Randomized Least Squares Policy Optimization

Haque Ishfaq [1 2] **Zhuoran Yang** [3] **Andrei Lupu** [1 2] **Viet Nguyen** [1 2] **Lewis Liu** [1] **Riashat Islam** [1 2]
Zhaoran Wang [4] **Doina Precup** [1 2 5]

Abstract

Policy Optimization (PO) methods with function approximation are one of the most popular classes of Reinforcement Learning (RL) algorithms. However, designing provably efficient policy optimization algorithms remains a challenge. Recent work in this area has focused on incorporating upper confidence bound (UCB)-style bonuses to drive exploration in policy optimization. In this paper, we present Randomized Least Squares Policy Optimization (RLSPO) which is inspired by Thompson Sampling. We prove that, in an episodic linear kernel MDP setting, RLSPO achieves $\widetilde{\mathcal{O}}(d^{3/2}H^{3/2}\sqrt{T})$ worst-case (frequentist) regret, where H is the number of episodes, T is the total number of steps and d is the feature dimension. Finally, we evaluate RLSPO empirically and show that it is competitive with existing provably efficient PO algorithms.

1. Introduction

One of the fundamental challenges in designing practical reinforcement learning (RL) algorithms is how to effectively trade off exploration and exploitation in an unknown environment. Exploration is well studied in value-based methods with provable guarantees in Markov Decision Processes (MDPs) with the tabular or linearly parameterized dynamics (Jaksch et al., 2010; Yang & Wang, 2019; Jin et al., 2020; Ayoub et al., 2020; Zhou et al., 2020; Cai et al., 2019b). However, in the case of Policy Optimization (PO) algorithms, exploration is less well-understood. PO algorithms model the agent's policy explicitly, typically using a parametric mapping from states to actions (Williams, 1992; Baxter & Bartlett, 2000; Sutton et al., 2000; Konda & Tsitsiklis, 2000), and have become quite popular especially for tasks with continuous states and/or actions (Schulman et al., 2015; 2017b). However, empirical results for deep RL have shown that PO algorithms can require a large number of samples to learn (Burda et al., 2018; Bellemare et al., 2016; Pathak et al., 2017). Recent works, eg., (Im & Halberda, 2012; Mnih et al., 2016; Schulman et al., 2015; Hazan et al., 2019), proposed several heuristics to improve the sample efficiency of PO. However, there is little theory to support such heuristics.

Upper Confidence Bounds (UCB) (Auer et al., 2002; Abbasi-Yadkori et al., 2011) and *Thompson Sampling (TS)* (Thompson, 1933; Osband et al., 2013b; Russo et al., 2018; Osband & Van Roy, 2017) are the two most popular exploration strategies that provide strong theoretical guarantees in both bandit and RL settings. UCB-style algorithms attain the optimal worst-case regret bound in the tabular setting (Jaksch et al., 2010; Azar et al., 2017; Osband & Roy, 2016) and are also provably efficient in the linear setting (Jin et al., 2020; Yang & Wang, 2019; Cai et al., 2019b). However, UCB algorithms are often too conservative empirically, compared to TS-based methods (Chapelle & Li, 2011; Osband et al., 2013a). Yet, theoretical results for PO have mainly been obtained for UCB approaches only (Cai et al., 2019b; Shani et al., 2020; Agarwal et al., 2020).

In this paper, we propose the *Randomized Least Squares Policy Optimization (RLSPO)* algorithm, and establish a worst-case frequentist regret bound for it.

Numerical experiments in the RiverSwim environment and randomly generated low-rank MDPs show that the practical performance of RLSPO is in line with the theoretical guarantees.

The main contributions of this paper are summarized as follows:

- We propose, to the best of our knowledge, the first Thompson Sampling based policy optimization algorithm with provable guarantees.
- Our main theoretical result is a $\widetilde{\mathcal{O}}(d^{3/2}H^{3/2}\sqrt{T})$ regret upper bound in terms of the planning horizon H, the feature dimension d and the total number of transitions T.
- We adopt optimistic sampling techniques to circumvent

[*]Equal contribution [1]Mila [2]School of Computer Science, McGill University [3]Department of Operations Research and Financial Engineering, Princeton University [4] Industrial Engineering & Management Sciences, Northwestern University [5]DeepMind, Montreal. Correspondence to: Haque Ishfaq <haque.ishfaq@mail.mcgill.ca>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

the challenges that arise in analysing the worst-case regret bound.

2. Background

For any set A, $\Delta(A)$ and $\langle \cdot, \cdot \rangle_A$ denote the probability simplex and the inner product over set A, respectively. For a positive definite matrix A and a vector x, we denote $\|x\|_A = \sqrt{x^T A x}$. We denote the cumulative distribution function of the standard Gaussian by $\Phi(\cdot)$. To denote function growth, we use $\widetilde{\mathcal{O}}(\cdot)$, ignoring poly-logarithmic factors.

2.1. Linear Function Approximation in RL

We consider the setting of *linear kernel MDPs*, where both the transition kernels and the reward functions are assumed to be linear in feature maps. Formally, we have the following assumption.

Assumption 2.1 (Linear Kernel MDP, (Ayoub et al., 2020; Zhou et al., 2020))**.** The MDP $(\mathcal{S}, \mathcal{A}, H, \mathbb{P}, r)$ is a *linear kernel MDP* with the kernel feature map $\psi : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}^{d_1}$ and the value feature map $\varphi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^{d_2}$ if, for any $h \in [H]$, there exists a vector $\mu_h \in \mathbb{R}^{d_1}$ with $\|\mu_h\|_2 \leq \sqrt{d_1}$ such that $\mathbb{P}_h(s' \mid s, a) = \langle \psi(s, a, s'), \mu_h \rangle$, for any $(s, a, s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}$, and there exists a vector $w_h \in \mathbb{R}^{d_2}$ with $\|w_h\|_2 \leq \sqrt{d_2}$ such that $r_h(s, a) = \langle \varphi(s, a), w_h \rangle$, for any $(s, a) \in \mathcal{S} \times \mathcal{A}$.

We further assume that for any $(s, a) \in \mathcal{S} \times \mathcal{A}$ and $V : \mathcal{S} \to [0, H]$,

$$\left\| \int_S \psi(s, a, s') \cdot V(s') ds' \right\|_2 \leq \sqrt{d_1} H.$$

Finally, we denote $\max(d_1, d_2) = d$.

Note that a finite MDP is also a linear kernel MDP. Let $\psi(s, a, s') = e_{(s,a,s')}$ be the canonical basis in $\mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|}$. For any $h \in [H]$ and $(s, a, s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}$, we can represent the transition kernel as $\mathbb{P}_h(s' \mid s, a) = \langle e_{(s,a,s')}, \mu_h \rangle$. Similarly, letting $\varphi(s, a) = e_{(s,a)}$ be the canonical basis in $\mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, we can represent the reward function as $r_h(s, a) = \langle e_{(s,a)}, w_h \rangle$.

