokyo Institute of Technology, Meguro, Tokyo 152-8552, Japan

Tingting Zhao
tingting@tust.edu.cn
Tianjian University of Science and Technology, Tian-Jin 30022, China

Kohei Hatano
hatano@inf.kyushu-u.ac.jp
Kyushu University, Fukuoka 819-0395, Japan

Masashi Sugiyama
sugi@k.u-tokyo.ac.jp
University of Tokyo, Tokyo 113-0033, Japan

We consider the learning problem under an online Markov decision process (MDP) aimed at learning the time-dependent decision-making policy of an agent that minimizes the regret—the difference from the best fixed policy. The difficulty of online MDP learning is that the reward function changes over time. In this letter, we show that a simple online policy gradient algorithm achieves regret $O(\sqrt{T})$ for T steps under a certain concavity assumption and $O(\log T)$ under a strong concavity assumption. To the best of our knowledge, this is the first work to present an online MDP algorithm that can handle continuous state, action, and parameter spaces with guarantee. We also illustrate the behavior of the proposed online policy gradient method through experiments.

1 Introduction

The Markov decision process (MDP) is a popular framework of reinforcement learning for sequential decision making (Sutton & Barto, 1998), where an agent takes an action depending on the current state, moves to the next state, receives a reward based on the last transition, and this process is repeated T times. The goal is to find an optimal decision-making policy (i.e.,

This letter is an expanded version of our earlier work presented at ECML/PKDD 2014 (Ma, Zhao, Hatano, & Sugiyama, 2014).

Neural Computation **28, 563–593** (2016)
doi:10.1162/NECO_a_00808

a conditional probability density of action given state) that maximizes the expected sum of rewards over T steps.

In the standard MDP formulation, the reward function is fixed over iterations. However, this assumption is often violated in reality. In this letter, we consider an online MDP scenario where the reward function is allowed to change over time. Such an online MDP problem is an extension of both online decision making and reinforcement learning (Yu, Mannor, & Shimkin, 2009):

- In an online decision-making problem, the agent needs to make a decision at each time step without knowledge of the future environment (Kalai & Vempala, 2005). A certain cost function will be observed only after the decision is made at each time step, and the goal is to minimize the regret against the best single decision. There is no assumption on the dynamics in the online decision making problem, and thus the decision can switch from one to another abruptly.
- In reinforcement learning, the dynamics are assumed to be Markovian. The reward function and transition dynamics are fixed but unknown to the agent, and thus the estimated reward function and transition function will converge to the true ones if sufficient samples are observed. The goal is to find the optimal policy that maximizes the cumulative reward without full information about the environment.

The goal of the online MDP problem is to find the best time-dependent policy that minimizes the regret, the difference from the best fixed policy. We expect the regret to be $o(T)$, by which the difference from the best fixed policy vanishes as T goes to infinity.

The MDP expert algorithm (MDP-E), which chooses the current best action at each state, was shown to achieve regret $O(\sqrt{T \log |A|})$ (Even-Dar, Kakade, & Mansour, 2004, 2009), where $|A|$ denotes the cardinality of the action space. Although this bound does not explicitly depend on the cardinality of the state space, the algorithm itself needs an expert algorithm for each state, and thus large state space may not be handled in practice. Another algorithm, called the lazy follow-the-perturbed-leader (lazy-FPL), divides the time steps into short periods, and policies are updated only at the end of each period using the average reward function (Yu et al., 2009). This lazy-FPL algorithm was shown to have regret $O(T^{3/4+\epsilon} \log T(|S| + |A|)|A|^2)$ for $\epsilon \in (0, 1/3)$. The online MDP algorithm, called the online relative entropy policy search, is considered in Zimin and Neu (2013), which was shown to have regret $O(L^2\sqrt{T \log(|S||A|/L)})$ for state space with L-layered structure. However, the regret bounds of these algorithms explicitly depend on $|S|$ and $|A|$, and the algorithms cannot be directly implemented for problems with continuous state and action spaces. The online algorithm for Markov decision processes (Abbasi-Yadkori, Bartlett, Kanade, Seldin, & Szepesvari, 2013) was shown to have regret $O(\sqrt{T \log |\Pi|} + \log |\Pi|)$ with changing transition probability distributions, where $|\Pi|$ is the cardinality of the policy set.

Although sublinear bounds still hold for continuous policy spaces, the algorithm cannot be used with infinite policy candidates directly. The online MDP problem is formulated as an online linear optimization problem in Dick, György, and Szepesvári (2014). By introducing the stationary occupation measures, the mirror descent with approximate projections was shown to have regret $O(\sqrt{T})$. However, the algorithm assumes that both the state and action spaces are finite. Yu et al. (2009), Abbasi-Yadkori et al. (2013), and Neu, György, and Szepesvári (2012) considered even more challenging online MDP problems under unknown or changing transition dynamics.

In practice, full information of the reward function may be hard to acquire, but only the value of the reward function for the current state and action is available. Such a setup, called the bandit feedback scenario, has attracted a great deal of attention recently. An extension of the lazy-FPL method to the bandit feedback scenario, called the exploratory-FPL algorithm (Yu et al., 2009), was shown to have regret $o(T)$. Neu, György, Szepesvári, and Antos (2010) proposed a method based on MDP-E that uses an unbiased estimator of the reward function and showed that its regret is $O(T^{2/3}(\ln T)^{1/3}\ln|A|)$. Neu, György, Szepesvári, and Antos (2014) further improved the regret bound to $O(\sqrt{T\ln T\ln|A|})$. However, this algorithm cannot be used in continuous state and action problems.

In this letter, we propose a simple online policy gradient (OPG) algorithm that can be implemented in a straightforward manner for problems with continuous state and action spaces, which could be seen as an extension of Dick et al. (2014).[1] Under the assumption that the expected average reward function is concave, we prove that the regret of our OPG algorithm with respect to a compact and convex parametric policies set is $O(\sqrt{T}(F^2+N))$, which is independent of the cardinality of the state and action spaces but is dependent on the diameter F and dimension N of the parameter space. Furthermore, regret $O(N^2\log T)$ is also proved under a strong concavity assumption on the expected average reward function. We also extend the proposed algorithm to a bandit feedback scenario and theoretically prove that the regret bound of the proposed algorithm is $O(\sqrt{T})$ with the concavity assumption. We numerically illustrate the superior behavior of the proposed OPG algorithm in continuous problems over MDP-E with different discretization schemes.

The remainder of this letter is organized as follows. In section 2, we give a formal definition of the online MDP problem. Our proposed algorithm is given in section 3, and regret analyses in full information and the bandit scenario are given in sections 4 and 5, with proofs presented in the appendix.

[1] Our OPG algorithm can also be seen as an extension of the online gradient descent algorithm (Zinkevich, 2003) to online MDP problems.

2 Online Markov Decision Process

In this section, we formulate the problem of online MDP learning. An online MDP is specified by:

- State space $S \subseteq \mathbb{R}^{D_s}$, which could be either continuous or discrete.
- Action space $A \subseteq \mathbb{R}^{D_a}$, which contains all possible actions $\boldsymbol{a}$. A could be either continuous or discrete.
- Transition density $p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$, which represents the conditional probability density of next state $\boldsymbol{s}'$ given current state $\boldsymbol{s}$ and action $\boldsymbol{a}$ to be taken. We assume that the transition density is fully available to the agent.
- Reward function sequence $r_1, r_2, \ldots, r_T$, which is a pre fixed real-valued function sequence and will not change no matter what action is taken.

An online MDP algorithm produces a stochastic time-dependent policy, a conditional probability density of action $\boldsymbol{a}$ to be taken given current state $\boldsymbol{s}$ at each time step. In this letter, we suppose that the online MDP algorithm $\mathcal{A}$ outputs parameter $\boldsymbol{\theta}_t = [\theta_t^{(1)}, \ldots, \theta_t^{(N)}]^\top \in \Theta \subset \mathbb{R}^N$ of stochastic policy $\pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t)$ at each time step t, where Θ is a convex and compact parameter set. Thus, algorithm $\mathcal{A}$ gives a sequence of policies:

$$\pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_1), \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_2), \ldots, \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_T).$$

Ideally the objective is to maximize the expected cumulative reward over T time steps of algorithm $\mathcal{A}$, which can be denoted as

$$R_{\mathcal{A}}(T) = \mathbb{E}\left[\sum_{t=1}^{T} r_t(\boldsymbol{s}_t, \boldsymbol{a}_t)|\mathcal{A}\right]. \tag{2.1}$$

In the above definition, $\mathbb{E}[\cdot|\mathcal{A}]$ denotes the expectation over the joint state-action distribution $p_t(\boldsymbol{s}, \boldsymbol{a}|\mathcal{A})$ given the algorithm $\mathcal{A}$ has been followed at each time step. The state-action distribution induced by $\mathcal{A}$ and the transition density at time step t can be expressed as

$$p_t(\boldsymbol{s}, \boldsymbol{a}|\mathcal{A}) = d_{\mathcal{A},t}(\boldsymbol{s}) \cdot \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t),$$

where the state distribution induced by $\mathcal{A}$ at time step t is defined as

$$d_{\mathcal{A},t}(\boldsymbol{s}) = p(\boldsymbol{s}_t = \boldsymbol{s}|\mathcal{A}).$$

However, maximizing the objective defined in equation 2.1 is not possible, since we cannot observe all T reward functions during the process of an online decision-making problem. Here, we instead design algorithm $\mathcal{A}$ that minimizes the regret against the baseline, which is the best parametric offline policy defined by

$$L_{\mathcal{A}}(T) = R_{\boldsymbol{\theta}^*}(T) - R_{\mathcal{A}}(T).$$

In this definition of the regret, we suppose that there exists $\boldsymbol{\theta}^*$ such that policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}^*)$ maximizes the expected cumulative rewards:

$$R_{\boldsymbol{\theta}^*}(T) = \mathbb{E}\left[\sum_{t=1}^{T} r_t(\boldsymbol{s}_t, \boldsymbol{a}_t)|\boldsymbol{\theta}^*\right].$$

The best offline parameter $\boldsymbol{\theta}^*$ is given by

$$\boldsymbol{\theta}^* = \operatorname*{argmax}_{\boldsymbol{\theta}\in\Theta} \mathbb{E}\left[\sum_{t=1}^{T} r_t(\boldsymbol{s}_t, \boldsymbol{a}_t)|\boldsymbol{\theta}\right], \tag{2.2}$$

where $\mathbb{E}[\cdot|\boldsymbol{\theta}]$ denotes the expectation over the state-action distribution given that the policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})$ has been followed at each time step.

