Adaptive Approximate Policy Iteration

Botao Hao Deepmind
Nevena Lazic Deepmind
Yasin Abbasi-Yadkori Deepmind
Pooria Joulani Deepmind
Csaba Szepesvári Deepmind

Abstract

Model-free reinforcement learning algorithms combined with value function approximation have recently achieved impressive performance in a variety of application domains. However, the theoretical understanding of such algorithms is limited, and existing results are largely focused on episodic or discounted Markov decision processes (MDPs). In this work, we present adaptive approximate policy iteration (AAPI), a learning scheme which enjoys a $\widetilde{O}(T^{2/3})$ regret bound for undiscounted, continuing learning in uniformly ergodic MDPs. This is an improvement over the best existing bound of $\widetilde{O}(T^{3/4})$ for the average-reward case with function approximation. Our algorithm and analysis rely on online learning techniques, where value functions are treated as losses. The main technical novelty is the use of a data-dependent adaptive learning rate coupled with a so-called optimistic prediction of upcoming losses. In addition to theoretical guarantees, we demonstrate the advantages of our approach empirically on several environments.

1 INTRODUCTION

Our work focuses on model-free algorithms for learning in *infinite-horizon undiscounted* Markov decision processes (MDPs), also known as average-reward MDPs. Although model-free algorithms have recently achieved impressive advances in multiple applications [Mnih et al., 2015, Van Hasselt et al., 2016], few performance guarantees exist, especially in the average-reward case with function approximation. In this work, we propose *Adaptive Approximate Policy Iteration* (AAPI), a model-free learning scheme that can work with function approximation, and utilizes an adaptive data-dependent learning rate. We analyze the performance of AAPI in infinite-horizon undiscounted MDPs in terms of high-probability regret.

Our approach follows the "online MDP" line of work [Even-Dar et al., 2009, Neu et al., 2014, Abbasi-Yadkori et al., 2019a], where the agent iteratively selects policies by running an online learning algorithm in each state, and the loss fed to each algorithm is the policy Q-function in that state. This results in a variant of approximate policy iteration (API), where the policy improvement step produces a policy optimal in hindsight w.r.t. *the average of all previous* Q-functions rather than just the most recent one. The original work of Even-Dar et al. [2009] studied this scheme with known dynamics, tabular representation, and adversarial reward functions. More recent works [Abbasi-Yadkori et al., 2019a,b] have adapted this approach to the case of unknown dynamics, stochastic rewards, and value function approximation. The averaging of value functions is further justified theoretically and empirically by Vieillard et al. [2019] and Vieillard et al. [2020].

A notable feature of our algorithm is thatwe exploit the fact that losses (Q-function estimates) are slow-changing. In particular, our policy improvement step relies on the adaptive optimistic follow-the-regularized-leader (AO-FTRL) update [Mohri and Yang, 2016]. The resulting policies are Boltzmann distributions over the sum of past estimated Q-functions, coupled with an optimistic prediction of the upcoming loss and a state-dependent adaptive learning rate (softmax temperature). Our policy improvement step can also be seen as regularizing each policy by the KL-divergence to the previous policy; the reduction to online learning offers a principled way to scale such regularization.

On the theoretical side, we prove the first $\widetilde{O}(T^{2/3})$ regret upper bound in the undiscounted, continuing setting with function approximation. This is an improvement over the best existing $\widetilde{O}(T^{3/4})$ bound of Abbasi-Yadkori et al. [2019a] for the same setting, which ignores the slow-changing na-

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

ture of the estimated Q-functions. Our analysis exploits the fact that the change in consecutive Q-function estimates can be bounded by the change in policies. We rely on the results of Rakhlin and Sridharan [2013], but employ a different regret decomposition, with additional information provided by MDP properties. We emphasize that our learning framework is not limited to a particular function approximation method, and that in practice it serves the purpose of appropriately regularizing the policy improvement step of API.

Related work. Most no-regret algorithms for infinite-horizon undiscounted MDPs are model-based, and only applicable to tabular representations [Bartlett, 2009, Jaksch et al., 2010, Ouyang et al., 2017, Fruit et al., 2018, Jian et al., 2019, Talebi and Maillard, 2018]. In the model-free tabular setting, Wei et al. [2019] show optimistic Q-learning achieves $O(\mathrm{sp}(V_*)(XA)^{1/3}T^{2/3})$ regret in weakly-communicating MDPs, where $\mathrm{sp}(V_*)$ is the span of the optimal state-value function, X, A are the size of state and action spaces. In the case of uniformly ergodic MDPs, they show a bound of $O(\sqrt{t_{\mathrm{mix}}^3 \rho AT})$ on the *expected regret*, where t_{mix} is the mixing time and ρ is the stationary distribution mismatch coefficient. In the model-free setting with function approximation, Abbasi-Yadkori et al. [2019a] achieve $O(d^{1/2}T^{3/4})$ regret in ergodic MDPs. Here d is the size of the compressed state-action space (XA for tabular representation, number of features for linear Q-functions).

In episodic MDPs with horizon H, Jin et al. [2018] show an $O(\sqrt{H^3XAT})$ regret bound for Q-learning with tabular representation. With linear function approximation, Yang and Wang [2019b], Jin et al. [2019], Cai et al. [2019] show an $O(\sqrt{d^3H^3T})$ regret bound for an optimistic version of least-squares value/policy iteration under linear MDPs assumption. The RLSVI algorithm [Osband et al., 2016] performs exploration in the value function parameter space, and therefore can be applied with function approximation. Its worse-case regret bound of $O(\sqrt{H^5X^3AT})$ holds in the tabular setting [Russo, 2019] and $O(d^2\sqrt{H^4T})$ holds under the linear MDPs assumption [Zanette et al., 2020].