3. Algorithm

In this section, we describe the RLSPO algorithm which alternates between two main steps: (i) policy improvement and (ii) policy evaluation within each episode.

Policy Improvement Step. In Algorithm 1, lines 5-10 execute the policy improvement step to obtain a new policy π^k. For any $(k, h) \in [K] \times [H]$, and $(s, a) \in \mathcal{S} \times \mathcal{A}$, we parameterize π_h^k as,

$$\pi_h^k(a \mid s) = \frac{\exp\left(E_h^k(s, a)\right)}{\sum_{a' \in \mathcal{A}} \exp\left(E_h^k(s, a')\right)}, \tag{3.1}$$

Algorithm 1 Randomized Least-Squares Policy Optimization (RLSPO)

1: Set M to be a fixed integer and λ to be positive number.
2: Initialize $\{Q_h^0\}_{h=1}^H$ as zero functions and $\{\pi_h^0\}_{h=1}^H$ as uniform distributions on $\mathcal{A}$.
3: **For** episode $k = 1, 2, \ldots, K$ **do**
4: Receive the initial state s_1^k.
5: **For** step $h = 1, 2, \ldots, H$ **do**
6: Update policy
7: $\pi_h^k(\cdot \mid \cdot) \propto \pi_h^{k-1}(\cdot \mid \cdot) \cdot \exp\{\alpha Q_h^{k-1}(\cdot, \cdot)\}$
8: Take the action following $a_h^k \sim \pi_h^k(\cdot \mid s_h^k)$.
9: Observe reward $r_h(s_h^k, a_h^k)$ and
10: get next state s_{t+1}^k.
11: Initialize V_{H+1}^k as a zero function.
12: **For** step $h = H, H-1, \ldots, 1$ **do**
13: $\Sigma_h^k \leftarrow \sum_{i=1}^{k-1} \phi_h^i(s_h^i, a_h^i) \phi_h^i(s_h^i, a_h^i)^\top + \lambda I$.
14: $\widehat{\theta}_h^k \leftarrow (\Sigma_h^k)^{-1} \sum_{i=1}^{k-1} V_{h+1}^i(s_{h+1}^i) \phi_h^i(s_h^i, a_h^i)$.
15: $\phi_h^k(\cdot, \cdot) \leftarrow \int_{\mathcal{S}} \psi(\cdot, \cdot, s') V_{h+1}^k(s') ds'$.
16: $\Lambda_h^k \leftarrow \sum_{i=1}^{k-1} \varphi(s_h^i, a_h^i) \varphi(s_h^i, a_h^i)^\top + \lambda I$.
17: $\widehat{w}_h^k \leftarrow (\Lambda_h^k)^{-1} \sum_{i=1}^{k-1} r_h(s_h^i, a_h^i) \varphi(s_h^i, a_h^i)$.
18: Sample i.i.d. $\{\xi_h^{k,j}\}_{j \in [M]} \sim \mathcal{N}(0, \sigma_1^2 (\Sigma_h^k)^{-1})$.
19: Sample i.i.d. $\{\epsilon_h^{k,j}\}_{j \in [M]} \sim \mathcal{N}(0, \sigma_2^2 (\Lambda_h^k)^{-1})$.
20: $\widetilde{P}_h \widetilde{V}_{h+1}^{k,j}(\cdot, \cdot) \leftarrow \phi_h^k(\cdot, \cdot)^\top (\widehat{\theta}_h^k + \xi_h^{k,j})$ for $j \in [M]$.
21: $\widetilde{r}_h^{k,j}(\cdot, \cdot) \leftarrow \varphi(\cdot, \cdot)^\top (\widehat{w}_h^k + \epsilon_h^{k,j})$ for $j \in [M]$.
22: $Q_h^k(\cdot, \cdot) \leftarrow \min\{\max_{j \in [M]} \widetilde{r}_h^{k,j}(\cdot, \cdot) +$
23: $\max_{j \in [M]} \widetilde{P}_h \widetilde{V}_{h+1}^{k,j}(\cdot, \cdot), H - h + 1\}^+$
24: $V_h^k(\cdot) \leftarrow \langle Q_h^k(\cdot, \cdot), \pi_h^k(\cdot \mid \cdot) \rangle_{\mathcal{A}}$.

where E_h^k is the potential function with the update rule:

$$E_h^k(s, a) = E_h^{k-1}(s, a) + \alpha Q_h^{\pi^{k-1}, k-1}(s, a). \tag{3.2}$$

Here, $\alpha > 0$ is the step-size for policy improvement. Moreover, we initialize the potential function to zero. For any $s \in \mathcal{S}$ and $h \in [H]$ the updated policy π^k in (3.1) takes the following closed form:

$$\pi_h^k(\cdot \mid s) \propto \pi_h^{k-1}(\cdot \mid s) \cdot \exp\{\alpha Q_h^{\pi^{k-1}, k-1}(s, \cdot)\}. \tag{3.3}$$

However, the Q-function $Q_h^{\pi^{k-1}, k-1}$ is yet to be estimated in Algorithm 1 through subsequent policy evaluation steps. Thus, we replace $Q_h^{\pi^{k-1}, k-1}$ with the estimated Q-function Q_h^{k-1} in (3.3), as shown in line 1 in Algorithm 1. This form of policy improvement steps is commonly known as exponential gradient updates (Kakade, 2002; Agarwal et al., 2019). Finally, we note that with Q_h^{k-1} in (3.3), π_h^k is the solution to the following optimization problem:

$$\begin{aligned} \arg\max_{\pi_h \in \Delta(\mathcal{A} \mid \mathcal{S})} \; & \mathbb{E}_{\pi^{k-1}} \Big[\langle Q_h^{k-1}(s_h, \cdot), \pi_h(\cdot \mid s_h) \rangle_{\mathcal{A}} \\ & + \frac{1}{\alpha} \mathcal{D}_{KL}\big(\pi_h(\cdot \mid s_h) \| \, \pi_h^{k-1}(\cdot \mid s_h)\big) \Big] . \end{aligned} \tag{3.4}$$

The optimization problem in (3.4) builds on related approaches by (Cai et al., 2019b) and (Liu et al., 2019), which also closely resembles the one-step iteration objective in the proximal policy optimization (PPO) algorithm (Schulman et al., 2017b) and trust-region policy optimization (TRPO) (Schulman et al., 2015).