We assume here that all candidate policies are parameterized by the parameter $\boldsymbol{\theta}$, which is different from related works with finite states and actions (Even-Dar et al., 2004, 2009; Neu, György, Szepesvári, et al., 2010; Yu et al., 2009; Dick et al., 2014). For continuous problems, it is a common choice to use a parametric policy (e.g., the gaussian policy), which was demonstrated to work well (Sutton & Barto, 1998; Peters & Schaal, 2006). For this reason, the best offline policy defined in equation 2.2 is a suitable baseline given that the best policy with respect to the class of all Markovian policies is not a suitable baseline for continuous problems. If the regret is bounded by a sublinear function with respect to T, the algorithm $\mathcal{A}$ is shown to be asymptotically as powerful as the best offline policy.

3 Online Policy Gradient Algorithm

In this section, we introduce an online policy gradient algorithm for solving the online MDP problem.

3.1 Algorithm.

Unlike previous work (Even-Dar et al., 2004, 2009; Neu, György, Szepesvári, et al., 2010), we do not use the expert algorithm in our method because it is not suitable for handling continuous state and action problems. Instead, we consider a gradient-based algorithm that updates the parameter of policy $\boldsymbol{\theta}$ along the gradient direction of the expected average reward function at each time step t.

More specifically, we assume that all the MDPs are ergodic whose state transitions are induced by the transition density $p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$ and the parametric policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}), \forall\boldsymbol{\theta} \in \Theta$. Then every policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})$ has a unique stationary state distribution $d_{\boldsymbol{\theta}}(\boldsymbol{s})$:

$$d_{\boldsymbol{\theta}}(\boldsymbol{s}) = \lim_{t\to\infty} p(\boldsymbol{s}_t = \boldsymbol{s}|\boldsymbol{\theta}).$$

Note that the stationary state distribution satisfies

$$d_{\boldsymbol{\theta}}(\boldsymbol{s}') = \int_{\boldsymbol{s}\in S} d_{\boldsymbol{\theta}}(\boldsymbol{s}) \int_{\boldsymbol{a}\in A} \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}) p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a}) \mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s}.$$

Let $\rho_t(\boldsymbol{\theta})$ be the expected average reward function of policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})$ at time step t:

$$\begin{aligned}\rho_t(\boldsymbol{\theta}) &= \mathbb{E}_{\boldsymbol{s}\sim d_{\boldsymbol{\theta}}(\boldsymbol{s}), \boldsymbol{a}\sim\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})}[r_t(\boldsymbol{s}, \boldsymbol{a})] \\ &= \int_{\boldsymbol{s}\in S} d_{\boldsymbol{\theta}}(\boldsymbol{s}) \int_{\boldsymbol{a}\in A} r_t(\boldsymbol{s}, \boldsymbol{a}) \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}) \mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s}, \end{aligned} \tag{3.1}$$

where the expectation is taken over the stationary state-action distribution of policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})$.

Then our online policy gradient (OPG) algorithm is given as follows:

- Initialize policy parameter $\boldsymbol{\theta}_1$.
- for $t = 1$ to ∞
  1. Observe current state $\boldsymbol{s}_t = \boldsymbol{s}$.
  2. Take action $\boldsymbol{a}_t = \boldsymbol{a}$ according to current policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}_t)$.
  3. Observe reward r_t from the environment.
  4. Move to next state $\boldsymbol{s}_{t+1}$.
  5. Update the policy parameter as

     $$\boldsymbol{\theta}_{t+1} = P(\boldsymbol{\theta}_t + \eta_t \nabla_{\boldsymbol{\theta}} \rho_t(\boldsymbol{\theta}_t)), \tag{3.2}$$

     where $P(\boldsymbol{\vartheta}) = \arg\min_{\boldsymbol{\theta}\in\Theta} \|\boldsymbol{\vartheta} - \boldsymbol{\theta}\|$ is the projection function on parameter space, $\|\cdot\|$ denotes the Euclidean norm. $\eta_t = \frac{1}{\sqrt{t}}$ is the step size, and $\nabla_{\boldsymbol{\theta}}\rho_t(\boldsymbol{\theta})$ is the gradient of $\rho_t(\boldsymbol{\theta})$:

     $$\begin{aligned}\nabla_{\boldsymbol{\theta}}\rho_t(\boldsymbol{\theta}) &\equiv \left[\frac{\partial\rho_t(\boldsymbol{\theta})}{\partial\theta^{(1)}}, \ldots, \frac{\partial\rho_t(\boldsymbol{\theta})}{\partial\theta^{(N)}}\right]^\top \\ &= \int_{\boldsymbol{s}\in S}\int_{\boldsymbol{a}\in A} d_{\boldsymbol{\theta}}(\boldsymbol{s})\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})(\nabla_{\boldsymbol{\theta}} \ln d_{\boldsymbol{\theta}}(\boldsymbol{s}) + \nabla_{\boldsymbol{\theta}} \ln \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})) \\ &\quad \times r_t(\boldsymbol{s}, \boldsymbol{a})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s}. \end{aligned} \tag{3.3}$$

In equation 3.3, the facts $\nabla_{\boldsymbol{\theta}} \ln d_{\boldsymbol{\theta}}(\boldsymbol{s}) = \frac{\nabla_{\boldsymbol{\theta}} d_{\boldsymbol{\theta}}(\boldsymbol{s})}{d_{\boldsymbol{\theta}}(\boldsymbol{s})}$ and $\nabla_{\boldsymbol{\theta}} \ln \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}) = \frac{\nabla_{\boldsymbol{\theta}}\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})}{\pi(\boldsymbol{a}|\boldsymbol{s};\theta)}$ are used. Here we assume that $\nabla_{\boldsymbol{\theta}} d_{\boldsymbol{\theta}}(\boldsymbol{s})$ and $\nabla_{\boldsymbol{\theta}}\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})$ are differentiable with respect to the policy parameter $\boldsymbol{\theta}$. If it is time-consuming to obtain the exact stationary state distribution, gradients estimated by a reinforcement learning algorithm may be used instead in practice. Since the transition and reward functions are known to the agent, it is straightforward to estimate the gradient efficiently by using a reinforcement learning technique (e.g., REINFORCE and policy gradients with parameter-based exploration; Sutton & Barto, 1998; Williams, 1992; Sehnke et al., 2010). Furthermore, some reinforcement learning techniques provided a convergence guarantee for the gradient estimation. Especially in the REINFORCE algorithm, the gradient is approximated by the empirical average value $\nabla_{\boldsymbol{\theta}}\bar{\rho}_t(\boldsymbol{\theta})$ after sufficient trajectories are collected as

$$\nabla_{\boldsymbol{\theta}}\bar{\rho}_t(\boldsymbol{\theta}) = \frac{1}{|H|}\sum_{n=1}^{|H|}\sum_{i=1}^{L}\nabla_{\boldsymbol{\theta}}\log\pi(\boldsymbol{a}_i|\boldsymbol{s}_i;\boldsymbol{\theta})R(\boldsymbol{h}_n),$$

where $\boldsymbol{h}_n$ is a rollout sample denoted as $\boldsymbol{h}_n = [\boldsymbol{s}_1, \boldsymbol{a}_1, \ldots, \boldsymbol{s}_L, \boldsymbol{a}_L]$, $H = \{\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_{|H|}\}$ is the set of collected trajectories with length L, and $R(\boldsymbol{h}_n)$ is the average reward obtained by trajectory $\boldsymbol{h}_n$. With theoretical guarantee, the REINFORCE algorithm has been shown to converge to the true gradient as $|H|$ and L tend to infinity. In the following analysis, we ignore the approximation error since it could be arbitrarily small by collecting a large enough number of samples.

When the reward function does not changed over time, the OPG algorithm is reduced to the ordinary policy gradient algorithm (Williams, 1992), an efficient and natural algorithm for continuous state and action MDPs. The OPG algorithm can also be regarded as an extension of the online gradient descent algorithm (Zinkevich, 2003), which maximizes $\sum_{t=1}^{T}\rho_t(\boldsymbol{\theta}_t)$, not $\mathbb{E}[\sum_{t=1}^{T} r_t(\boldsymbol{s}_t, \boldsymbol{a}_t)|\mathcal{A}]$. As we showed in the definition of $\rho_t(\boldsymbol{\theta}_t)$, the stationary state distribution $d_{\boldsymbol{\theta}_t}(\boldsymbol{s})$ of policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}_t)$ is used, which is different from the state distribution $d_{\mathcal{A},t}(\boldsymbol{s})$ used in $\mathbb{E}[\sum_{t=1}^{T} r_t(\boldsymbol{s}_t, \boldsymbol{a}_t)|\mathcal{A}]$. As we will prove in section 4, the regret bound of the OPG algorithm is $O(\sqrt{T})$ under a certain concavity assumption and $O(\log T)$ under a strong concavity assumption on the expected average reward function. Unlike previous work (Even-Dar et al., 2004, 2009; Yu et al., 2009; Neu, György, Szepesvári, et al., 2010), these bounds do not depend on the cardinality of state and action spaces since a parameterized policy space is considered. Therefore, the OPG algorithm would be suitable for handling continuous state and action online MDPs.

3.2 Bandit Feedback. Here we extend the OPG algorithm to the bandit feedback scenario, where the *entire* reward function is not available; only the value of the reward function for the current state and action is observed:

$$\boldsymbol{s}_1, \boldsymbol{a}_1, r_1(\boldsymbol{s}_1, \boldsymbol{a}_1), \ldots, \boldsymbol{s}_t, \boldsymbol{a}_t, r_t(\boldsymbol{s}_t, \boldsymbol{a}_t).$$

Due to lack of the entire reward function, we replace reward function r_t in the OPG algorithm with a random reward function given by

$$\hat{r}_t(\boldsymbol{s}, \boldsymbol{a}) = \frac{r_t(\boldsymbol{s}, \boldsymbol{a})}{d_{\mathcal{A},t}(\boldsymbol{s})\pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t)} \delta(\boldsymbol{s}_t = \boldsymbol{s}, \boldsymbol{a}_t = \boldsymbol{a}), \tag{3.4}$$

where $d_{\mathcal{A},t}(\boldsymbol{s})$ can be calculated recursively using the following equation:

$$d_{\mathcal{A},t}(\boldsymbol{s}) = \int_{\boldsymbol{s}' \in S} \int_{\boldsymbol{a} \in A} d_{\mathcal{A},t-1}(\boldsymbol{s}')\pi(\boldsymbol{a}|\boldsymbol{s}'; \boldsymbol{\theta}_{t-1}) p(\boldsymbol{s}|\boldsymbol{s}', \boldsymbol{a}) \mathrm{d}\boldsymbol{a} \mathrm{d}\boldsymbol{s}'.$$