Another thread of the literature [Ross et al., 2011, Ross and Bagnell, 2014] proposes a reduction of model-free RL to any no-regret online learning. While Ross et al. [2011] mainly focus on imitation learning, Ross and Bagnell [2014] consider the finite-horizon case and uses a generic no-regret online learner that may result in a worse regret guarantee. Very recently, Cheng et al. [2019] exploit optimistic mirror descent to speed up policy optimization in RL, but do not provide a regret analysis in average-reward case.

AAPI is also similar to the conservative policy iteration works, which attempt to stabilize API by regularizing each policy towards the previous policy [Kakade and Langford, 2002, Schulman et al., 2015, 2017, Abdolmaleki et al., 2018, Geist et al., 2019, Vieillard et al., 2020]. In particular, Neu et al. [2017] identify several state-of-the-art entropy-regularized RL algorithms as approximate variants of mirror descent, and Shani et al. [2019] provides convergence rates for a mirror descent like algorithm in the discounted setting. Vieillard et al. [2020] provides a systematical analysis of regularization in RL. To the best of the authors' knowledge, none of these works use adaptive data-dependent learning rate to accelerate policy learning.

2 PROBLEM SETTING

We first introduce some notation. We use $\Delta_{\mathcal{S}}$ to denote the space of probability distributions defined on the set $\mathcal{S}$ and write $[d] = \{1, 2, \ldots, d\}$. For vectors $u, v \in \mathbb{R}^d$, we define the weighted ℓ_2-norm as $\|v\|_u^2 = \sum_{i=1}^d u_i v_i^2$ and ℓ_∞-norm as $\|u\|_\infty = \max_{j\in[d]} u_j$. In general, we treat discrete distributions as row vectors.

Infinite-horizon undiscounted MDPs are often characterized by a finite state space $\mathcal{X}$, a finite action space $\mathcal{A}$, a reward function $r : \mathcal{X} \times \mathcal{A} \to [0, 1]$, and a transition probability function $P : \mathcal{X} \times \mathcal{A} \to \Delta_{\mathcal{X}}$. The agent does not know the transition probability and the reward function in advance. A policy $\pi : \mathcal{X} \to \Delta_{\mathcal{A}}$ is a mapping from a state to a distribution over actions. Let $\{(x_t^\pi, a_t^\pi)\}_{t=1}^\infty$ denote the state-action sequence obtained by following policy π. The expected average reward of policy π is defined as

$$\lambda_\pi := \lim_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^T r(x_t^\pi, a_t^\pi)\right].$$

The agent interacts with the environment as follows: at each round t, the agent observes a state $x_t \in \mathcal{X}$, chooses an action $a_t \sim \pi_t(\cdot|x_t)$, and receives a reward $r(x_t, a_t)$. The environment then transitions to the next state x_{t+1} with probability $\mathbb{P}(x_{t+1}|x_t, a_t)$. The initial state x_1 is randomly generated from some unknown distribution. Let π^* be an unknown fixed policy. The regret of an algorithm with respect to this fixed policy is defined as

$$R_T = \sum_{t=1}^T \Big(\lambda_{\pi^*} - r(x_t, a_t)\Big), \tag{2.1}$$

where $a_t \sim \pi_t(\cdot|x_t)$. The learning goal is to find an algorithm that minimizes the long-term regret R_T. Note that R_T is still a random variable so we will bound it with high probability.

For each policy π, we denote $\mathcal{P}^\pi \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{X}|}$ to be the Markov chain induced by π, where the component $(\mathcal{P}^\pi)_{x,x'}$

is the transition probability from x to x' under π, i.e. $(\mathcal{P}^\pi)_{x,x'} = \sum_{a\in\mathcal{A}} \pi(a|x)P(x'|x,a)$. For a distribution μ over $\mathcal{X}$, we let $\mu\mathcal{P}^\pi$ be the distribution over $\mathcal{X}$ that results from executing the policy π for one step after the initial state is sampled from μ. A stationary distribution μ_π of a policy π over states satisfies $\mu_\pi \mathcal{P}^\pi = \mu_\pi$. For a policy π, its expected reward can be expressed as

$$\lambda_\pi = \mathbb{E}_{x\sim\mu_\pi, a\sim\pi(\cdot|x)}\big[r(x,a)\big].$$

In this work, we focus on ergodic MDPs, a sub-class of weakly communicating MDPs. An MDP is ergodic if the Markov chain induced by any policy π is both irreducible and aperiodic, which means any state is reachable from any other state by following a suitable policy. It is well-known that all ergodic MDPs have an unique stationary state distribution, and so μ_π and λ_π are well-defined. In addition, ergodic MDPs have a finite *mixing time*, defined below.

Definition 2.1. The mixing time of ergodic MDPs is defined as $t_{\text{mix}} :=$

$$\max_\pi \min\left\{t \geq 1, \left\|(\mathcal{P}^\pi)^t(x,\cdot) - \mu_\pi\right\|_1 \leq \frac{1}{4}, \forall x \in \mathcal{X}\right\},$$

that characterizes how fast MDPs reach stationary distributions from any state under any policy.

Finally, we define the value function under policy π as

$$V_\pi(x) = \mathbb{E}^\pi\Big[\sum_{t=1}^{\infty}(r(x_t,a_t) - \lambda_\pi)|x_1 = x\Big],$$

where $\mathbb{E}^\pi$ is with respect to the sample path induced by π. The state-action value function $Q_\pi(x,a)$ and $V_\pi(x)$ can also be defined as the unique solutions to the Bellman equation:

$$\begin{aligned} Q_\pi(x,a) &= r(x,a) - \lambda_\pi + \sum_{x'} P(x'|x,a)V_\pi(x') \\ V_\pi(x) &= \sum_a \pi(a|x)Q_\pi(x,a). \end{aligned} \tag{2.2}$$

3 ALGORITHM

AAPIis a variant of approximate policy iteration and it proceeds in phases. Suppose the total number of rounds is T. We divide T into K phases of length $\tau = T/K$ and assume τ is an integer for simplicity. Within each phase, our algorithm performs two tasks: policy evaluation and policy improvement.