Policy Evaluation Step. At the end of the k-th episode, RLSPO performs one iteration of least-squares temporal difference learning (LSTD) (Bradtke & Barto, 1996; Boyan, 2002) to evaluate policy π^k based on the $k-1$ historical trajectories (lines 12-24 in Algorithm 1). For each $h \in [H]$, in lieu of estimating $\mathbb{P}_h V_{h+1}^{\pi_k}$ in the Bellman equations (A.1), we estimate $\mathbb{P}_h V_{h+1}^k$ by ${\phi_h^k}^\top \theta_h^k$ where θ_h^k is iteratively updated by solving the regularized least-squares problem over θ_h:

$$\widehat{\theta}_h^k \leftarrow \mathop{\arg\min}_{\theta_h \in \mathbb{R}^{d_1}} \sum_{i=1}^{k-1} (V_{h+1}^i(s_{h+1}^i) - \phi_h^i(s_h^i, a_h^i)^\top \theta_h)^2 + \lambda \|\theta_h\|_2^2. \tag{3.5}$$

Here $\phi_h^i(\cdot,\cdot) = \int_{\mathcal{S}} \psi(\cdot,\cdot,s') V_{h+1}^i(s') ds'$, $V_{h+1}^i(\cdot) = \langle Q_{h+1}^i(\cdot,\cdot), \pi_{h+1}^i(\cdot\,|\,\cdot)\rangle_{\mathcal{A}}$ for $h \in [H-1]$ and $V_{H+1}^i = 0$, and $\lambda > 0$ is the regularization parameter which is specified in Theorem 4.1. Note that ϕ_h^i can be interpreted as the feature vector induced by the estimated value function of all the possible next states.

Likewise, we estimate r_h^k by $\varphi^\top w_h^k$, where w_h^k is updated by solving another regularized least squares problem over w_h:

$$\widehat{w}_h^k \leftarrow \mathop{\arg\min}_{w_h \in \mathbb{R}^{d_2}} \sum_{i=1}^{k-1} (r_h(s_h^i, a_h^i) - \varphi_h(s_h^i, a_h^i)^\top w_h)^2 + \lambda \|w_h\|_2^2,$$

where $\lambda > 0$ is the regularization parameter. Then, we perturb the estimated parameters $\widehat{\theta}_h^k$ and $\widehat{w}_h^k$ by adding mean-zero Gaussian noise and employ the optimistic sampling technique to encourage exploration, which we discuss below.

Exploration with Gaussian Noise. As described in (Abeille et al., 2017), TS does not necessarily need to sample from an actual Bayesian posterior distribution. In fact, any distribution with suitable concentration and anti-concentration properties would guarantee a small regret. More precisely, instead of sampling from a true Bayesian posterior, we can sample a noise value ξ_h^k from Gaussian distribution $\mathcal{N}(0, \sigma_1^2(\Sigma_h^k)^{-1})$ and add it to our RLS estimate $\widehat{\theta}_h^k$ to get a perturbed estimate of $\mathbb{P}_h V_{h+1}^k$. This process is equivalent to sampling from the Gaussian distribution $\mathcal{N}(\widehat{\theta}_h^k, \sigma_1^2(\Sigma_h^k)^{-1})$ where σ_1^2 is the posterior inflation scalar. The variance of the noise ξ_h^k is proportional to the inverse of the regularized design matrix $\Sigma_h^k = \sum_{i=1}^{k-1} \phi_h^i(s_h^i, a_h^i)\phi_h^i(s_h^i, a_h^i)^\top + \lambda I$, where the ϕ_h^i's are the estimated value function induced features of the state action pairs encountered in previous episodes. Intuitively, this choice of Gaussian distribution results in perturbations that exhibit higher variances in the less explored directions. In a similar manner, we can sample a noise value ϵ_h^k from Gaussian distribution $\mathcal{N}(0, \sigma_2^2(\Lambda_h^k)^{-1})$, where $\Lambda_h^k = \sum_{i=1}^{k-1} \varphi(s_h^i, a_h^i)\varphi(s_h^i, a_h^i)^\top + \lambda I$, and add it to RLS estimate $\widehat{w}_h^k$ to get a perturbed estimate of r_h^k.

Optimistic Sampling. In the analysis of the randomized TS algorithm, one key challenge is to show that the algorithm is optimistic with a constant probability. The optimistic sampling technique which we present now is a key ingredient in circumventing this technical difficulty in our worst-case regret analysis. In optimistic sampling, in order to boost the probability of being optimistic, we sample M independent noises $\{\xi_h^{k,j}\}_{j\in[M]}$ and $\{\epsilon_h^{k,j}\}_{j\in[M]}$ from $\mathcal{N}(0, \sigma_1^2(\Sigma_h^k)^{-1})$ and $\mathcal{N}(0, \sigma_2^2(\Lambda_h^k)^{-1})$ respectively. We specify the exact value of M, σ_1 and σ_2 in Theorem 4.1. Then, we form the optimistic estimate for the action value function

$$Q_h^k(\cdot,\cdot) = \min\Big\{ \max_{j\in[M]} \widetilde{r}_h^{k,j}(\cdot,\cdot) + \max_{j\in[M]} \widetilde{P}_h \widetilde{V}_{h+1}^{k,j}(\cdot,\cdot), H - h + 1 \Big\}^+, \tag{3.6}$$

where for all $j \in [M]$,

$$\widetilde{P}_h \widetilde{V}_{h+1}^{k,j}(\cdot,\cdot) = \phi_h^k(\cdot,\cdot)^\top (\widehat{\theta}_h^k + \xi_h^{k,j}), \tag{3.7}$$

and

$$\widetilde{r}_h^{k,j}(\cdot,\cdot) = \varphi(\cdot,\cdot)^\top (\widehat{w}_h^k + \epsilon_h^{k,j}). \tag{3.8}$$

4. Main Result

Our main result is a frequentist worst-case regret bound for RLSPO under linear kernel MDP setting from Assumption 2.1.

Theorem 4.1. Let $\alpha = \sqrt{2\log|\mathcal{A}|/H^2K}$, $\lambda = 1$, $\sigma_1 = \widetilde{O}(H\sqrt{d})$, $\sigma_2 = \sqrt{d}$ and $M = d\log(\delta/18)/\log c_0$, where $c_0 = \Phi(1) = 0.841$ and $\delta \in (0,1]$. Under Assumption 2.1 and the assumption that $\log|\mathcal{A}| = O(d^3[\log(dT/\delta)]^2)$, the regret of Algorithm 1 satisfies

$$\text{Regret}(T) \le \widetilde{\mathcal{O}}(d^{3/2}H^{3/2}\sqrt{T}),$$

with probability at least $1-\delta$.

Theorem 4.1 asserts that when λ, α, σ and M are set properly, RLSPO will suffer total regret of at most $\widetilde{\mathcal{O}}(d^{3/2}H^{3/2}\sqrt{T})$. We emphasize that our regret is completely independent of $|\mathcal{S}|$ and $|\mathcal{A}|$, which is crucial in the large state-space setting where we need to use function approximation.

Remark 4.2. As can be seen in Table 1, there is a $\sqrt{d}$ gap between the regret bound of RLSPO and OPPO. As shown in (Hamidi & Bayati, 2020), this gap of $\sqrt{d}$ in worst-case regret between UCB and TS based method is unavoidable. When converted to linear bandit by setting $H = 1$, our regret bound matches the best known regret upper bound for LinTS due to (Abeille et al., 2017).