Note that the above reward function is an unbiased estimator of $r_t(\boldsymbol{s}, \boldsymbol{a})$ for all $t = 1, \ldots, T$ (Yu et al., 2009):

$$\mathbb{E}_{p_t(\boldsymbol{s},\boldsymbol{a})}[\hat{r}_t(\boldsymbol{s}, \boldsymbol{a})|\mathcal{A}] = r_t(\boldsymbol{s}, \boldsymbol{a}), \forall \boldsymbol{s} \in S, \boldsymbol{a} \in A.$$

In the previous equation, $\mathbb{E}_{p(\boldsymbol{s}_t,\boldsymbol{a}_t)}[\cdot|\mathcal{A}]$ denotes the expectation over the joint state-action distribution $p_t(\boldsymbol{s}, \boldsymbol{a})$ by the policies picked by algorithm $\mathcal{A}$ at time step t, where $p_t(\boldsymbol{s}, \boldsymbol{a}) = d_{\mathcal{A},t}(\boldsymbol{s})\pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t)$. By the definition $\rho_t(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{s} \sim d_{\boldsymbol{\theta}}(\boldsymbol{s}), \boldsymbol{a} \sim \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})}[r_t(\boldsymbol{s}, \boldsymbol{a})]$, the estimated expected average reward function satisfies

$$\mathbb{E}_{p_t(\boldsymbol{s},\boldsymbol{a})}\left[\hat{\rho}_t(\boldsymbol{\theta})|\mathcal{A}\right] = \rho_t(\boldsymbol{\theta}),$$

where

$$\hat{\rho}_t(\boldsymbol{\theta}) = \int_{\boldsymbol{s} \in S} d_{\boldsymbol{\theta}}(\boldsymbol{s}) \int_{\boldsymbol{a} \in A} \hat{r}_t(\boldsymbol{s}, \boldsymbol{a})\pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}) \mathrm{d}\boldsymbol{a} \mathrm{d}\boldsymbol{s}.$$

The gradient of $\hat{\rho}_t(\boldsymbol{\theta})$ with respect to the parameter $\boldsymbol{\theta}$ can be obtained by passing the derivative through the integral as

$$\begin{aligned}\mathbb{E}_{p_t(\boldsymbol{s},\boldsymbol{a})}\left[\frac{\partial \hat{\rho}_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}|\mathcal{A}\right] &= \int_{\boldsymbol{s} \in S} \int_{\boldsymbol{a} \in A} d_{\mathcal{A},t}(\boldsymbol{s})\pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t) \frac{\partial \hat{\rho}_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \mathrm{d}\boldsymbol{a} \mathrm{d}\boldsymbol{s} \\ &= \int_{\boldsymbol{s} \in S} \int_{\boldsymbol{a} \in A} \left(\frac{\partial \log d_{\theta}(\boldsymbol{s})}{\partial \boldsymbol{\theta}} + \frac{\partial \log \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right) \\ &\quad \times d_{\theta}(\boldsymbol{s})\pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}) r_t(\boldsymbol{s}, \boldsymbol{a}) \mathrm{d}\boldsymbol{a} \mathrm{d}\boldsymbol{s} \\ &= \frac{\partial \rho_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}.\end{aligned}$$

As the previous equation shows, we replaced the gradient of the expected average reward function $\frac{\partial \rho_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$ in equation 3.2 with its unbiased estimator $\frac{\partial \hat{\rho}_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$.

As will be proved in section 5, the regret bound of the OPG method with bandit feedback is still $O(\sqrt{T})$, although the bound is looser than that in the full-feedback case. If it is not possible to calculate the state distribution directly, its estimate obtained by reinforcement learning may be employed in practice (Ng, Parr, & Koller, 1999).

4 Regret Analysis with Full Feedback

In this section, we provide a regret bound for the OPG algorithm in the full-feedback case.

4.1 Assumptions.

First, we introduce the assumptions required in the proofs. Some assumptions have already been used in related works for discrete state and action MDPs, and we extend them to continuous state and action MDPs.

Assumption 1. There exists a positive number τ, such that for two arbitrary distributions d and d' over S and for every policy parameter $\boldsymbol{\theta} \in \Theta$,

$$\int_{\boldsymbol{s}\in S}\int_{\boldsymbol{s}'\in S} |d(\boldsymbol{s}) - d'(\boldsymbol{s})| p(\boldsymbol{s}'|\boldsymbol{s};\boldsymbol{\theta}) \mathrm{d}\boldsymbol{s}' \mathrm{d}\boldsymbol{s} \leq e^{-1/\tau} \int_{\boldsymbol{s}\in S} |d(\boldsymbol{s}) - d'(\boldsymbol{s})| \mathrm{d}\boldsymbol{s},$$

where

$$p(\boldsymbol{s}'|\boldsymbol{s};\boldsymbol{\theta}) = \int_{\boldsymbol{a}\in A} \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}) p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a}) \mathrm{d}\boldsymbol{a}.$$

τ is called the mixing time (Even-Dar et al., 2004, 2009).

Assumption 2. There exists a positive constant C_1 depending on the specific policy model π such that for two arbitrary policy parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ and for every $\boldsymbol{s} \in S$,

$$\int_{\boldsymbol{a}\in A} |\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}) - \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}')| \mathrm{d}\boldsymbol{a} \leq C_1 \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_1,$$

where $\|\cdot\|_1$ denotes the L_1 norm.

The gaussian policy is a common choice in continuous state and action MDPs. Below, we consider the gaussian policy with mean $\mu(\boldsymbol{s}) = \boldsymbol{\theta}^\top \phi(\boldsymbol{s})$ and standard deviation σ, where $\boldsymbol{\theta}$ is the policy parameter and $\phi(\boldsymbol{s}) : S \to \mathbb{R}^N$ is the basis function. The KL divergence between these two policies is given by

$$
\begin{aligned}
&D(p(\cdot|s;\boldsymbol{\theta})||p(\cdot|s;\boldsymbol{\theta'}))\\
&\quad = \int_{a\in A} \mathcal{N}_{\theta,\sigma}(a)\left\{\log \mathcal{N}_{\theta,\sigma}(a) - \log \mathcal{N}_{\theta',\sigma}(a)\right\} \mathrm{d}a\\
&\quad = \int_{a\in A} \mathcal{N}_{\theta,\sigma}(a)\left\{\frac{1}{2\sigma^2}\left(-(a-\boldsymbol{\theta}^\top \phi(\boldsymbol{s}))^2 + (a-\boldsymbol{\theta}'^\top \phi(\boldsymbol{s}))^2\right)\right\} \mathrm{d}a\\
&\quad \le \frac{\|\phi(\boldsymbol{s})\|_\infty^2}{2\sigma^2}\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_1^2.
\end{aligned}
$$

By Pinsker's inequality, the following inequality holds:

$$
\|p(\cdot|s,\boldsymbol{\theta}) - p(\cdot|s,\boldsymbol{\theta}')\|_1 \le \frac{\|\phi(\boldsymbol{s})\|_\infty}{\sigma}\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_1. \tag{4.1}
$$

This implies that the gaussian policy model satisfies assumption 2 with $C_1 = \frac{\Phi}{\sigma}$, where $\|\phi(\boldsymbol{s})\|_\infty \le \Phi, \forall \boldsymbol{s} \in S$. Note that we do not specify any policy model in the analysis, and therefore the following theoretical analysis is valid for other stochastic policy models as long as the assumptions are satisfied.

Assumption 3. All the reward functions in online MDPs are bounded. For simplicity, we assume that the reward functions satisfy

$$
r_t(\boldsymbol{s},\boldsymbol{a}) \in [0,1], \forall \boldsymbol{s}\in S, \forall \boldsymbol{a} \in A, \forall t = 1,\ldots,T.
$$

Assumption 4. For all $t = 1,\ldots,T$, the second derivative of the expected average reward function satisfies

$$
\nabla_\theta^2 \rho_t(\boldsymbol{\theta}) \le 0, \tag{4.2}
$$

where $\boldsymbol{\theta} \in \Theta$ and Θ is the parameter set, which is convex and compact.

Assumption 4 means that the expected average reward function is concave, which is currently our sufficient condition to guarantee the $O(\sqrt{T})$-regret bound for the OPG algorithm. This assumption can be relaxed to locally concave expected average reward functions, where all the results still hold locally. More specifically the standard policy gradient algorithm (Sutton & Barto, 1998; Peters & Schaal, 2006) has been shown to converge to a local optimal solution, and we use a local optimal policy as the baseline in the definition of the regret instead of the global optimal solution.

4.2 Regret Bound with Concavity. We have the following theorem.

Theorem 1. *The regret against the best offline policy of the OPG algorithm is bounded as*

$$L_{\mathcal{A}}(T) \leq \sqrt{T}\frac{F^2}{2} + \sqrt{T}C_2N + 2\sqrt{T}\tau^2 C_1 C_2 N + 4\tau,$$

where F is the diameter of Θ and $C_2 = \frac{2C_1 - C_1 e^{-1/\tau}}{1-e^{-1/\tau}}$.

Note that the constant C_1 depends on the specific policy model involved, which is claimed in assumption 2.

To prove theorem 1, we decompose the regret in the same way as previous work has (Even-Dar et al., 2004, 2009; Neu, György, & Szepesvári, 2010; Neu, György, Szepesvári, et al., 2010):

$$\begin{aligned} L_{\mathcal{A}}(T) &= R_{\boldsymbol{\theta}^*}(T) - R_{\mathcal{A}}(T) \\ &\leq \left(R_{\boldsymbol{\theta}^*}(T) - \sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}^*)\right) + \left(\sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}^*) - \sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}_t)\right) \\ &\quad + \left(\sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}_t) - R_{\mathcal{A}}(T)\right). \end{aligned} \tag{4.3}$$

In the OPG method, $\rho_t(\boldsymbol{\theta})$ is used for optimization, and the sum of the expected average reward functions $\sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}^*)$ is calculated based on the stationary state distribution $d_{\boldsymbol{\theta}^*}(\boldsymbol{s})$ of the policy parameterized by $\boldsymbol{\theta}^*$. However, the sum of the expected rewards $R_{\boldsymbol{\theta}^*}(T)$is calculated by $d_{\boldsymbol{\theta},t}(\boldsymbol{s})$, the state distribution at time step t following policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}^*)$. A similar argument can be obtained for $\sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}_t)$ and $R_{\mathcal{A}}(T)$. These differences affect the first and third terms of the decomposed regret equation 4.3.

Below, we bound each of the three terms in lemmas 1, 2, and 3, which are proved in appendixes A, B, and C, respectively.

Lemma 1. *The difference between the return and the expected average reward function of the best offline policy parameter satisfies*

$$\left| R_{\boldsymbol{\theta}^*}(T) - \sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}^*) \right| \leq 2\tau.$$

The first term has already been analyzed for discrete state and action online MDPs in Even-Dar et al. (2004, 2009), Neu et al. (2014), and Dick et al. (2014), and we extended it to continuous state and action spaces in lemma 1.