Policy evaluation. In each phase $k \in [K]$, the algorithm executes the current policy π_k for τ time steps, and computes an estimate $\widehat{Q}_{\pi_k}$ of the true action-value function Q_{π_k}. We leave unspecified the value function estimation method $\mathcal{G}$; for example, one can use incremental algorithms, or both on-policy and off-policy data. AAPI is better interpreted as a learning schema.

Policy improvement. For each state $x \in \mathcal{X}$, the policy improvement step takes the form of the adaptive optimistic follow-the-regularized-leader (AO-FTRL) update [Mohri and Yang, 2016]: $\pi_{k+1}(a|x) =$

$$\begin{aligned} \operatorname*{argmax}_{f\in\mathcal{F}} \Big\langle f, \sum_{s=1}^{k} \widehat{Q}_{\pi_s}(x,\cdot) + M_{k+1}(x,\cdot)\Big\rangle \\ - \eta_k(x)\mathcal{R}(f). \end{aligned} \tag{3.1}$$

(See Step 3 in Section 5 for a generic description of AO-FTRL.) The terms in Eq. (3.1) are as follows:

- The estimates $\widehat{Q}_{\pi_s}(x,\cdot) \in \mathbb{R}^{|\mathcal{A}|}$ are the loss functions fed to the AO-FTRL algorithm. $\mathcal{R}(f)$ is the negative entropy regularizer, and $\mathcal{F}$ is the probability simplex.

- The side-information $M_{k+1}(x,\cdot) \in \mathbb{R}^{|\mathcal{A}|}$ is a vector computable based on past information and being predictive of the next loss $\widehat{Q}_{\pi_{k+1}}(x,\cdot)$. Since the policies are expected to change slowly due to the nature of exponential-weight-average type algorithms, we set $M_{k+1}(x,\cdot) = \widehat{Q}_{\pi_k}(x,\cdot)$ (better guesses such as off-policy estimates can be used if available).

- The choice of learning rate $\eta_k(x)$ is crucial both theoretically and empirically. In particular, we choose $\eta_k(x)$ in a data-dependent fashion as $\eta_k(x) =$

$$\eta\sqrt{2\sum_{s=1}^{k}\|\widehat{Q}_{\pi_s}(x,\cdot) - M_s(x,\cdot)\|_\infty^2}\,. \tag{3.2}$$

  A notable feature of $\eta_k(x)$ is that it is also state-dependent.

Based on (3.1), the next policy is a Boltzmann distribution (a consequence of negative entropy regularizer) over the sum of all past state-action value estimates and the side-information: $\pi_{k+1}(a|x) \propto$

$$\exp\Big(\eta_k^{-1}(x)\big(\sum_{s=1}^{k}\widehat{Q}_{\pi_s}(x,a) + M_{k+1}(x,a)\big)\Big). \tag{3.3}$$

The overall algorithm is summarized in Algorithm 1.

Algorithm 1 Adaptive approximate policy iteration (AAPI)

Input: phase length τ, number of phases K, initial state x_0, parameter η, value function estimation algorithm $\mathcal{G}$.
Initialize: $\pi_1(a|x) = 1/|\mathcal{A}|, \forall x, a$;
Repeat:
for $k = 1, \ldots, K$ **do**
  Execute π_k for τ time steps and collect dataset $\mathcal{D}_k$.
  Estimate $\widehat{Q}_{\pi_k}$ from $\mathcal{D}_1, \ldots, \mathcal{D}_k$ using $\mathcal{G}$.
  Calculate adaptive learning rate $\eta_k(x)$ as in (3.2):

$$\eta_k(x) = \eta \sqrt{2 \sum_{s=1}^{k} \|\widehat{Q}_{\pi_s}(x, \cdot) - M_s(x, \cdot)\|_\infty^2},$$

  with where $M_s = \widehat{Q}_{\pi_{s-1}}$.
  Let $q(x, a) = \sum_{s=1}^{k} \widehat{Q}_{\pi_s}(x, a) + M_{k+1}(x, a)$.
  Update next policy as:

$$\pi_{k+1}(a|x) \propto \exp\left(\eta_k(x)^{-1} q(x, a)\right),$$

  where $\eta_k(x)$ is defined in Eq. (3.2).
end for
Output: π_{K+1}

4 ANALYSIS

To derive a regret bound for Algorithm 1, we decompose the cumulative regret (2.1) as follows:

$$R_T = \sum_{t=1}^{T} \left(\lambda_{\pi_t} - r(x_t, a_t)\right) + \sum_{t=1}^{T} \left(\lambda_{\pi^*} - \lambda_{\pi_t}\right). \quad (4.1)$$

The first term captures the sum of differences between observed rewards and their long term averages. If policies are changing slowly, or if they are kept fixed for extended periods of time, we expect this term to capture the noise in the regret. The second term is called *pseudo-regret* in literature. It measures the difference between the expected reward of a fixed policy and the policies produced by the algorithm.

We first impose a condition on the quality of policy evaluation at each phase. For a probability distribution μ on $\mathcal{X}$ and a stochastic policy π, define $\mu \otimes \pi$ to be the distribution on $\mathcal{X} \times \mathcal{A}$ that puts the probability mass $\mu(x)\pi(a|x)$ on pair $(x, a) \in \mathcal{X} \times \mathcal{A}$. Recall that μ_{π^*} is the stationary distribution of π^* over the states.