Remark 4.3. The choice of learning rate α and it's dependency on K comes from the proof of Lemma F.2 and Lemma H.2 in the appendix which are inspired from the analysis of mirror descent algorithm in optimization literature. If K is unknown, we can use "doubling trick"(Besson & Kaufmann, 2018) from online learning literature - at every power of 2 episode (i.e. at episode 2^i for some i), we reset and assume $K = 2^i$. It is standard knowledge that this trick increases the overall regret by only a constant factor. The assumption $\log|\mathcal{A}| = O(d^3[\log(dT/\delta)]^2)$ is used when we apply Lemma F.2 in the proof of Theorem 4.1. In practice, this assumption does not really impose a limit on the number of actions since we can still have exponentially many actions.

4.1. Proof and key lemmas

Now we describe some key lemmas that are used in the proof of Theorem 4.1. We use the following quantities in the statements of our lemmas: $\beta(\delta) = \widetilde{O}(H^2 d)$, $\nu(\delta) = \widetilde{O}(H^2 d)$, $\gamma(\delta) = \widetilde{O}(H^2 d^2)$, and $\alpha(\delta) = \widetilde{O}(d^2)$. The exact values of these quantities, proofs and remaining details can be found in the appendix.

First, we consider the events $\mathcal{G}_h^k(\xi, \delta)$ and $\mathcal{G}_h^k(\epsilon, \delta)$ when for fixed episode k and time-step h, the maximum norms of the sampled Gaussian noise in Line 1 and 1 in Algorithm 1 are bounded. When the aforementioned events occur for every time-step in each episode, we denote the event by $\mathcal{G}(K, H, \delta)$. Concretely we give the following definitions.

Definition 4.4 (Good events)**.** For any $\delta > 0$ and positive integer M, we define the following random events

$$\mathcal{G}_h^k(\xi, \delta) \stackrel{\text{def}}{=} \Big\{ \max_{j\in[M]} \|\xi_h^{k,j}\|_{\Sigma_h^k} \le \sqrt{\gamma(\delta)} \Big\},$$
$$\mathcal{G}_h^k(\epsilon, \delta) \stackrel{\text{def}}{=} \Big\{ \max_{j\in[M]} \|\epsilon_h^{k,j}\|_{\Lambda_h^k} \le \sqrt{\alpha(\delta)} \Big\},$$
$$\mathcal{G}(K, H, \delta) \stackrel{\text{def}}{=} \bigcap_{k\le K} \bigcap_{h\le H} \big(\mathcal{G}_h^k(\xi, \delta) \cap \mathcal{G}_h^k(\epsilon, \delta)\big).$$

The next lemma shows that the event $\mathcal{G}(K, H, \delta)$ occurs with high probability. The crux of the proof of Theorem 4.1 is to show that under this event the regret is small.

Lemma 4.5 (Good event probability)**.** For any positive integer K and any $\delta > 0$, we would have the event $\mathcal{G}(K, H, \delta')$ with probability at least $1 - \delta$, where $\delta' = \delta/2MT$.

Concentration Events. One key tool that is used in the proof of Theorem 4.1, is the anti-concentration property of the maximum of samples of Gaussian random variables. Characterization of this property in our context allows us to ensure frequent optimism.

Lemma 4.6. Consider a d-dimensional multivariate normal distribution $N(0, A\Sigma^{-1})$ where A is a scalar. Let $\eta_1, \eta_2, \ldots, \eta_M$ be M independent samples from the distribution. Then for any $\delta > 0$

$$\mathbb{P}\left(\max_{j\in[M]} \|\eta_j\|_\Sigma \le c\sqrt{dA\log(d/\delta)} \right) \ge 1 - M\delta,$$

where c is some absolute constant.

The next lemma characterizes the concentration behavior of value function. This result is achieved by using the concentration properties of self-normalizing processes (Abbasi-Yadkori et al., 2011).

Lemma 4.7. Let $\lambda = 1$ in Algorithm 1. For any $\delta > 0$, conditioned on the event $\mathcal{G}(K, H, \delta)$, we have for all $(k, h) \in [K] \times [H]$,

$$\Big\| \sum_{i=1}^{k-1} \phi_h^i(s_h^i, a_h^i) \cdot \Big(V_{h+1}^i(s_{h+1}^i) - (\mathbb{P}_h V_{h+1}^i)(s_h^i, a_h^i) \Big) \Big\|_{(\Sigma_h^k)^{-1}} \le C_1 \sqrt{dH^2 \log(dT/\delta)}, \tag{4.1}$$

with probability at least $1 - \delta$ where $C_1 > 0$ is an absolute constant.

Now, we define the model prediction error,

$$l_h^k(s, a) = r_h^k(s, a) + \mathbb{P}_h V_{h+1}^k(s, a) - Q_h^k(s, a). \tag{4.2}$$

This depicts the prediction error using V_{h+1}^k instead of $V_{h+1}^{\pi^k}$ in the Bellman equations (A.1). Using Lemma 4.6, we prove the following lemma in Appendix E that characterizes the model prediction error l_h^k in Algorithm 1.

Lemma 4.8 (stochastic upper confidence bound)**.** Let $\lambda = 1$ in Algorithm 1. For any $\delta > 0$, conditioned on the event $\mathcal{G}(K, H, \delta)$, for all $(s, a, h, k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]$, with probability at least $1 - (\delta + 2c_0^M)$, we have

$$l_h^k(s, a) \le 0, \tag{4.3}$$

and

$$-l_h^k(s, a) \le \Big(\sqrt{\alpha(\delta)} + \sqrt{d} \Big) \big\|\varphi(s, a)\big\|_{(\Lambda_h^k)^{-1}} + \Big(\sqrt{\nu(\delta)} + \sqrt{\gamma(\delta)} \Big) \big\|\phi_h^k(s, a)\big\|_{(\Sigma_h^k)^{-1}}, \tag{4.4}$$

where $c_0 = \Phi(1) = 0.841$.

Lemma 4.8 states that our state-action value estimate is uniformly optimistic at least with a constant probability. Moreover, due to the uncertainty that comes from observing finite amount of historical data, the model prediction error $l_h^k(s,a)$ can be possibly large for state-action pairs (s,a) that are less visited. Also, from Lemma 4.8, we see the importance of optimistic sampling. Since $c_0 < 1$, for $M \geq 1$, c_0^M is a decreasing function in M. Thus as we increase the number of noise samples M, we increase the probability of having the inequalities (4.3) and (4.4) to hold.

5. Conclusion and future work

In this paper, we proposed a Thompson Sampling based policy optimization algorithm and proved that it has a worst-case regret bound of $\widetilde{\mathcal{O}}(d^{3/2}H^{3/2}\sqrt{T})$. Our proof involves an optimistic sampling technique for increasing the probability of being optimistic, which could be of broader interest. Our empirical results demonstrate that, similarly to value-based methods, TS has the potential to outperform alternative exploration strategies such as UCB in policy optimization algorithms.