Lemma 2. *The expected average reward function satisfies*

$$\left|\sum_{t=1}^{T}(\rho_t(\boldsymbol{\theta}^*) - \rho_t(\boldsymbol{\theta}_t))\right| \leq \sqrt{T}\frac{F^2}{2} + \sqrt{T}C_2N.$$

Lemma 2 is obtained by using the result of Zinkevich (2003).

Lemma 3. *The difference between the return and the expected average reward function of $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}_t), \forall t = 1, \ldots, T$ given by the OPG algorithm $\mathcal{A}$ satisfies*

$$\left|R_{\mathcal{A}}(T) - \sum_{t=1}^{T}\rho_t(\boldsymbol{\theta}_t)\right| \leq 2\tau^2C_1C_2N\sqrt{T} + 2\tau.$$

Lemma 3 is similar to lemma 5.2 in Even-Dar et al. (2009), but our bound does not depend on the cardinality of state and action spaces.

Combining lemmas 1 to 3, we can immediately obtain theorem 1.

4.3 Regret Analysis under Strong Concavity.

Next we derive a sharper regret bound for the OPG algorithm under a strong concavity assumption.

Theorem 1 shows the theoretical guarantee of the OPG algorithm with the concave assumption. If the expected reward function is strongly concave,

$$\nabla_\theta^2\rho_t \leq -HI_N, \tag{4.4}$$

where H is a positive constant and I_N is the $N \times N$ identity matrix, we have following theorem:

Theorem 2. *The regret against the best offline policy of the OPG algorithm is bounded as*

$$L_{\mathcal{A}}(T) \leq \frac{C_2^2N^2}{2H}(1 + \log T) + \frac{2\tau^2C_1C_2N}{H}\log T + 4\tau,$$

with step size $\eta_t = \frac{1}{Ht}$.

In theorem 2, $C_2 = \frac{2C_1 - C_1e^{-1/\tau}}{1-e^{-1/\tau}}$, where C_1 depends on the specific policy model. We again consider the same decomposition as equation 4.3, and the first term of the regret bound is exactly the same as lemma 1.

The second term is bounded by the following proposition given the strong concavity assumption, equation 4.4, and step size $\eta_t = \frac{1}{Ht}$:

Proposition 1.

$$\sum_{t=1}^{T}(\rho_t(\boldsymbol{\theta}^*) - \rho_t(\boldsymbol{\theta}_t)) \leq \frac{C_2^2 N^2}{2H}(1 + \log T).$$

The proof of proposition 1 is given in appendix D, which follows the same line as Hazan, Agarwal, and Kale (2007).

From the proof of lemma 3, the bound of the third term with the strong concavity assumption, equation 4.4, is given by proposition 2:

Proposition 2.

$$\sum_{t=1}^{T}\rho_t(\boldsymbol{\theta}_t) - R_{\mathcal{A}}(T) \leq \frac{2\tau^2 C_1 C_2 N}{H}\log T + 2\tau. \tag{4.5}$$

The result of proposition 2 is obtained by following the same line as the proof of lemma 3 with a different step size. Combining lemma 1 and propositions 1 and 2, we can obtain theorem 2.

5 Regret Analysis with Bandit Feedback

In this section, we prove a regret bound for the OPG algorithm in the bandit-feedback case.

5.1 Regret Bound with Concavity in the Bandit Scenario. Suppose that there exist $\xi > 0$ and $\epsilon > 0$ such that the policy and the state distribution satisfy

$$\begin{aligned}\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}_t) &\geq \xi, \forall \boldsymbol{s} \in S, \forall \boldsymbol{a} \in A, \forall t = 1, \ldots, T,\\ d_{\mathcal{A},t}(\boldsymbol{s}) &\geq \epsilon, \forall \boldsymbol{s} \in S, \forall t = 1, \ldots, T.\end{aligned}$$

Note that the above assumptions yield the state and action spaces to be compact, where the gaussian policy cannot be used directly.

Then we have the following theorem:

Theorem 3. *The regret of the OPG algorithm with bandit feedback is*

$$\begin{aligned}L_{\mathcal{A}}(T) &= R_{\boldsymbol{\theta}^*}(T) - R_{\mathcal{A}}(T)\\ &\leq 4\tau + \frac{F^2}{2}\sqrt{T} + (C_3 + C_4)N\sqrt{T}\\ &\quad + 2\tau^2(C_1C_3N + C_1C_4N)\sqrt{T},\end{aligned}$$

where $C_3 = \frac{C_1}{\epsilon(1-e^{-1/\tau})}$, $C_4 = \frac{C_1}{\xi\epsilon}$, *and* C_1 *depends on the specific policy model as assumption 2.*

Theorem 3 can be proved by extending the proof of theorem 1 as follows.

The same regret decomposition as equation 4.3 is still possible in the bandit-feedback setting. The first term can be bounded in the same way as the full-information case; lemma 1 still holds. However, the bounds for the second and third terms, originally given in lemmas 2 and 3, should be modified as follows:

Lemma 4. *The expected average reward function given by the online policy gradient algorithm with bandit feedback satisfies*

$$\left|\sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}^*) - \rho_t(\boldsymbol{\theta}_t)\right| \leq \frac{F^2}{2}\sqrt{T} + (C_3 + C_4)N\sqrt{T}.$$

The bound of the second part is still $O(\sqrt{T})$, but it is looser than the bound in the full-information scenario, which is caused by the estimated gradient of the expected average reward function.

Lemma 5. *The third term of the regret of the online policy gradient algorithm with bandit feedback is bounded as*

$$\left|R_{\mathcal{A}}(T) - \sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}_t)\right| \leq 2\tau^2(C_1C_3N + C_1C_4N)\sqrt{T} + 2\tau.$$

Proofs of lemmas 4 and 5 are given in appendix G. From these lemmas, we can immediately obtain theorem 3.

6 Experiments

In this section, we illustrate the behavior of the OPG algorithm through experiments.

6.1 Target Tracking.

The task is to let an agent track an abruptly moving target located in one-dimensional real space $S = \mathbb{R}$. The action space is also one-dimensional real space $A = \mathbb{R}$, and we can change the position of the agent as $s' = s + a$. The reward function is given by evaluating the distance between the agent and target as

$$r_t(s, a) = e^{-\frac{1}{2}(s-\mathrm{tar}(t))^2 - \frac{1}{2}a^2}, \tag{6.1}$$

where $\mathrm{tar}(t) \in [-3, 3]$ denotes the position of the target at time step t. The mechanism for moving the target is set as the uniform distribution over the interval $[-3, 3]$.

We use the gaussian policy with mean parameter $\mu = \theta \cdot s$ and standard deviation parameter $\sigma = 3$ in this experiment. From the standard argument (Peters & Schaal, 2006), the stationary state distribution is the gaussian distribution with zero mean parameter and standard deviation parameter $\tilde{\sigma} = \frac{\sigma}{\sqrt{-\theta^2-2\theta}}, \theta \in (-2, 0)$.[2] Then for all $t = 1, \ldots, T$, the expected average reward functions are given by

$$\begin{aligned}\rho_t(\theta) &= \int_{s\in S} \mathcal{N}_{0,\tilde{\sigma}}(s) \int_{a\in A} \mathcal{N}_{\mu,\sigma}(a) e^{-\frac{1}{2}(s-\mathrm{tar}(t))^2-\frac{1}{2}a^2} \mathrm{d}a\mathrm{d}s \\ &= \frac{1}{\varpi} \exp\left(-\frac{\mathrm{tar}(t)^2(\varpi^2-\tilde{\sigma}^2-\sigma^2\tilde{\sigma}^2)}{2\varpi^2}\right),\end{aligned}$$

where $\varpi = \sqrt{1+\sigma^2+\tilde{\sigma}^2+\sigma^2\tilde{\sigma}^2+\tilde{\sigma}^2\theta^2}$. This implies that $\rho_t(\theta)$ is concave with respect to the parameter θ, and thus $\rho_t(\theta)$ satisfies assumption 3 for all $t = 1, \ldots, T$.[3]

As a baseline method for comparison, we consider the MDP-E algorithm (Even-Dar et al., 2004, 2009), where the exponential weighted average algorithm is used as the best expert. Since MDP-E can handle only discrete states and actions, we discretize the state and action spaces. More specifically, the state space is discretized as

$$(-\infty, -6], (-6, -6+c], (-6+c, -6+2c], \ldots, (6, +\infty),$$

and the action space is discretized as

$$-6, -6+c, -6+2c, \ldots, 6.$$

We consider the following five setups for c:

$$c = 6, 2, 1, 0.5, 0.1.$$

In the experiment, the state distribution and the gradient are estimated by the policy gradient estimator REINFORCE introduced in Peters and

[2] Note that the parameter space is not closed in this experiment. When θ takes a value less than -1.99 or more than -0.01 during gradient update iterations, we project it back to -1.99 or -0.01, respectively.

[3] The analysis of concavity is presented in appendix I.

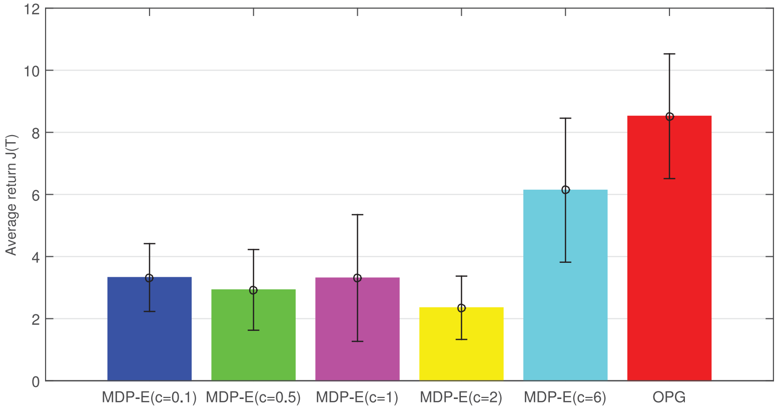


Figure 1: Average and standard deviation of returns of the OPG algorithm and the MDP-E algorithm with different discretization resolution c.