Condition 4.1 . For each phase $k \in [K]$, denote $D_{\pi_k} = \widehat{Q}_{\pi_k} - Q_{\pi_k}$. We assume the following holds with probability $1 - \delta$,

$$\max\left\{\|D_{\pi_k}\|_{\mu_{\pi^*} \otimes \pi^*}, \|D_{\pi_k}\|_{\mu_{\pi^*} \otimes \pi_k}, \|D_{\pi_k}\|_\infty\right\} \leq \varepsilon_0 + \widetilde{C}\sqrt{\frac{\log(1/\delta)}{\tau}}, \quad (4.2)$$

where ε_0 is the irreducible approximation error and $\widetilde{C}$ is a problem dependent constant. Additionally, there exists a constant b such that $\widehat{Q}_{\pi_k}(x, a) \in [b, b + Q_{\max}]$ for any pair $(x, a) \in \mathcal{X} \times \mathcal{A}$ and $k \in [K]$.

Remark 4.2. The problem dependent constant $\widetilde{C}$ will in general depend on $d, t_{\text{mix}}, \mu_{\pi^*}, \mu_{\pi_k}$. Here, d is the dimension of the representation (e.g. $|\mathcal{X}||\mathcal{A}|$ for the tabular case, or number of features for the linear value function case).

Remark 4.3. The requirement for the $\mu_{\pi^*} \otimes \pi^*$-norm and $\mu_{\pi^*} \otimes \pi_k$-norm has been shown to hold, for example, with linear value function approximation using the LSPE algorithm [Bertsekas and Ioffe, 1996], under Assumptions B.1-B.3 given in the Appendix. Lemma B.4 in the Appendix shows that the requirement for ℓ_∞-norm can also be satisfied, for example, with linear value functions, under similar conditions.

Remark 4.4. The estimation error generally depends on the mismatch between distributions μ_{π_k} and μ_{π^*}. With value functions linear in features $\phi(x, a) \in \mathbb{R}^d$, this mismatch depends on the spectra of matrices $\mathbb{E}_\nu[\phi(x, a)\phi(x, a)^\top]$ for different distributions ν, and need not scale in the number of state-action pairs. See Assumption A4 in Abbasi-Yadkori et al. [2019a] for a more detailed explanation.

Theorem 4.5 (Main result)**.** Consider an ergodic MDP and suppose Condition 4.1 holds. By choosing the phase length $\tau = (\widetilde{C}/\rho t_{\text{mix}}^3)^{2/3} T^{2/3}$, we have with probability at least $1 - 1/T$,

$$R_T = \widetilde{O}\left(t_{\text{mix}}^2 (\rho \widetilde{C}^2)^{1/3} T^{2/3} + T\varepsilon_0\right),$$

where ρ is the distribution mismatch coefficient that has used in previous work [Kakade and Langford, 2002, Agarwal et al., 2020, Wei et al., 2019] and $\widetilde{\mathcal{O}}(\cdot)$ hides universal constants and poly-logarithmic factors.

Remark 4.6. It is worth comparing the above result with the regret bound presented in Abbasi-Yadkori et al. [2019a]. Ignoring the irreducible error ε_0, we improve the leading order of their general result (Corollary 4.6 in Abbasi-Yadkori et al. [2019a]) from $\widetilde{O}(T^{3/4})$ to $\widetilde{O}(T^{2/3})$. When specialized to linear value function approximation where $\widetilde{C}$ scales with $d^{1/2}$ (Theorem 5 in Abbasi-Yadkori et al. [2019a]), we improve their results from $\widetilde{O}(d^{1/2}T^{3/4})$ to $\widetilde{O}(d^{1/3}T^{2/3})$.

Remark 4.7. It is worth to mention that Wei et al. [2019] obtains $\widetilde{O}(\sqrt{T})$ regret in terms of *expected regret* in the

tabular case for ergodic MDPs while we consider *high-probability regret*. In particular, their analysis does not account for the estimation and approximation errors in Q-functions that will significantly complicate the analysis and result in a worse regret bound.

5 PROOF SKETCH

In this section, we provide a proof sketch for Theorem 4.5. Technical details are deferred to Appendix A. At a high level, we bound the two terms in the regret decomposition Eq. (4.1) separately. While the first term is bounded by the fast mixing condition, the second term is split into the regret due to value function estimation error and the regret due to online learning reduction.

Step 1: fast mixing. To bound the first term in Eq. (4.1), we require the following uniform fast mixing condition, which is used frequently in online MDP literature [Even-Dar et al., 2009, Neu et al., 2014]. Note that ergodic MDPs that this paper focuses on automatically satisfy this condition.

Condition 5.1 (Uniform fast mixing). There exists a number $t_{\text{mix}} > 0$ such that for any policy π and any pair of distributions μ and μ' over $\mathcal{X}$, it holds that

$$\big\|(\mu-\mu')\mathcal{P}^{\pi}\big\|_1 \le \exp(-1/t_{\text{mix}})\big\|\mu-\mu'\big\|_1. \tag{5.1}$$

The following lemma provides upper bounds for the first term (see e.g. Lemma 4.4 in Abbasi-Yadkori et al. [2019a] for a proof).

Lemma 5.2. Suppose that Condition 5.1 holds. The following inequality holds with probability at least $1-\delta$,

$$\begin{aligned}&\Big|\sum_{t=1}^{T}\big(\lambda_{\pi_t}-r(x_t,a_t)\big)\Big|\\ &\le Kt_{\text{mix}}+4\sqrt{2}t_{\text{mix}}\sqrt{KT\log(T/\delta)},\end{aligned}$$

where K is the number of phases.