References

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

Abbasi-Yadkori, Y., Bartlett, P., Bhatia, K., Lazic, N., Szepesvári, C., and Weisz, G. POLITEX: Regret bounds for policy iteration using expert prediction. volume 97, pp. 3692–3702, 2019.

Abeille, M., Lazaric, A., et al. Linear thompson sampling revisited. *Electronic Journal of Statistics*, 11(2):5165–5197, 2017.

Agarwal, A., Kakade, S. M., Lee, J. D., and Mahajan, G. Optimality and approximation with policy gradient methods in markov decision processes. *CoRR*, abs/1908.00261, 2019. URL http://arxiv.org/abs/1908.00261.

Agarwal, A., Henaff, M., Kakade, S., and Sun, W. Pc-pg: Policy cover directed exploration for provable policy gradient learning. *arXiv preprint arXiv:2007.08459*, 2020.

Agrawal, S. and Goyal, N. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pp. 127–135, 2013.

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

Ayoub, A., Jia, Z., Szepesvari, C., Wang, M., and Yang, L. Model-based reinforcement learning with value-targeted regression. In *International Conference on Machine Learning*, pp. 463–474. PMLR, 2020.

Azar, M. G., Osband, I., and Munos, R. Minimax regret bounds for reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 263–272. JMLR. org, 2017.

Baxter, J. and Bartlett, P. L. Direct gradient-based reinforcement learning. In *2000 IEEE International Symposium on Circuits and Systems. Emerging Technologies for the 21st Century. Proceedings (IEEE Cat No. 00CH36353)*, volume 3, pp. 271–274. IEEE, 2000.

Bellemare, M., Srinivasan, S., Ostrovski, G., Schaul, T., Saxton, D., and Munos, R. Unifying count-based exploration and intrinsic motivation. In *Advances in neural information processing systems*, pp. 1471–1479, 2016.

Besson, L. and Kaufmann, E. What doubling tricks can and can't do for multi-armed bandits. *arXiv preprint arXiv:1803.06971*, 2018.

Bhandari, J. and Russo, D. Global optimality guarantees for policy gradient methods. *arXiv preprint arXiv:1906.01786*, 2019.

Boyan, J. A. Technical update: Least-squares temporal difference learning. *Machine learning*, 49(2-3):233–246, 2002.

Bradtke, S. J. and Barto, A. G. Linear least-squares algorithms for temporal difference learning. *Machine learning*, 22(1-3):33–57, 1996.

Burda, Y., Edwards, H., Storkey, A., and Klimov, O. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

Cai, Q., Yang, Z., Jin, C., and Wang, Z. Provably efficient exploration in policy optimization. *CoRR*, abs/1912.05830, 2019a. URL http://arxiv.org/abs/1912.05830.

Cai, Q., Yang, Z., Jin, C., and Wang, Z. Provably efficient exploration in policy optimization. *arXiv preprint arXiv:1912.05830*, 2019b.

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

Fazel, M., Ge, R., Kakade, S., and Mesbahi, M. Global convergence of policy gradient methods for the linear quadratic regulator. In *International Conference on Machine Learning*, pp. 1467–1476. PMLR, 2018.

Hamidi, N. and Bayati, M. On worst-case regret of linear thompson sampling. *arXiv preprint arXiv:2006.06790*, 2020.

Hazan, E., Kakade, S. M., Singh, K., and Soest, A. V. Provably efficient maximum entropy exploration. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pp. 2681–2691, 2019. URL http://proceedings.mlr.press/v97/hazan19a.html.

Im, H. and Halberda, J. The effects of sampling and internal noise on the representation of ensemble average size. *Attention, perception psychophysics*, 75, 11 2012. doi: 10.3758/s13414-012-0399-4.

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

Jin, C., Yang, Z., Wang, Z., and Jordan, M. I. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pp. 2137–2143. PMLR, 2020.

Kakade, S. M. A natural policy gradient. In *Advances in neural information processing systems*, pp. 1531–1538, 2002.

Konda, V. R. and Tsitsiklis, J. N. Actor-critic algorithms. In *Advances in neural information processing systems*, pp. 1008–1014, 2000.

Liu, B., Cai, Q., Yang, Z., and Wang, Z. Neural proximal/trust region policy optimization attains globally optimal policy. *arXiv preprint arXiv:1906.10306*, 2019.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T. P., Harley, T., Silver, D., and Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pp. 1928–1937, 2016. URL http://proceedings.mlr.press/v48/mniha16.html.

Nachum, O., Norouzi, M., Xu, K., and Schuurmans, D. Bridging the gap between value and policy based reinforcement learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pp. 2775–2785, 2017.

Osband, I. and Roy, B. V. On lower bounds for regret in reinforcement learning. 2016.

Osband, I. and Van Roy, B. Why is posterior sampling better than optimism for reinforcement learning? In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 2701–2710. JMLR. org, 2017.

Osband, I., Russo, D., and Roy, B. V. (more) efficient reinforcement learning via posterior sampling. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pp. 3003–3011, 2013a.

Osband, I., Russo, D., and Van Roy, B. (more) efficient reinforcement learning via posterior sampling. In *Advances in Neural Information Processing Systems*, pp. 3003–3011, 2013b.

Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 16–17, 2017.

Russo, D., Roy, B. V., Kazerouni, A., Osband, I., and Wen, Z. A tutorial on thompson sampling. *Foundations and Trends in Machine Learning*, 11(1):1–96, 2018. doi: 10.1561/2200000070. URL https://doi.org/10.1561/2200000070.

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

Schulman, J., Abbeel, P., and Chen, X. Equivalence between policy gradients and soft q-learning. *CoRR*, abs/1704.06440, 2017a. URL http://arxiv.org/abs/1704.06440.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017b.

Shani, L., Efroni, Y., Rosenberg, A., and Mannor, S. Optimistic policy optimization with bandit feedback. In *International Conference on Machine Learning*, pp. 8604–8613. PMLR, 2020.

Singh, S., Lewis, R. L., Barto, A. G., and Sorg, J. Intrinsically motivated reinforcement learning: An evolutionary perspective. *IEEE Trans. on Auton. Ment. Dev.*, 2(2):70–82, June 2010. ISSN 1943-0604. doi: 10.1109/TAMD.2010.2051031. URL https://doi.org/10.1109/TAMD.2010.2051031.

Strehl, A. L. and Littman, M. L. An analysis of model-based interval estimation for markov decision processes. *Journal of Computer and System Sciences*, 74(8):1309–1331, 2008.

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

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

Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Wang, Y., Wang, R., Du, S. S., and Krishnamurthy, A. Optimism in reinforcement learning with generalized linear function approximation. *arXiv preprint arXiv:1912.04136*, 2019.

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

Yang, L. and Wang, M. Sample-optimal parametric q-learning using linearly additive features. In *International Conference on Machine Learning*, pp. 6995–7004. PMLR, 2019.