Schaal (2006). $I = 20$ independent experiments are run with $T = 100$ time steps, and the average return $J(T)$ is used for evaluating the performance:

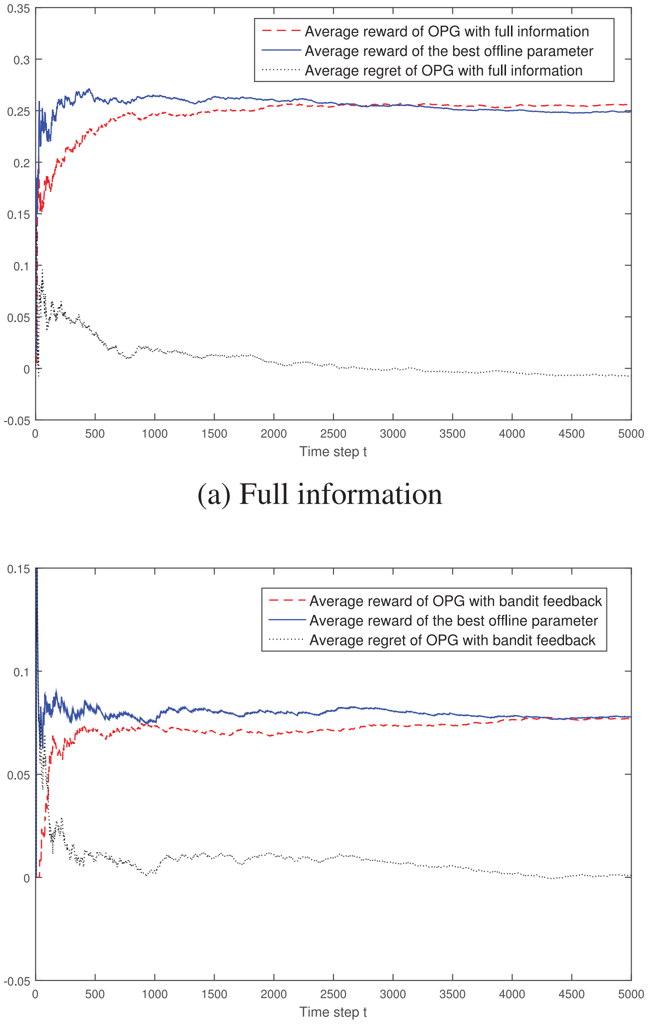
$$J(T) = \frac{1}{I}\sum_{i=1}^{I}\left[\sum_{t=1}^{T} r_t(s_t, a_t)\right].$$

The results are plotted in Figure 1, showing that the OPG algorithm works better than the MDP-E algorithm with the best discretization resolution. This illustrates the advantage of directly handling continuous state and action spaces without discretization. The MDP-E algorithm performs poorly when the discretization resolution is too small. The regret caused by the MDP-E algorithm increases as the cardinalities of state and action spaces increase. On the other hand, the performance of the MDP-E algorithm is limited when the discretization resolution is too large. Moreover, it is difficult to design the best discretization method without knowledge of the target movement.

Figure 2 shows the average rewards and average regrets for full-information and bandit-feedback cases, which substantiate the theoretical results.[4]

6.2 Linear-Quadratic Regulator. The linear-quadratic regulator (LQR) is a simple system, where the transition dynamics is linear and the reward

[4] The state and action spaces are bounded to $[-2, 2]$ in the bandit-feedback experiment.



(a) Full information

(b) Bandit feedback

Figure 2: Average rewards and average regrets of the OPG algorithm with full information and bandit feedback

function is quadratic. This system is instructive because we can compute the best offline parameter and the gradient directly (Peters & Schaal, 2006). Here, an online LQR system is simulated to illustrate the parameter update trajectory of the OPG algorithm.

Let state and action spaces be one-dimensional real space: $S = \mathbb{R}$, $A = \mathbb{R}$. The transitions are deterministically performed as

$$s' = s + a.$$

The reward function is defined as

$$r_t(s, a) = -\frac{1}{2}Q_t s^2 - \frac{1}{2}R_t a^2,$$

where $Q_t \in \mathbb{R}$ and $R_t \in \mathbb{R}$ are chosen from $\{1, \ldots, 10\}$ uniformly at time step $t = 10, 20, 30, \ldots, 10{,}000$.[5] Thus, the reward function is changing abruptly.

We use the gaussian policy with mean parameter $\mu = \theta \cdot s$ and standard deviation parameter $\sigma = 0.1$ and $\sigma = 1$ in full-information and bandit-feedback experiments, respectively. The best offline parameter is given by $\theta^* = -0.92$, and the initial parameter for the OPG algorithm is drawn uniformly at random.

From the standard argument (Peters & Schaal, 2006), the expected average reward function of the above LQR system is given by

$$\rho_t(\theta) = -\frac{1}{2}(R_t + P_t)\sigma^2,$$

where P_t is the positive-definite solution of the modified Ricatti equation $P_t = Q_t + P_t + 2\theta P_t + \theta^2 P_t + \theta^2 R_t$. Then the second-order derivative of $\rho_t(\theta)$ is given by

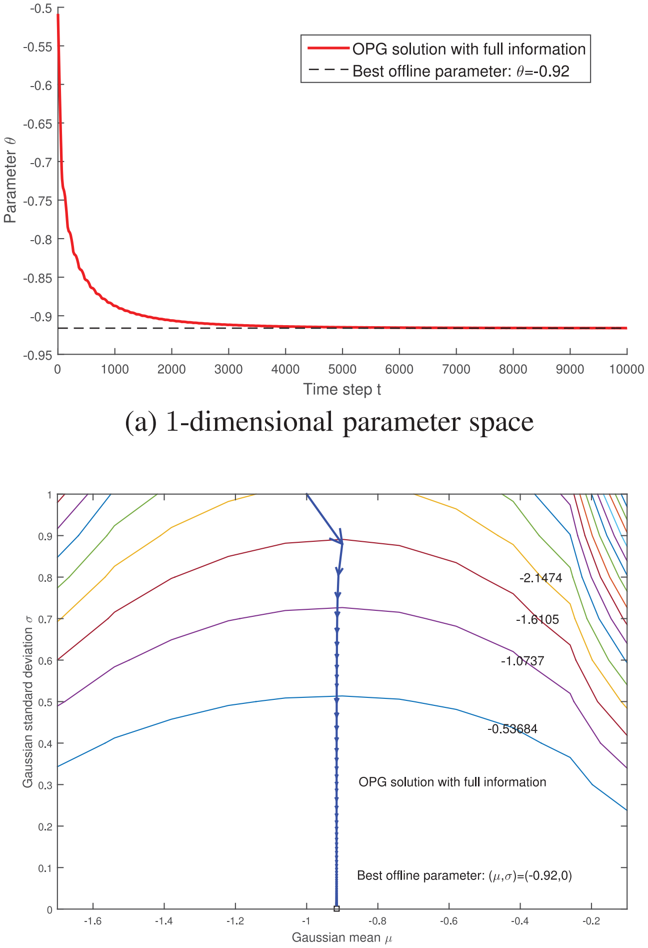
$$\frac{\partial^2 \rho_t(\theta)}{\partial \theta^2} = \frac{\sigma^2 Q_t (6\theta^2 + 12\theta + 8) - 4\sigma^2\theta^3 R_t}{2(2\theta + \theta^2)^3}.$$

Given that P is the positive-definite solution, which yields $-2 < \theta < 0$, we can obtain $\frac{\partial^2 \rho_t(\theta)}{\partial \theta^2} \leq 0$. This means that the expected average reward function of the target LQR system is always concave with respect to the policy parameter.

In Figure 3a, a parameter update trajectory of OPG with full information in the online LQR problem is plotted by the solid line, and the best offline parameter is denoted by the dashed line. This shows that the OPG solution quickly approaches the best offline parameter.

Next, we also include the gaussian standard deviation σ in the policy parameter: $\boldsymbol{\theta} = (\mu, \sigma)^\top$. When σ takes a value less than 0.01 during gradient

[5] The reward function is not bounded, which violates assumption 3. However, it is interesting to illustrate that the parameter updated by the OPG algorithm still converges to the best offline parameter.



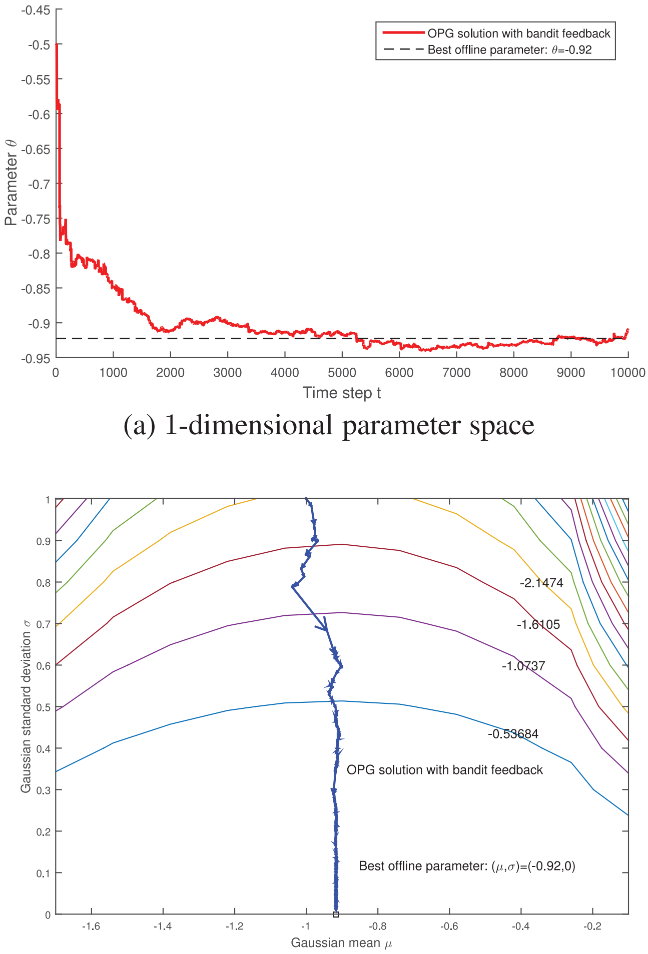
(a) 1-dimensional parameter space

(b) 2-dimensional parameter space

Figure 3: Trajectory of the OPG solution with full information and the best offline parameter.

update iterations, we project it back to 0.01. A parameter update trajectory is plotted in Figure 3b, showing again that the OPG solution smoothly approaches the best offline parameter.

In Figure 4a, the solid line shows the trajectory of the OPG algorithm with bandit feedback in the online LQR system simulation. The result validates that the OPG solution converges to the best offline parameter with a slightly slower speed compared with the full-information result.



(a) 1-dimensional parameter space

(b) 2-dimensional parameter space

Figure 4: Trajectory of the OPG solution with bandit feedback and the best offline parameter.

The parameter trajectory is shown in Figure 4b, when the standard deviation σ is included in the parameter. The OPG solution still approaches the best offline mean parameter as we expect.