Step 2: decomposition. We bound the second term (pseudo regret) in Eq. (4.1). Since the policy is only updated at the end of each phase of length τ (see line 9 in Algorithm 1), we have $\pi_t = \pi_k$ for $t \in \{\tau(k-1),\dots,\tau k\}$. Thus, the pseudo-regret term can be rewritten as

$$\sum_{t=1}^{T}\Big(\lambda_{\pi^*}-\lambda_{\pi_t}\Big)=\tau\sum_{k=1}^{K}\Big(\lambda_{\pi^*}-\lambda_{\pi_k}\Big). \tag{5.2}$$

We slightly abuse the notation by writing $Q_\pi(x,\pi') = \sum_a \pi'(a|x)Q_\pi(x,a)$. In particular, $Q_\pi(x,\pi)$ is exactly the value function $V_\pi(x)$ by Definition 2.2. Applying the performance difference lemma (Lemma C.1 in the supplementary material), we have

$$\lambda_{\pi^*}-\lambda_{\pi_k}=\Big\langle\mu_{\pi^*},Q_{\pi_k}(\cdot,\pi_*)-Q_{\pi_k}(\cdot,\pi_k)\Big\rangle.$$

Bridging by empirical estimations, we decompose (5.2) into $R_{1T}+R_{2T}$, where

$$\begin{aligned}R_{1T}&=\tau\sum_{k=1}^{K}\Big\langle\mu_{\pi^*},Q_{\pi_k}(\cdot,\pi_*)-\widehat{Q}_{\pi_k}(\cdot,\pi_*)\Big\rangle\\&\quad+\tau\sum_{k=1}^{K}\Big\langle\mu_{\pi^*},\widehat{Q}_{\pi_k}(\cdot,\pi_k)-Q_{\pi_k}(\cdot,\pi_k)\Big\rangle,\\R_{2T}&=\tau\sum_{k=1}^{K}\Big\langle\mu_{\pi^*},\widehat{Q}_{\pi_k}(\cdot,\pi^*)-\widehat{Q}_{\pi_k}(\cdot,\pi_k)\Big\rangle.\end{aligned} \tag{5.3}$$

Step 3: estimation error. The term R_{1T} quantifies the regret incurred in the policy evaluation step due to the estimation error and function approximation error of Q-function in each phase. It can be bounded as in Theorem 4.1 of Abbasi-Yadkori et al. [2019a] under similar assumptions, which we reproduce here for completeness.

Lemma 5.3. Suppose Condition 4.1 holds. Then

$$R_{1T}\le T\Big(\varepsilon_0+\widetilde{C}\sqrt{\frac{\log(1/\delta)}{\tau}}\Big), \tag{5.4}$$

with probability at least $1-\delta$.

Step 4: online learning reduction. Minimizing R_{2T} can be cast into an online learning problem [Cesa-Bianchi and Lugosi, 2006, Shalev-Shwartz et al., 2012], and this observation determines the choice of our algorithm. Previous work has tackled this subproblem using mirror descent, resulting in $\widetilde{O}(T^{3/4})$ regret after optimizing τ ignoring the irreducible error ε_0. Here we instead use the AO-FTRL framework, which allows us to show an improved $\widetilde{O}(T^{2/3})$ regret bound. As we show, the reason we can benefit from optimism is that the losses (Q-functions) change slowly, and we carefully transfer this knowledge to the adaptive learning rate. This is the main technical contribution of the paper.

First, we state the framework of AO-FTRL and its regret results. Let $\{q_t\}_{t=1}^T$ be a sequence of loss vectors and let $\{f_t\}_{t=1}^T \subseteq \mathcal{F}$ be a sequence of prediction vectors, where $\mathcal{F}$ is the probability simplex. At the beginning of each round, the algorithm receives a side-information vector M_t. In literature, $\{M_s\}_{s=1}^t$ are also called predictable sequences [Rakhlin and Sridharan, 2012], and the algorithm can be seen as a way of utilizing prior knowledge about loss sequences. The algorithm then selects an action f_t, and suffers a cost $\langle f_t, q_t\rangle$. The goal of this online learning problem is

to minimize the cumulative regret with respect to the best action in hindsight f^*, defined as $\widetilde{R}_T = \sum_{t=1}^T \langle f_t - f^*, q_t \rangle$.

Let $\mathcal{R} : \mathcal{F} \to \mathbb{R}$ be a 1-strongly convex regularizer on $\mathcal{F}$ with respect to some norm $\|\cdot\|$ and denote by $\|\cdot\|_*$ its dual norm. Initialize $f_1 = \operatorname{argmin}_{f\in\mathcal{F}} \mathcal{R}(f)$. At each round t, AO-FTRL has the following form:

$$f_{t+1} = \operatorname*{argmin}_{f\in\mathcal{F}} \langle f, \sum_{s=1}^t q_s + M_{t+1}\rangle + \eta_t \mathcal{R}(f),$$
$$\eta_t = \eta \sqrt{\sum_{s=1}^t \|q_s - M_s\|_*^2},$$

where η is an absolute constant. It's easy to see that η_t is non-decreasing. For simplicity, we assume $M_1 = 0, \eta_0 = 0$. Next lemma provides a generic regret bound for AO-FTRL. The detailed proof is deferred to Appendix A.2 in the supplementary material.

Lemma 5.4. Choose $\eta = \sqrt{2/\mathcal{R}(f^*)}$ and denote $R_{\max} = \max_f \mathcal{R}(f)$. The cumulative regret for AO-FTRL is upper-bounded by

$$\widetilde{R}_T \le \sqrt{2R_{\max} \sum_{t=1}^T \|q_t - M_t\|_*^2} - \sum_{t=1}^T \frac{\eta_t}{4} \|f_t - f_{t+1}\|^2 + \langle M_{T+1}, f^* - f_{T+1}\rangle. \tag{5.5}$$

Remark 5.5. Unlike the AO-FTRL analyses of Rakhlin and Sridharan [2012], Mohri and Yang [2016], but similarly to, e.g., the analysis of Joulani et al. [2017], Eq. (5.5) has a key negative term (at the expense of a slightly larger constant factor in the main positive term). These negative terms, which are retained from a tight regret bound on the forward regret of AO-FTRL [Joulani et al., 2017], track the evolution of the policy f_t. With the proper choice of M_t, the norm terms $\|q_t - M_t\|_*$ will also be controlled by the evolution of f_t (see Lemma 5.6), and the aforementioned negative terms allow us to greatly reduce the contribution of the norm terms $\|q_t - M_t\|_*$ to the overall regret.