Zanette, A., Brandfonbrener, D., Brunskill, E., Pirotta, M., and Lazaric, A. Frequentist regret bounds for randomized least-squares value iteration. In *International Conference on Artificial Intelligence and Statistics*, pp. 1954–1964. PMLR, 2020a.

Zanette, A., Lazaric, A., Kochenderfer, M., and Brunskill, E. Learning near optimal policies with low inherent bellman error. In *International Conference on Machine Learning*, pp. 10978–10989. PMLR, 2020b.

Zheng, Z., Oh, J., and Singh, S. On learning intrinsic rewards for policy gradient methods. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pp. 4649–4659, 2018.

Zhou, D., He, J., and Gu, Q. Provably efficient reinforcement learning for discounted mdps with feature mapping. *arXiv preprint arXiv:2006.13165*, 2020.

A. Background in Episodic Reinforcement Learning

We consider undiscounted episodic MDPs of the form $(\mathcal{S}, \mathcal{A}, H, \mathbb{P}, r)$, where $\mathcal{S}$ is the (possibly uncountable) state space, $\mathcal{A}$ is a finite action space, H is the number of steps in each episode, $\mathbb{P} = \{\mathbb{P}_h\}_{h=1}^H$ are the state transition probability distributions, and $r = \{r_h\}_{h=1}^H$ are the reward functions. For each $h \in [H]$, $\mathbb{P}_h(\cdot \mid s, a)$ is the transition kernel over the next states if action a is taken at state s during the h-th time step of the episode. Also, $r_h : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the deterministic reward function at step h.[1]

A (stochastic) policy π is a collection of H functions $\{\pi_h : \mathcal{S} \to \Delta(\mathcal{A})\}_{h\in[H]}$. We denote by $\pi(\cdot \mid s)$ the action distribution of policy π for state s, and by π^* the optimal policy.

The value function $V_h^\pi : \mathcal{S} \to \mathbb{R}$ at step $h \in [H]$ is the expected sum of remaining rewards until the end of the episode, received under π when starting from $s_h = s$,

$$V_h^\pi(s) = \mathbb{E}_\pi\Big[\sum_{h'=h}^{H} r_{h'}(s_{h'}, a_{h'}) \mid s_h = s\Big].$$

The action-value function $Q_h^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the expected sum of remaining rewards under π when starting from state-action pair $(s_h, a_h) = (s, a)$,

$$Q_h^\pi(s, a) = r_h(s, a) + \mathbb{E}_\pi\Big[\sum_{h'=h+1}^{H} r_{h'}(s_{h'}, a_{h'}) \mid s_h = s, a_h = a\Big].$$

We denote $V_h^*(s) = V_h^{\pi^*}(s)$ and $Q_h^*(s, a) = Q_h^{\pi^*}(s, a)$. Moreover, to simplify notation, we denote $[P_h V_{h+1}](s, a) = \mathbb{E}_{s'\sim\mathbb{P}_h(\cdot|s,a)} V_{h+1}(s')$, where P_h is the operator induced by the transition kernel $\mathbb{P}_h(\cdot|\cdot, \cdot)$.

Recall that value functions obey the Bellman equations:

$$\begin{aligned} Q_h^\pi(s, a) &= (r_h + \mathbb{P}_h V_{h+1}^\pi)(s, a), \\ V_h^\pi(s) &= \langle Q_h^\pi(s, \cdot), \pi_h(\cdot \mid s)\rangle_{\mathcal{A}}, \\ V_{H+1}^\pi(s) &= 0 \end{aligned} \tag{A.1}$$

The agent aims to learn the optimal policy by acting in the environment for K episodes. Before starting each episode $k \in [K]$, the agent chooses a policy π^k and an adversary chooses the initial state s_1^k. Then, at each time step $h \in [H]$, the agent observes $s_h^k \in \mathcal{S}$, picks an action $a_h^k \in \mathcal{A}$, receives a reward $r_h(s_h^k, a_h^k)$ and transitions to the next state $s_{h+1}^k \sim \mathbb{P}_h(\cdot \mid s_h^k, a_h^k)$. The episode ends after the agent collects the H-th reward and reaches the state s_{H+1}^k. The suboptimality of the agent at the k-th episode is captured by the difference between $V_1^{\pi^k}(s_1^k)$ and $V_1^*(s_1^k)$. The total regret after K episodes is

$$\text{Regret}(K) = \sum_{k=1}^{K} \Big[V_1^*(s_1^k) - V_1^{\pi^k}(s_1^k)\Big]. \tag{A.2}$$

B. Related Work

Exploration in Policy Optimization. Most commonly used exploration tactics for PO methods are based on heuristics. For example, works such as (Mnih et al., 2016; Schulman et al., 2017a; Nachum et al., 2017) use entropy regularization to induce exploration. In yet another class of works, intrinsic motivation (Singh et al., 2010; Pathak et al., 2017; Zheng et al., 2018) is added to the rewards to achieve the same effect. However, none of the these works provide theoretical guarantees for their methods.

Thompson Sampling in linear bandits. (Agrawal & Goyal, 2013) provided a TS algorithm for linear bandit problems in which the selected actions and the observed rewards are used to update a Gaussian prior over the parameter space. In the linear bandit setting, Abeille et al. (2017) showed that in order to design TS algorithms, sampling from an actual Bayesian posterior is not necessary, and the same order of frequentist regret can be obtained as long as the distribution from which TS samples obeys certain concentration and anti-concentration properties. In this work, we follow (Abeille et al., 2017) and adopt a randomized algorithm formulation for RLSPO.

[1] We consider deterministic reward functions for notational simplicity. Our results straightforwardly generalize to stochastic reward functions. Moreover, note that we are assuming that rewards are in $[0, 1]$ for normalization.

Algorithm	Regret	Policy Based	Exploration	Setting
RLSPO	$\widetilde{\mathcal{O}}(d^{3/2}H^{3/2}\sqrt{T})$	Yes	TS	LKM
OPPO (Cai et al., 2019a)	$\widetilde{\mathcal{O}}(dH^{3/2}\sqrt{T})$	Yes	UCB	LKM
POMD (Shani et al., 2020)	$\widetilde{\mathcal{O}}(\sqrt{S^2AH^3T})$	Yes	UCB	Tabular
OPT-LSVI (Jin et al., 2020)	$\widetilde{\mathcal{O}}(d^{3/2}H^{3/2}\sqrt{T})$	No	UCB	LM
OPT-RLSVI (Zanette et al., 2020a)	$\widetilde{\mathcal{O}}(d^2H^2\sqrt{T})$	No	TS	LM
LSVI-UCB (Wang et al., 2019)	$\widetilde{\mathcal{O}}(d^{3/2}H\sqrt{T})$	No	UCB	OC