7 Conclusion

In this letter, we proposed an online policy gradient method for continuous state and action online MDPs and showed that the regret of the proposed method is $O(\sqrt{T})$ under a certain concavity assumption on the expected average reward function. A notable fact is that the regret bound does not depend on the cardinality of state and action spaces, which makes the proposed algorithm suitable in handling continuous states and actions. We further extended our method to the bandit-feedback scenario and showed that the regret of the extended method is still $O(\sqrt{T})$. Furthermore, we also established the $O(\log T)$ regret bound under a strong concavity assumption for the full information setup. Through experiments, we illustrated that directly handling continuous state and action spaces by the proposed method is more advantageous than discretizing them and applying an existing method.

Our future work will extend the current theoretical analysis to nonconcave expected average reward functions, where gradient-based algorithms suffer from the local optimal problem. A difficulty in this situation it that the regret bound with bandit feedback becomes trivial when the lower bounds of policy and state distributions are too small. Thus, improving our current result in the bandit feedback scenario is an important future work. Another important challenge is to develop an effective method to estimate the stationary state distribution, which is required in our algorithm.

Appendix A: Proof of Lemma 1

The following proposition holds, which can be obtained by recursively using assumption 1:

Proposition 3. *For any policy parameter $\boldsymbol{\theta}$, the state distribution $d_{\boldsymbol{\theta},t}$ at time t and stationary state distribution $d_{\boldsymbol{\theta}}$ satisfy*

$$\int_{s\in S} |d_{\theta,t}(\boldsymbol{s}) - d_{\theta}(\boldsymbol{s})| ds \leq 2e^{-t/\tau}.$$

The first part of the regret bound in theorem 1 is caused by the difference between the state distribution at time t and the stationary state distribution following the best offline policy parameter $\boldsymbol{\theta}^*$,

$$\begin{aligned}\left|R_{\boldsymbol{\theta}^*}(T) - \sum_{t=1}^{T} \rho_t(\boldsymbol{\theta}^*)\right| &= \left|\sum_{t=1}^{T}\left[\int_{s\in S} d_{\theta^*,t}(\boldsymbol{s}) \int_{\boldsymbol{a}\in A} r_t(\boldsymbol{s},\boldsymbol{a})\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}^*)\mathrm{d}\boldsymbol{s}\mathrm{d}\boldsymbol{a}\right.\right.\\ &\qquad \left.\left. - \int_{\boldsymbol{s}\in S} d_{\theta^*}(\boldsymbol{s}) \int_{\boldsymbol{a}\in A} r_t(\boldsymbol{s},\boldsymbol{a})\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}^*)\mathrm{d}\boldsymbol{s}\mathrm{d}\boldsymbol{a}\right]\right|\\ &\leq \sum_{t=1}^{T}\int_{s\in S}\left|d_{\theta^*,t}(\boldsymbol{s}) - d_{\theta^*}(\boldsymbol{s})\right| \mathrm{d}\boldsymbol{s}\end{aligned}$$

$$\leq 2\sum_{t=1}^{T} e^{-t/\tau}$$
$$\leq 2\tau,$$

where the second inequality can be obtained by assumption 1.

Appendix B: Proof of Lemma 2

The following proposition is a continuous extension of lemma 6.3 in Even-Dar et al. (2009):

Proposition 4. *For two policies with different parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$, an arbitrary distribution d over S, and the constant $C_1 > 0$ given in assumption 2, it holds that*

$$\int_{\boldsymbol{s}\in S} d(\boldsymbol{s}) \int_{\boldsymbol{s}'\in S} |p(\boldsymbol{s}'|\boldsymbol{s};\boldsymbol{\theta}) - p(\boldsymbol{s}'|\boldsymbol{s};\boldsymbol{\theta}')| d\boldsymbol{s}' d\boldsymbol{s} \leq C_1\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_1,$$

where

$$p(\boldsymbol{s}'|\boldsymbol{s};\boldsymbol{\theta}) = \int_{\boldsymbol{a}\in A} \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}) p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a}) d\boldsymbol{a}.$$

Then we have the following proposition, which is proved in appendix E:

Proposition 5. *For all $t = 1, \ldots, T$, the expected average reward function $\rho_t(\boldsymbol{\theta})$ for two different parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ satisfies*

$$|\rho_t(\boldsymbol{\theta}) - \rho_t(\boldsymbol{\theta}')| \leq C_2\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_1.$$

From proposition 5, we have the following proposition:

Proposition 6. *Let*

$$\boldsymbol{\theta} = [\theta^{(1)}, \ldots, \theta^{(i)}, \ldots, \theta^{(N)}],$$
$$\boldsymbol{\theta}' = [\theta^{(1)}, \ldots, \theta^{(i)'}, \ldots, \theta^{(N)}],$$

and suppose that the expected average reward $\rho_t(\boldsymbol{\theta})$ for all $t = 1, \ldots, T$ is Lipschitz continuous with respect to each dimension $\theta^{(i)}$. Then we have

$$|\rho_t(\boldsymbol{\theta}) - \rho_t(\boldsymbol{\theta}')| \leq C_2|\theta^{(i)} - \theta^{(i)'}|, \forall i = 1, \ldots, N.$$

From proposition 6, we have the following proposition:

Proposition 7. *For all $t = 1, \ldots, T$, the partial derivative of expected average reward function $\rho_t(\boldsymbol{\theta})$ with respect to $\theta^{(i)}$ is bounded as*

$$\left|\frac{\partial \rho_t(\boldsymbol{\theta})}{\partial \theta^{(i)}}\right| \leq C_2, \forall i = 1, \ldots, N,$$

and $\|\nabla_{\boldsymbol{\theta}} \rho_t(\boldsymbol{\theta})\|_1 \leq NC_2$.

From proposition 7, the result of online convex optimization (Zinkevich, 2003) is applicable to the current setup. More specifically we have

$$\sum_{t=1}^{T} \left(\rho_t(\boldsymbol{\theta}^*) - \rho_t(\boldsymbol{\theta}_t)\right) \leq \frac{F^2}{2}\sqrt{T} + C_2 N\sqrt{T},$$

which concludes the proof.

Appendix C: Proof of Lemma 3

The following proposition holds, which can be obtained from assumption 2 and

$$\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t+1}\|_1 \leq \eta_t \|\nabla_{\boldsymbol{\theta}} \rho_t(\boldsymbol{\theta}_t)\|_1 \leq C_2 N \eta_t.$$

Proposition 8. *Consecutive policy parameters $\boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_{t+1}$ given by the OPG algorithm satisfy*

$$\int_{\boldsymbol{a}\in A} |\pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t) - \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_{t+1})| d\boldsymbol{a} \leq C_1 C_2 N \eta_t.$$

From propositions 4 and 8, we have the following proposition:

Proposition 9. *For consecutive policy parameters $\boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_{t+1}$ given by the OPG algorithm and arbitrary transition probability density $p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$, it holds that*

$$\int_{\boldsymbol{s}\in S} d(\boldsymbol{s}) \int_{\boldsymbol{s}'\in S} \int_{\boldsymbol{a}\in A} p(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a}) \\ \times |\pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t) - \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_{t+1})| d\boldsymbol{a} d\boldsymbol{s}' d\boldsymbol{s} \leq C_1 C_2 N \eta_t.$$

Then the following proposition holds, which is proved in appendix F following the same line as lemma 5.1 in Even-Dar et al. (2009):

Proposition 10. *The state distribution $d_{\mathcal{A},t}$ given by algorithm $\mathcal{A}$ and the stationary state distribution $d_{\boldsymbol{\theta}_t}$ of policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}_t)$ satisfy*

$$\int_{s\in S} |d_{\boldsymbol{\theta}_t}(\boldsymbol{s}) - d_{\mathcal{A},t}(\boldsymbol{s})| d\boldsymbol{s} \leq 2\tau^2\eta_{t-1}C_1C_2N + 2e^{-t/\tau}.$$

Although the original bound given in Even-Dar et al. (2004, 2009) depends on the cardinality of the action space, that is not the case in the current setup.

Then the third term of the decomposed regret, equation 4.3, is expressed as

$$\begin{aligned}\left|R_{\mathcal{A}}(T) - \sum_{t=1}^{T}\rho_t(\boldsymbol{\theta}_t)\right| &= \left|\sum_{t=1}^{T}\int_{s\in S} d_{\mathcal{A},t}(\boldsymbol{s})\int_{a\in A} r_t(\boldsymbol{s},\boldsymbol{a})\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}_t)\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s}\right.\\ &\qquad \left. - \sum_{t=1}^{T}\int_{s\in S} d_{\boldsymbol{\theta}_t}(\boldsymbol{s})\int_{a\in A} r_t(\boldsymbol{s},\boldsymbol{a})\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}_t)\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s}\right|\\ &\leq \sum_{t=1}^{T}\int_{s\in S}|d_{\mathcal{A},t}(\boldsymbol{s}) - d_{\pi_t}(\boldsymbol{s})|\mathrm{d}\boldsymbol{s}\\ &\leq 2\tau^2C_1C_2N\sum_{t=1}^{T}\eta_t + 2\sum_{t=1}^{T}e^{-t/\tau}\\ &\leq 2\tau^2C_1C_2N\sqrt{T} + 2\tau,\end{aligned}$$

which concludes the proof.