The reason that minimizing R_{2T} can be cast into an online learning problem is as follows. By the definition of $Q_\pi(x, \pi')$ in Step 2, we rewrite R_{2T} in (5.3) as

$$R_{2T} = \tau \sum_{x\in\mathcal{X}} \mu_{\pi^*}(x) \sum_{k=1}^K \left\langle \pi^*(\cdot|x) - \pi_k(\cdot|x), \widehat{Q}_{\pi_k}(x, \cdot)\right\rangle.$$

For each state $x \in \mathcal{X}$, we view $\pi_k(\cdot|x)$ as the prediction vector and $\widehat{Q}_{\pi_k}(x, \cdot)$ as the loss vector. The equivalence between R_{2T} and $\widetilde{R}_T$ enables us to utilize the generic regret bound for AO-FTRL in Lemma 5.4 for each individual state.

Next, we will show that under some conditions, the change in the true Q values can be bounded by the change of policies. This is a unique property of ergodic MDPs that allows us to benefit from the negative term in (5.5). To ensure Q_π is unique, we assume $\sum_x \mu_\pi(x) V_\pi(x) = 0$.

Lemma 5.6 (Relative Q-function Error)**.** For any two successive policies π_{k-1} and π_k, the following holds for any state-action pair (x, a),

$$\left|Q_{\pi_k}(x,a) - Q_{\pi_{k-1}}(x,a)\right| \le t_{\text{mix}}^2 \log_2^2(K) \max_x \left\|\pi_{k-1}(\cdot|x) - \pi_k(\cdot|x)\right\|_1 + \frac{2}{K^3}.$$

The detailed proof of Lemma 5.6 is deferred to Appendix A.4. Combining the result in Lemmas 5.4 and 5.6, we can derive the following lemma.

Lemma 5.7. Suppose Condition 4.1 holds. Then the following upper bound holds with probability at least $1 - \delta$,

$$R_{2T} \lesssim \tau t_{\text{mix}}^4 \rho \log_2^4(K) + T\Big(\frac{\widetilde{C}^2 \log(1/\delta)}{\tau} + \varepsilon_0^2\Big), \tag{5.6}$$

where $\lesssim$ hides universal constant factors.

The detailed proof of Lemma 5.7 is deferred to Appendix A.3 in the supplementary material. Finally, we optimize τ to be $(\widetilde{C}/\rho t_{\text{mix}}^3)^{2/3} T^{2/3}$ and reach our conclusion.

Remark 5.8. Within the upper bound (5.6), $\widetilde{C}^2 \log(1/\delta)/\tau + \varepsilon_0^2$ stands for the approximation error and estimation error per round. When value functions can be computed exactly (known MDP) and for phase length $\tau = 1$, the online learning reduction regret for AAPI scales logarithmically in the number of phases K, while POLITEX [Abbasi-Yadkori et al., 2019a] scales as $\sqrt{K}$. This is the main reason that we can improve the regret from $\widetilde{O}(T^{3/4})$ to $\widetilde{O}(T^{2/3})$.

6 EXPERIMENTS

In this section we provide an empirical evaluation of AAPIon several environments. We compare AAPIto POLITEX, which corresponds to updating policies using a mirror descent rule rather than AO-FTRL. We also evaluate RLSVI ([Osband et al., 2016], Algorithms 1 and 2 with $\sigma^2 = 1$ and tuned λ), where policies are greedy w.r.t. a randomized estimate of Q_*. Overall, we find that AAPI performs well in discrete-state environments such as DeepSea [Osband et al., 2017], whereas the adaptive per-state learning rate is less helpful in environments such as CartPole [Barto et al., 1983] with continuous states and smooth dynamics.

We approximate all value functions using least-squares Monte Carlo, i.e. linear regression from state-action fea-

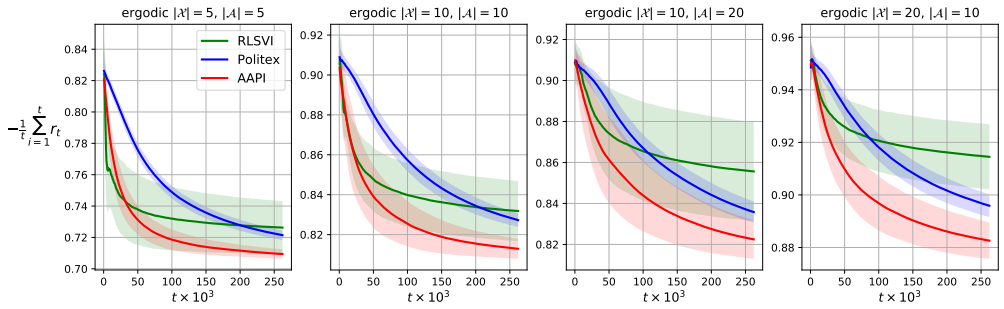

Figure 1: Evaluation on a tabular ergodic MDP.

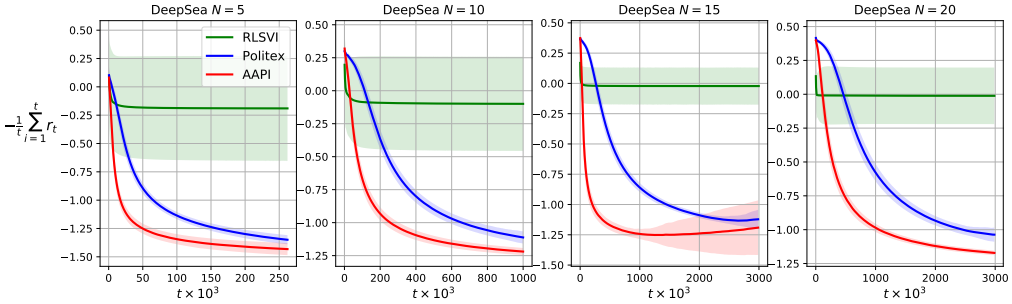

Figure 2: Evaluation on DeepSea environments of different sizes.