Table 1: Regret bounds summary

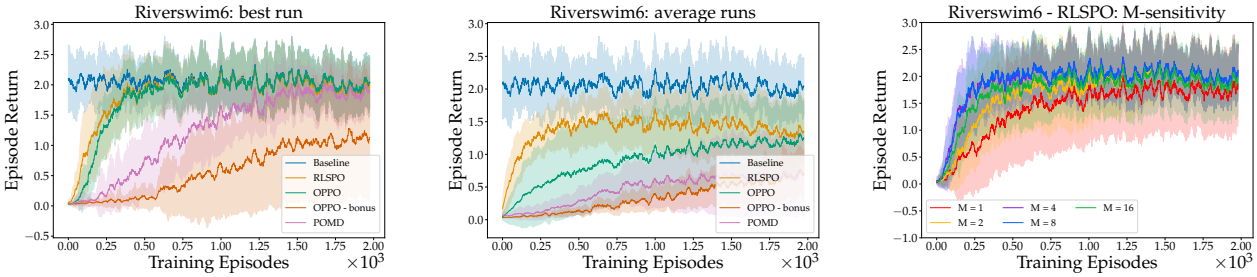
Regret minimization in linear setting. Recently, there has been a series of works that studies provably efficient RL algorithms with linear function approximation (Jin et al., 2020; Cai et al., 2019b; Zanette et al., 2020a;b). Among these studies, Jin et al. (2020) proposes a UCB-style optimistic modification of Least-Squares Value Iteration (OPT-LSVI) and provides regret bound in the linear MDP setting. More recently, Zanette et al. (2020b) proposed the optimistic LSVI type algorithm ELEANOR, which works under a low inherent Bellman error assumption, slightly weaker than the linear MDP assumption. However, their algorithm is computationally intractable. Another recent work (Wang et al., 2019) introduces a new expressivity condition named *optimistic closure* under which they propose a variant of optimistic LSVI.

Compared to value based methods, there has been much less work on the theory of policy optimization - both from the computational and statistical perspective. Some recent works (Fazel et al., 2018; Abbasi-Yadkori et al., 2019; Bhandari & Russo, 2019; Liu et al., 2019; Agarwal et al., 2019) studied the computational efficiency and convergence properties of policy optimization. However, these works rely on the assumption of either known transition model or the existence of a well-explored behavior policy, which bypasses the need to address exploration-exploitation trade-off. On the exploration side, the most related work to ours is the OPPO algorithm proposed by Cai et al. (2019b). OPPO achieves a regret bound of $\widetilde{\mathcal{O}}(dH^{3/2}\sqrt{T})$ under the full-information feedback linear kernel MDP setting. Even though there are some structural similarities, the RLSPO algorithm we proposed is fundamentally different from OPPO due to the way exploration is performed in each algorithm. OPPO encourages exploration by adding a UCB-style bonus function to the estimated action-value function, and thus falls under the *optimism in the face of uncertainty* class of algorithms. In comparison, RLSPO performs exploration using a Thompson Sampling approach. In terms of the frequentist regret bound, RLSPO is worse than OPPO by a factor of $\sqrt{d}$. Similar to OPPO, Shani et al. (2020) proposes Optimistic POMD, an optimistic variant of TRPO (Schulman et al., 2015) under bandit feedback. However, their work only considers tabular MDPs. Another recent work (Agarwal et al., 2020) proposes the PC-PG algorithm, which uses an ensemble of learned policies that provides a policy cover over the state space.

In regards to Thompson Sampling, the closest work to ours is the OPT-RLSVI algorithm proposed by Zanette et al. (2020a), where they proved a frequentist regret bound under linear MDP assumption. OPT-RLSVI induces exploration by perturbing the least-squares approximation of the action-value function using a mean-zero Gaussian noise. However, OPT-RLSVI is a value iteration algorithm, while RLSPO is a policy optimization algorithm. Moreover, to achieve TS-like exploration, they sample just one noise value at each time step, whereas our optimistic sampling technique samples multiple noise values (see Sec. 3). Consequently, their construction of the Q-function and analysis technique cannot be directly adapted to our policy optimization problem. However, our analysis based on the optimistic sampling technique can be adapted to value iteration (which we leave for future work).

Table 1 summarizes the comparison of several regret upper bounds for both policy-based and value-based RL algorithms, in the tabular, linear MDP (LM) and linear kernel MDP (LKM) settings. The only exception is the LSVI-UCB algorithm (Wang et al., 2019), which works with the "optimistic closure (OC)" condition. For the linear MDP and linear kernel MDP settings, d denotes the number of features and T denotes the total number of steps. In the tabular setting, S and A denote the number of states and actions, respectively. *Policy-based* denotes whether the algorithm updates its policies directly using a policy improvement step, and *Exploration* denotes the type of exploration strategy used for each method. We emphasize that although both linear MDP and linear kernel MDP include tabular MDP as a special case, neither LM or LKM is special case of the other. Thus, the results for linear MDP and linear kernel MDP are not directly comparable. We mention results in LM for completeness.

C. Experiments



(a) Episode returns for best runs (b) Episode returns for all runs (c) Episode returns for different M

Figure 1: Comparison of RLSPO, OPPO, OPPO without bonus and POMD on the RiverSwim environment. a) Mean episode returns for best runs, after hyperparameter optimization. b) Mean episode returns, averaged over *all* hyperparameter combinations tested. c) Mean episode returns for different values of M. In all figures, the baseline is the optimal policy (going always right). The shaded areas represent one standard deviation. When well tuned to the task, OPPO can perform as well if not slightly better than RLSPO. However, RLSPO is much more robust to hyperparameter changes, as seen on the middle, making it significantly easier to tune.

To complement our theoretical results, we conduct numerical experiments in the standard variant of the RiverSwim environment (Strehl & Littman, 2008) and in randomly generated low-rank MDPs (Jin et al., 2020; Zanette et al., 2020a; Ayoub et al., 2020) to compare the performance of RLSPO with OPPO (Cai et al., 2019b) and POMD (Shani et al., 2020). Here we discuss the results in the RiverSwim environment and defer the results of the randomly generated MDPs to the Appendix I.

Note that, because RiverSwim is a stationary environment, there is no need to learn different policies at every time step, as is done in the general definition of linear kernel MDPs. To accommodate this relaxation, it suffices to modify Algorithm 1 and OPPO in order to have a single set of parameters θ^k and w^k and perform a single policy evaluation step using *all* previously encountered states, rather than one policy evaluation for each step h. Note that this modification also allows our algorithm to be used in variable-horizon settings.