Appendix D: Proof of Proposition 1

The proof of proposition 1 can be obtained from Hazan et al. (2007), that is, by the Taylor approximation, the expected average reward function can be decomposed as

$$\begin{aligned}&\rho_t(\boldsymbol{\theta}^*) - \rho_t(\boldsymbol{\theta}_t)\\ &\quad = \nabla_\theta\rho_t(\boldsymbol{\theta}_t)^\top(\boldsymbol{\theta}^* - \boldsymbol{\theta}_t) + \frac{1}{2}(\boldsymbol{\theta}^* - \boldsymbol{\theta}_t)^\top\nabla_\theta^2\rho_t(\boldsymbol{\xi}_t)(\boldsymbol{\theta}^* - \boldsymbol{\theta}_t)\\ &\quad \leq \nabla_\theta\rho_t(\boldsymbol{\theta}_t)^\top(\boldsymbol{\theta}^* - \boldsymbol{\theta}_t) - \frac{H}{2}\|\boldsymbol{\theta}^* - \boldsymbol{\theta}_t\|^2, \qquad \text{(D.1)}\end{aligned}$$

where $\boldsymbol{\xi}_t$ is some point between $\boldsymbol{\theta}^*$ and $\boldsymbol{\theta}_t$. The last inequality comes from the strong concavity assumption, equation 4.4. Given the parameter updating rule,

$$\nabla_\theta \rho_t(\boldsymbol{\theta}^* - \boldsymbol{\theta}_t) = \frac{1}{2\eta_t}((\boldsymbol{\theta}^* - \boldsymbol{\theta}_t)^2 - (\boldsymbol{\theta}^* - \boldsymbol{\theta}_{t+1})^2) + \eta_t \|\nabla_\theta \rho_t(\boldsymbol{\theta}_t)\|^2,$$

summing up all T terms of equation D.1, and setting $\eta_t = \frac{1}{Ht}$ yield

$$\begin{aligned}
&\sum_{t=1}^{T}(\rho_t(\boldsymbol{\theta}^*) - \rho_t(\boldsymbol{\theta}_t)) \\
&\quad \leq \sum_{t=1}^{T}\left(\frac{1}{\eta_{t+1}} - \frac{1}{\eta_t} - H\right)\|\boldsymbol{\theta}^* - \boldsymbol{\theta}_t\|^2 + \|\nabla_t \rho_t(\boldsymbol{\theta}_t)\|^2 \sum_{t=1}^{T}\eta_t \\
&\quad \leq \frac{C_2^2 N^2}{2H}(1 + \log T).
\end{aligned}$$

Appendix E: Proof of Proposition 5

For two different parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$, we have

$$\begin{aligned}
|\rho_t(\boldsymbol{\theta}) - \rho_t(\boldsymbol{\theta}')| &= \bigg|\int_{s\in S} d_{\boldsymbol{\theta}}(\boldsymbol{s})\int_{a\in A}\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})r_t(\boldsymbol{s},\boldsymbol{a})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s} \\
&\quad - \int_{s\in S} d_{\boldsymbol{\theta}'}(\boldsymbol{s})\int_{a\in A}\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}')r_t(\boldsymbol{s},\boldsymbol{a})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s}\bigg| \\
&\leq \int_{s\in S} |d_{\boldsymbol{\theta}}(\boldsymbol{s}) - d_{\boldsymbol{\theta}'}(\boldsymbol{s})| \int_{a\in A}\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta})r_t(\boldsymbol{s},\boldsymbol{a})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s} \\
&\quad + \int_{s\in S} d_{\boldsymbol{\theta}'}(\boldsymbol{s})\int_{a\in A}\left|\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}) - \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}')\right| r_t(\boldsymbol{s},\boldsymbol{a})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s}.
\end{aligned} \tag{E.1}$$

The first equation comes from equation 3.1, and the second inequality is obtained from the triangle inequality. Since assumptions 2 and 3 imply

$$\int_{s\in S} d_{\boldsymbol{\theta}'}(\boldsymbol{s})\int_{a\in A}|\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}) - \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}')|r_t(\boldsymbol{s},\boldsymbol{a})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{s} \leq C_1\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_1,$$

and also

$$\int_{\boldsymbol{a}\in A} \pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\theta}) r_t(\boldsymbol{s},\boldsymbol{a})\mathrm{d}\boldsymbol{a} \leq 1,$$

equation E.1 can be written as

$$\begin{aligned}
|\rho_t(\boldsymbol{\theta}) - \rho_t(\boldsymbol{\theta}')| &\leq \int_{\boldsymbol{s}\in S} |d_{\boldsymbol{\theta}}(\boldsymbol{s}) - d_{\boldsymbol{\theta}'}(\boldsymbol{s})|\mathrm{d}\boldsymbol{s} + C_1\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_1 \\
&= \int_{\boldsymbol{s}\in S}\int_{\boldsymbol{s}'\in S} |d_{\boldsymbol{\theta}}(\boldsymbol{s}')p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta}) - d_{\boldsymbol{\theta}'}(\boldsymbol{s}')p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta}')|\mathrm{d}\boldsymbol{s}'\mathrm{d}\boldsymbol{s} \\
&\quad + C_1\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_1 \\
&\leq \int_{\boldsymbol{s}\in S}\int_{\boldsymbol{s}'\in S} |d_{\boldsymbol{\theta}}(\boldsymbol{s}')p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta}) - d_{\boldsymbol{\theta}'}(\boldsymbol{s}')p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta})|\mathrm{d}\boldsymbol{s}'\mathrm{d}\boldsymbol{s} \\
&\quad + \int_{\boldsymbol{s}\in S}\int_{\boldsymbol{s}'\in S} d_{\boldsymbol{\theta}'}(\boldsymbol{s}')|p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta}) - p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta}')|\mathrm{d}\boldsymbol{s}'\mathrm{d}\boldsymbol{s} \\
&\quad + C_1\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_1 \\
&\leq e^{-1/\tau}\int_{\boldsymbol{s}\in S} |d_{\boldsymbol{\theta}}(\boldsymbol{s}) - d_{\boldsymbol{\theta}'}(\boldsymbol{s})|\mathrm{d}\boldsymbol{s} + 2C_1\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_1.
\end{aligned}$$

The second equality comes from the definition of the stationary state distribution, and the third inequality can be obtained from the triangle inequality. The last inequality follows from assumption 1 and proposition 4. Thus, we have

$$|\rho_t(\boldsymbol{\theta}) - \rho_t(\boldsymbol{\theta}')| \leq \frac{2C_1 - C_1 e^{-1/\tau}}{1-e^{-1/\tau}}\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_1,$$

which concludes the proof.

Appendix F: Proof of Proposition 10

This proof is following the same line as lemma 5.1 in Even-Dar et al. (2009):

$$\begin{aligned}
&\int_{\boldsymbol{s}\in S} |d_{\mathcal{A},k}(\boldsymbol{s}) - d_{\boldsymbol{\theta}_t}(\boldsymbol{s})|\mathrm{d}\boldsymbol{s} \\
&\quad = \int_{\boldsymbol{s}\in S}\int_{\boldsymbol{s}'\in S} \left|d_{\mathcal{A},k-1}(\boldsymbol{s}')p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta}_k) - d_{\boldsymbol{\theta}_t}(\boldsymbol{s}')p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta}_t)\right|\mathrm{d}\boldsymbol{s}'\mathrm{d}\boldsymbol{s} \\
&\quad \leq \int_{\boldsymbol{s}\in S}\int_{\boldsymbol{s}'\in S} \left|d_{\mathcal{A},k-1}(\boldsymbol{s}')p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta}_t) - d_{\boldsymbol{\theta}_t}(\boldsymbol{s}')p(\boldsymbol{s}|\boldsymbol{s}';\boldsymbol{\theta}_t)\right|\mathrm{d}\boldsymbol{s}'\mathrm{d}\boldsymbol{s}
\end{aligned}$$

$$+\int_{s\in S}\int_{s'\in S}\left|d_{\mathcal{A},k-1}(s')p(s|s';\boldsymbol{\theta}_k)-d_{\mathcal{A},k-1}(s')p(s|s';\boldsymbol{\theta}_t)\right|\mathrm{d}s'\mathrm{d}s$$

$$\leq e^{-1/\tau}\int_{s\in S}\left|d_{\mathcal{A},k-1}(s)-d_{\boldsymbol{\theta}_t}(s)\right|\mathrm{d}s+2(t-k)C_1C_2N\eta_{t-1}. \tag{F.1}$$

The first equation comes from the definition of the stationary state distribution, and the second inequality can be obtained by the triangle inequality. The third inequality holds from assumption 1 and

$$\begin{aligned}&\int_{s\in S}\int_{s'\in S}\left|d_{\mathcal{A},k-1}(s')p(s|s';\boldsymbol{\theta}_k)-d_{\mathcal{A},k-1}(s')p(s|s';\boldsymbol{\theta}_t)\right|\mathrm{d}s\\&\quad\leq C_1\|\boldsymbol{\theta}_t-\boldsymbol{\theta}_k\|_1\\&\quad\leq C_1\sum_{i=k}^{t-1}\eta_i\|\nabla_{\boldsymbol{\theta}}\rho_i(\boldsymbol{\theta}_i)\|_1\\&\quad\leq 2(t-k)C_1C_2N\eta_{t-1}.\end{aligned}$$

Recursively using equation F.1, we have

$$\begin{aligned}\int_{s\in S}|d_{\mathcal{A},t}(s)-d_{\pi_t}(s)|\mathrm{d}s&\leq 2\sum_{k=2}^{t}e^{-(t-k)/\tau}(t-k)C_1C_2N\eta_{t-1}+2e^{-t/\tau}\\&\leq 2\tau^2C_1C_2N\eta_{t-1}+2e^{-t/\tau},\end{aligned}$$

which concludes the proof.

Appendix G: Proofs of Lemmas 4 and 5

As we show in section 5, an unbiased estimator of reward function is used for updating the parameter $\boldsymbol{\theta}$; we also show that the corresponding estimated gradient is unbiased, which can be bounded by the following lemma, which is proved in appendix H.

Lemma 6. *The estimated gradient* $\nabla_{\boldsymbol{\theta}}\hat{\rho}_t(\boldsymbol{\theta})$ *satisfies*

$$\|\nabla_{\boldsymbol{\theta}}\hat{\rho}_t(\boldsymbol{\theta})\|_1\leq C_3N+C_4N.$$

Following the same line with the proof of lemma 3.1 in Flaxman, Kalai, and McMahan (2005), we first define the auxiliary functions for all $\boldsymbol{x}\in\Theta$ as

$$\varrho_t(\boldsymbol{x})=\rho_t(\boldsymbol{x})+\boldsymbol{x}^\top\kappa_t,$$

where $\kappa_t = \nabla_{\boldsymbol{\theta}} \hat{\rho}_t(\boldsymbol{\theta}_t) - \nabla_{\boldsymbol{\theta}} \rho_t(\boldsymbol{\theta}_t)$. It is observed that

$$\nabla_{\boldsymbol{x}} \varrho_t(\boldsymbol{\theta}_t) = \nabla_{\boldsymbol{\theta}} \hat{\rho}_t(\boldsymbol{\theta}_t),$$

and the unbiased estimation satisfies

$$\mathbb{E}_{p_t(\boldsymbol{s},\boldsymbol{a})}[\varrho_t(\boldsymbol{\theta}_t)|\mathcal{A}] = \rho_t(\boldsymbol{\theta}_t),$$

where the above equation follows from the fact that $\mathbb{E}_{p_t(\boldsymbol{s},\boldsymbol{a})}[\kappa_t|\mathcal{A}] = 0$, and $\mathbb{E}_{p_t(\boldsymbol{s},\boldsymbol{a})}[\boldsymbol{\theta}_t \kappa_t|\mathcal{A}] = 0$. Thus, we can obtain

$$\sum_{t=1}^{T} \left(\rho_t(\boldsymbol{\theta}^*) - \rho_t(\boldsymbol{\theta}_t) \right) \leq \frac{F^2}{2}\sqrt{T} + (C_3 + C_4)N\sqrt{T},$$

which concludes the proof of lemma 4 by using the result of lemma 6. Similarly, using lemma 6 in the proof of lemma 3, we obtain lemma 5.