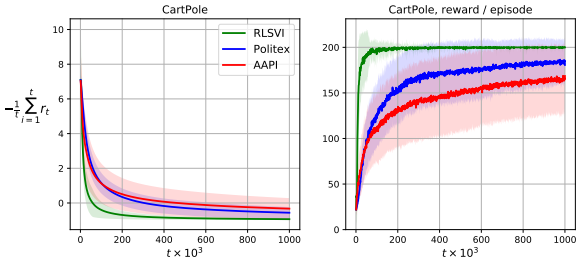

Figure 3: Evaluation on the CartPole environment.

tures to empirical returns. For MDPs with a large or continuous state space $\mathcal{X}$, updating per-state learning rates can be impractical. Instead, we store the weights of past Q-functions in memory, and for each state in the trajectory, we compute the learning rate using a subset of $n_k \leq 30$ randomly-selected past weight vectors (we correct the scale of the estimate by multiplying with $\sqrt{k/n_k}$. With rich function approximation such that neural networks, one can keep a fixed buffer with a subset of the previous Q-functions (chosen in a randomized way, or keeping the most recent K networks as in Abbasi-Yadkori et al. [2019a]), or train distillation networks that summarize the sum of previous Q-functions. Another possibility is to parameterize π_k and optimize the objective w.r.t. the parameters. For Boltzmann policies, we tune the constant η for the learning rate $\eta_k(x)$ in the range $[0.01, 100]$. For each environment and algorithm we evaluate $-\sum_{s=1}^{t} r_t/t$ and plot the mean and standard deviation over 50 runs. The environments we evaluate on are as follows.

Tabular ergodic MDPs. We consider a simple tabular MDP where $r(1, a) = 1$, $r(x, a) = 0$ for $x \neq 1$. On any action in state 1, the environment transitions to a randomly chosen state $x \neq 1$. On action 1 in a state $x \neq 1$, the environment transitions to state $x - 1$ with probability 0.9, and to a randomly chosen state with probability 0.1. On all other actions in $x \neq 1$, the environment transitions to a randomly chosen state. We represent state-action pairs using one-hot indicator vectors of size $|\mathcal{X}||\mathcal{A}|$, and experiment with different sizes of the state and action spaces $\mathcal{X}$ and $\mathcal{A}$.

DeepSea [Osband et al., 2017]. In the DeepSea environment, states comprise an $N \times N$ grid, and there are two actions. The environment transitions and costs are deterministic. The agent starts in the top-left cell $(0, 0)$. On action 0, the agent transitions down and left, and receives reward 0. On action 1, the agent transitions down and right, and receives reward -1. On transitioning to the bottom-right cell $(N - 1, N - 1)$, the agent receives reward $2N$. The infinite-horizon version of the environment wraps the environment around the vertical axis. An optimal strategy first takes the action 1 N times (to get to $(N - 1, N - 1)$) and then takes

an equal number of 0 and 1 actions, and has expected average reward close to 1.5. A simple strategy that always takes action 1 has an average reward 1, and a suboptimal strategy that only takes action 0 has an average reward of 0. We represent states as length-$2N$ vectors containing one-hot indicators for each grid coordinate, and estimate linear Q-functions.

CartPole [Barto et al., 1983]. In the CartPole environment, the goal is to balance an inverted pole attached by an unactuated joint to a cart, which moves along a frictionless rail. There are two actions, corresponding to pushing the cart to the left or right. The observation consists of the position and velocity of the cart, pole angle, and pole velocity at the tip. There is a reward of +1 for every timestep that the pole remains upright. The episodic version of the environment ends if the pole angle is more than 15 degrees from vertical, if the cart moves more than 2.4 units from the center, or after 200 steps. In the infinite-horizon version, if the episode ends after h steps, we return a reward of $h - 200$ and reset. For this task, in addition to the given observation, we extract multivariate Fourier basis features Konidaris et al. [2011] of order 4.

Discussion. In most of our experiments, adaptive learning rate speeds up the convergence of approximate policy iteration, compared to using a constant learning rate as in Politex. The adaptive per-state learning rate is less helpful in CartPole, possibly because observations are continuous and dynamics are smooth, so there is higher generalization across states.

7 CONCLUSION

We have presented AAPI, a model-free learning scheme that can work with function approximation, and enjoys a $\widetilde{O}(T^{2/3})$ regret guarantee in infinite-horizon undiscounted, ergodic MDPs. AAPIimproves upon previous results for this setting by using the slow-changing property of policies in both theory and practice. One direction for future work is improving the policy evaluation stage. While we estimate each value function solely using the τ on-policy transitions, better estimates can potentially be obtained using all data. Using more sophisticated side-information, such as a weighted average of past Q-estimates or an off-policy estimate of the Q-function may also be helpful in practice. Other future work may include practical implementations of the algorithm when trained with neural networks that maintain only a subset of past networks in memory; one possible practical approach is given by Vieillard et al. [2020].

Acknowledgements

Csaba Szepesvári gratefully acknowledges funding from the Canada CIFAR AI Chairs Program, Amii and NSERC.

References

Yasin Abbasi-Yadkori, Kush Bhatia, Peter Bartlett, Nevena Lazić, Csaba Szepesvári, and Gellért Weisz. POLITEX: Regret bounds for policy iteration using expert prediction. In *International Conference on Machine Learning*, pages 3692–3702, 2019a.

Yasin Abbasi-Yadkori, Nevena Lazić, and Csaba Szepesvári. Model-free linear quadratic control via reduction to expert prediction. In *AISTATS*, 2019b.

Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=S1ANxQW0b.

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. Optimality and approximation with policy gradient methods in markov decision processes. In *Conference on Learning Theory*, pages 64–66, 2020.

Peter L. Bartlett. Regal: A regularization based algorithm for reinforcement learning in weakly communicating mdps. In *In Proceedings of the 25th Annual Conference on Uncertainty in Artificial Intelligence*, 2009.

Andrew G Barto, Richard S Sutton, and Charles W Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE transactions on systems, man, and cybernetics*, (5):834–846, 1983.

Dimitri P Bertsekas and Sergey Ioffe. Temporal differences-based policy iteration and applications in neuro-dynamic programming. Technical report, Lab. for Info. and Decision Systems Report LIDS-P-2349, MIT, Cambridge, MA, 1996.

Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. Provably efficient exploration in policy optimization. *arXiv preprint arXiv:1912.05830*, 2019.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Ching-An Cheng, Xinyan Yan, Nathan Ratliff, and Byron Boots. Predictor-corrector policy optimization. In *International Conference on Machine Learning*, pages 1151–1161, 2019.

Eyal Even-Dar, Sham M Kakade, and Yishay Mansour. Online markov decision processes. *Mathematics of Operations Research*, 34(3):726–736, 2009.

Ronan Fruit, Matteo Pirotta, Alessandro Lazaric, and Ronald Ortner. Efficient bias-span-constrained exploration-exploitation in reinforcement learning. In *International Conference on Machine Learning*, 2018.

Mathieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized Markov decision processes. In *ICML*, 2019.

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

QIAN Jian, Ronan Fruit, Matteo Pirotta, and Alessandro Lazaric. Exploration bonus for regret minimization in discrete and continuous average reward mdps. In *Advances in Neural Information Processing Systems*, pages 4891–4900, 2019.

Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is q-learning provably efficient? In *Advances in Neural Information Processing Systems*, pages 4863–4873, 2018.

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. *arXiv preprint arXiv:1907.05388*, 2019.

Pooria Joulani, András György, and Csaba Szepesvári. A modular analysis of adaptive (non-) convex optimization: Optimism, composite objectives, and variational bounds. *arXiv preprint arXiv:1709.02726*, 2017.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *ICML*, volume 2, pages 267–274, 2002.

George Konidaris, Sarah Osentoski, and Philip Thomas. Value function approximation in reinforcement learning using the fourier basis. In *Twenty-fifth AAAI conference on artificial intelligence*, 2011.

H Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. *The Journal of Machine Learning Research*, 18(1):3117–3166, 2017.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Mehryar Mohri and Scott Yang. Accelerating online convex optimization via adaptive prediction. In *Artificial Intelligence and Statistics*, pages 848–856, 2016.

G. Neu, A. Gyorgy, C. Szepesvari, and A. Antos. Online markov decision processes under bandit feedback. *IEEE Transactions on Automatic Control*, 59(3):676–691, March 2014. ISSN 2334-3303. doi: 10.1109/TAC.2013.2292137.

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

Ian Osband, Benjamin Van Roy, and Zheng Wen. Generalization and exploration via randomized value functions. In *International Conference on Machine Learning*, 2016.

Ian Osband, Daniel Russo, Z Wen, and B Van Roy. Deep exploration via randomized value functions. *Journal of Machine Learning Research*, 2017.

Yi Ouyang, Mukul Gagrani, Ashutosh Nayyar, and Rahul Jain. Learning unknown markov decision processes: A thompson sampling approach. In *Advances in Neural Information Processing Systems*, pages 1333–1342, 2017.

Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. *arXiv preprint arXiv:1208.3728*, 2012.

Sasha Rakhlin and Karthik Sridharan. Optimization, learning, and games with predictable sequences. In *Advances in Neural Information Processing Systems*, pages 3066–3074, 2013.

Stephane Ross and J Andrew Bagnell. Reinforcement and imitation learning via interactive no-regret learning. *arXiv preprint arXiv:1406.5979*, 2014.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635, 2011.

Daniel Russo. Worst-case regret bounds for exploration via randomized value functions. In *Advances in Neural Information Processing Systems*, pages 14410–14420, 2019.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897, 2015.

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

Shai Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

Lior Shani, Yonathan Efroni, and Shie Mannor. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. *arXiv preprint arXiv:1909.02769*, 2019.

Mohammad Sadegh Talebi and Odalric-Ambrym Maillard. Variance-aware regret bounds for undiscounted reinforcement learning in mdps. In *Algorithmic Learning Theory*, 2018.

Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Thirtieth AAAI conference on artificial intelligence*, 2016.

Nino Vieillard, Bruno Scherrer, Olivier Pietquin, and Matthieu Geist. Momentum in reinforcement learning. *arXiv preprint arXiv:1910.09322*, 2019.

Nino Vieillard, Tadashi Kozuno, Bruno Scherrer, Olivier Pietquin, Rémi Munos, and Matthieu Geist. Leverage the average: an analysis of regularization in rl. *arXiv preprint arXiv:2003.14089*, 2020.

Chen-Yu Wei, Mehdi Jafarnia-Jahromi, Haipeng Luo, Hiteshi Sharma, and Rahul Jain. Model-free reinforcement learning in infinite-horizon average-reward markov decision processes, 2019.

Lin Yang and Mengdi Wang. Sample-optimal parametric q-learning using linearly additive features. In *International Conference on Machine Learning*, pages 6995–7004, 2019a.

Lin F Yang and Mengdi Wang. Reinforcement leaning in feature space: Matrix bandit, kernels, and regret bound. *arXiv preprint arXiv:1905.10389*, 2019b.

Andrea Zanette, David Brandfonbrener, Emma Brunskill, Matteo Pirotta, and Alessandro Lazaric. Frequentist regret bounds for randomized least-squares value iteration. In *International Conference on Artificial Intelligence and Statistics*, pages 1954–1964, 2020.