Appendix H: Proof of Lemma 6

The estimated gradient is expressed as

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}} \hat{\rho}_t(\boldsymbol{\theta}_t) = & \int_{s \in S} \int_{\boldsymbol{a} \in A} d_{\boldsymbol{\theta}_t}(\boldsymbol{s}) \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t) \hat{r}_t(\boldsymbol{s}, \boldsymbol{a}) \\
& \times (\nabla_{\boldsymbol{\theta}} \ln d_{\boldsymbol{\theta}_t}(\boldsymbol{s}) + \nabla_{\boldsymbol{\theta}} \ln \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t)) d\boldsymbol{s} d\boldsymbol{a} \\
= & \frac{\nabla_{\boldsymbol{\theta}} d_{\boldsymbol{\theta}_t}(\boldsymbol{s}_t)}{d_{\mathcal{A},t}(\boldsymbol{s}_t)} r_t(\boldsymbol{s}_t, \boldsymbol{a}_t) \\
& + \frac{d_{\boldsymbol{\theta}_t}(\boldsymbol{s}_t)}{d_{\mathcal{A},t}(\boldsymbol{s}_t)} \ln \nabla_{\boldsymbol{\theta}} \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t) r_t(\boldsymbol{s}_t, \boldsymbol{a}_t).
\end{aligned}$$

Consider the stationary distribution as a function of parameter $\boldsymbol{\theta}$ for all $\boldsymbol{s} \in S$, Then, from proposition 5, the bound for the gradient of the stationary distribution is given by

$$|\nabla_{\boldsymbol{\theta}} d_{\boldsymbol{\theta}_t}(\boldsymbol{s})| \leq \frac{C_1 N}{1 - e^{-1/\tau}}, \forall \boldsymbol{s} \in S, \forall t = 1, \ldots, T.$$

Similarly, from assumption 2, the bound for the gradient of policy π is given by

$$|\nabla_{\boldsymbol{\theta}} \ln \pi(\boldsymbol{a}|\boldsymbol{s}; \boldsymbol{\theta}_t)| \leq \frac{C_1 N}{\xi}, \forall \boldsymbol{s} \in S, \forall \boldsymbol{a} \in A, \forall t = 1, \ldots, T.$$

Then we have

$$\|\nabla_{\boldsymbol{\theta}} \hat{\rho}_t(\boldsymbol{\theta}_t)\|_1 \leq \frac{C_1 N}{\epsilon(1 - e^{-1/\tau})} + \frac{C_1 N}{\epsilon \xi}, \forall t = 1, \ldots, T.$$

Appendix I: Concavity Analysis for Target Tracking

The reward function in the target tracking experiment is defined as

$$r_t(s, a) = e^{-\frac{1}{2}(s - \mathrm{tar}(t))^2 - \frac{1}{2}a^2}, \forall t = 1, \ldots, T.$$

Then for all $t = 1, \ldots, T$, the expected average reward function is given by

$$\begin{aligned}\rho_t(\theta) &= \int_{s \in S} \mathcal{N}_{0,\tilde{\sigma}}(s) \int_{a \in A} \mathcal{N}_{\mu,\sigma}(a) e^{-\frac{1}{2}(s - \mathrm{tar}(t))^2 - \frac{1}{2}a^2} \mathrm{d}a\mathrm{d}s \\ &= \frac{1}{\varpi} \exp\left(-\frac{\mathrm{tar}(t)^2(\varpi^2 - \tilde{\sigma}^2 - \sigma^2\tilde{\sigma}^2)}{2\varpi^2}\right),\end{aligned}$$

where $\varpi = \sqrt{1 + \sigma^2 + \tilde{\sigma}^2 + \sigma^2\tilde{\sigma}^2 + \tilde{\sigma}^2\theta^2}$ and $\tilde{\sigma} = \frac{\sigma}{\sqrt{-\theta^2 - 2\theta}}$. For verifying the concavity of $\rho_t(\theta)$, we obtain the derivative of $\rho_t(\theta)$ with respect to θ by plugging in $\sigma = 3$ as

$$\begin{aligned}&\frac{\partial \rho_t(\theta)}{\partial \theta} \\ &\quad = \sqrt{\frac{-\theta^2 - 2\theta}{-\theta^2 - 20\theta + 90}} \exp\left(-\frac{t^2}{2} \cdot \frac{-\theta^2 - 20\theta}{-\theta^2 - 20\theta + 90}\right) \\ &\qquad \times \left[-\mathrm{tar}(t)^2 \frac{-90(\theta + 10)}{(-\theta^2 - 20\theta + 90)^2} - \frac{-9\theta^2 + 90\theta + 90}{(-\theta^2 - 20\theta + 90)(-\theta^2 - 2\theta)}\right].\end{aligned}$$

We observed that $\frac{\partial \rho_t(\theta)}{\partial \theta}$ is monotonically nonincreasing as shown in Figure 5. Thus, the defined expected average reward functions $\rho_t(\theta), \forall t = 1, \ldots, T$ are concave with respect to the parameter θ.

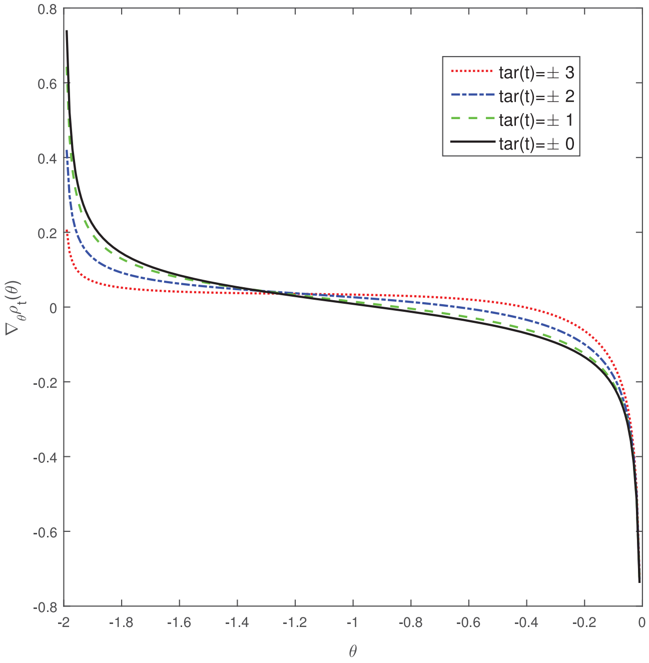


Figure 5: The derivative of $\rho_t(\theta)$ with respect to θ.

Acknowledgments

Y.M. was supported by the MEXT scholarship and the JST CREST program. T.Z. was supported by NSFC 61502339 and SRF for ROCS, SEM. K.H. was supported by MEXT KAKENHI 25330261 and 24106010. M.S. was supported by KAKENHI 23120004.

References

Abbasi-Yadkori, Y., Bartlett, P., Kanade, V., Seldin, Y., & Szepesvari, C. (2013). Online learning in Markov decision processes with adversarially chosen transition probability distributions. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems, 26* (pp. 2508–2516). Red Hook, NY: Curran.

Dick, T., György, A., & Szepesvári, C. (2014). Online learning in Markov decision processes with changing cost sequences. In *Proceedings of the 31st International Conference on Machine Learning* (pp. 512–520). JMLR.

Even-Dar, E., Kakade, S. M., & Mansour, Y. (2004). Experts in a Markov decision process. In L. K. Saul, Y. Weiss, & L. Bottou (Eds.), *Advances in neural information processing system, 17* (pp. 401–408). Cambridge, MA: MIT Press.

Even-Dar, E., Kakade, S. M., & Mansour, Y. (2009). Online Markov decision processes. *Mathematics of Operations Research*, *34*(3), 726–736.

Flaxman, A., Kalai, A., & McMahan, B. (2005). Online convex optimization in the bandit setting: Gradient descent without a gradient. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms* (pp. 385–394). Philadelphia: SIAM.

Hazan, E., Agarwal, A., & Kale, S. (2007). Logarithmic regret algorithms for online convex optimization. *Machine Learning*, *69*(2–3), 169–192.

Kalai, A., & Vempala, S. (2005). Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, *71*(3), 291–307.

Ma, Y., Zhao, T., Hatano, K., & Sugiyama, M. (2014). An online policy gradient algorithm for Markov decision processes with continuous states and actions. In *Proceedings of the Machine Learning and Knowledge Discovery in Databases—European Conference* (pp. 354–369). New York: Springer-Verlag.

Neu, G., György, A., & Szepesvári, C. (2010). The online loop-free stochastic shortest-path problem. In *Proceedings of the 23rd Conference on Learning Theory* (pp. 231–243).

Neu, G., György, A., & Szepesvári, C. (2012). The adversarial stochastic shortest path problem with unknown transition probabilities. In *Proceedings of the International Conference on Artificial Intelligence and Statistics* (pp. 805–813). JMLR.

Neu, G., György, A., Szepesvári, C., & Antos, A. (2010). Online Markov decision processes under bandit feedback. In J. D. Lafferty, C.K.I. Williams, J. Shawe-Taylor, R. S. Zemel, & A. Culotta (Eds.), *Advances in neural information processing systems, 23* (pp. 1804–1812). Red Hook, NY: Curran.

Neu, G., György, A., Szepesvári, C., & Antos, A. (2014). Online Markov decision processes under bandit feedback. *IEEE Transactions on Automatic Control*, *59*(3), 676–691.

Ng, A. Y., Parr, R., & Koller, D. (1999). Policy search via density estimation. In S. A. Solla, T. K. Leen, & K.-R. Müller (Eds.), *Advances in neural information processing systems, 12* (pp. 1022–1028). Cambridge, MA: MIT Press.

Peters, J., & Schaal, S. (2006). Policy gradient methods for robotics. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 2219–2225). Piscataway, NJ: IEEE.

Sehnke, F., Osendorfer, C., Rückstiess T., Graves A., Peters J., & Schmidhuber, J. (2010). Parameter-exploring policy gradients. *Neural Networks*, *23*(4), 551–559.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.

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

Yu, J. Y., Mannor, S., & Shimkin, N. (2009). Markov decision processes with arbitrary reward processes. *Mathematics of Operations Research*, *34*(3), 737–757.

Zimin, A., & Neu, G. (2013). Online learning in episodic Markovian decision processes by relative entropy policy search. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems*, *26* (pp. 1583–1591). Red Hook, NY: Curran.

Zinkevich, M. (2003). Online convex programming and generalized infinitesimal gradient ascent. In T. Fawcett & N. Mishra (Eds.), *Proceedings of the 20th International Conference on Machine Learning ICML* (pp. 928–936). Cambridge, MA: AAAI Press.

Received November 20, 2014; accepted August 11, 2015.