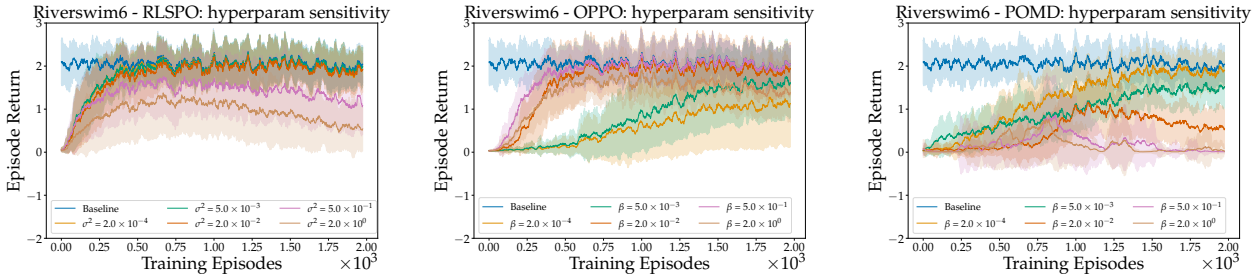
In Fig. 1a and Fig. 1b, we set $M = 25$ for RLSPO.[2] We then varied σ_1^2 for RLSPO (we fixed $\sigma_2^2 = d$) and the β constant for OPPO, which scales the exploration bonus. We also optimized the learning rate, α, which impacts both algorithms similarly. Finally, for all hyperparameter combinations, we evaluated the algorithms over 20 different seeds.

In Fig. 1, we directly compare the performance of RLSPO against OPPO and POMD. As can be seen in Fig. 1a, where we evaluate the optimized algorithms, all three reach a near-optimal policy after approximately 1000 episodes. Additionally, when finely tuned, OPPO can perform as well if not slightly better than RLSPO in the early phases of training. However, making OPPO and POMD reach that level of performance is difficult. Indeed, in Fig. 1b, we show the average performance of all three algorithms across all hyperparameter combinations we tried. Evidently, OPPO and POMD perform significantly worse than RLSPO on average, with a greater performance variance, thus showing its sensitivity to hyperparameter changes. By comparison, RLSPO exhibits a much more consistent behaviour, with the majority of combinations tested reaching near-optimal policies by the end of training. Fig 1c shows the performance of RLSPO when using different values of M.

In Fig. 2, we further emphasize that point by fixing the algorithm exploration parameter (σ_1^2 for RLSPO; β for OPPO and POMD) and optimizing the learning rate. In Fig. 2a, we see that if α is optimized, RLSPO will reach an optimal policy by the end of training in almost all cases, with only a small drop in performance for the two highest covariance values. Comparatively, in Fig. 2b and Fig. 2c, OPPO and POMD suffer much more from both high and low exploration bonuses, failing to get near the optimal policy for the highest and the two lowest β values. Note that, for both algorithms, we set the experiment such that there is factor of 10^4 between the lowest and the highest exploration parameter tested.

In summary, a fine-tuned OPPO can slightly outperform RLSPO, but the amount of fine tuning required to achieve similar performance for OPPO is typically much greater, leading to a heavier time and computational cost.

[2]Different M values were found to give very similar results.



(a) Episode returns for best runs (b) Episode returns for all runs (c) POMD episode for different β

Figure 2: Algorithm sensitivity to their main hyperparameter, tested in RiverSwim. a) Mean RLSPO episode returns for different σ_1^2 values. b) Mean OPPO episode returns for different β values. c) Mean POMD episode returns for different β values. In all three figures, the baseline is the optimal policy and the shaded areas represent one standard deviation. OPPO and POMD performances are significantly more sensitive to β than RLSPO is to σ_1^2, even if α was optimized individually for each run.

D. Definitions

In this section, we define the filtration generated by the history of the state-action sequence, observed rewards and the sampled noises. Then we define the values $\beta(\delta)$, $\nu(\delta)$, $\gamma(\delta)$ and $\alpha(\delta)$ which we use to provide our high probability confidence bounds. Furthermore, we specify the exact values of the posterior inflations σ_1 and σ_2 used in the Gaussian distributions in Line 1 and 1 of Algorithm 1 respectively. Recall that M is the number of noise samples we draw from each of the Gaussian distributions. We then introduce the good events which we will show to happen with high probability in Lemma 4.5 and under which the regret will be small. Finally, we define model prediction error which will simplify our discussion.

Definition D.1 (Filtrations)**.** We denote the σ-algbera generated by the set $\mathcal{G}$ using $\sigma(\mathcal{G})$. We define the following filtrations:

$$\begin{aligned}
\mathcal{F}^k &\stackrel{\text{def}}{=} \sigma\left(\{(s_t^i, a_t^i, r_t^i)\}_{\{i,t\}\in[k-1]\times[H]} \bigcup \{(\xi_t^{i,j}, \epsilon_t^{i,j})\}_{\{i,t,j\}\in[k-1]\times[H]\times[M]}\right), \\
\mathcal{F}_{h,1}^k &\stackrel{\text{def}}{=} \sigma\left(\mathcal{F}^k \bigcup \{(s_t^k, a_t^k, r_t^k)\}_{t\in[h]} \bigcup \{(\xi_t^{k,j}, \epsilon_t^{k,j}) : t \geq h,\ 1 \leq j \leq M\}\right), \\
\mathcal{F}_{h,2}^k &\stackrel{\text{def}}{=} \sigma\left(\mathcal{F}_{h,1}^k \bigcup \{x_{h+1}^k\}\right).
\end{aligned}$$

Definition D.2 (Noise bounds)**.** For any $\delta > 0$ and some large enough constants C_1, C_2 and C_3, let

$$\begin{aligned}
\sqrt{\beta(\delta)} &\stackrel{\text{def}}{=} C_1 H \sqrt{d \log(dT/\delta)}, \\
\sqrt{\nu(\delta)} &\stackrel{\text{def}}{=} \sqrt{\beta(\delta)} + \sqrt{d}, \\
\sqrt{\gamma(\delta)} &\stackrel{\text{def}}{=} C_2 \sqrt{d\nu(\delta) \log(d/\delta)}, \\
\sqrt{\alpha(\delta)} &\stackrel{\text{def}}{=} C_3 d \sqrt{\log(d/\delta)}.
\end{aligned}$$

Definition D.3 (Noise distribution)**.** In Algorithm 1, we set the following values for σ_1 and σ_2

$$\begin{aligned}
\sigma_1 &= \sqrt{\nu(\delta)}, \\
\sigma_2 &= \sqrt{d}.
\end{aligned}$$

Thus for all $j \in [M]$, we have,

$$\{\xi_h^{k,j}\} \sim \mathcal{N}\big(0, \nu(\delta)(\Sigma_h^k)^{-1}\big),$$

and

$$\{\epsilon_h^{k,j}\} \sim \mathcal{N}\big(0, d(\Lambda_h^k)^{-1}\big).$$

Definition D.4 (Good events)**.** For any $\delta > 0$ and positive integer M, we define the following random events

$$\mathcal{G}_h^k(\xi, \delta) \stackrel{\text{def}}{=} \Big\{ \max_{j \in [M]} \|\xi_h^{k,j}\|_{\Sigma_h^k} \leq \sqrt{\gamma(\delta)} \Big\},$$
$$\mathcal{G}_h^k(\epsilon, \delta) \stackrel{\text{def}}{=} \Big\{ \max_{j \in [M]} \|\epsilon_h^{k,j}\|_{\Lambda_h^k} \leq \sqrt{\alpha(\delta)} \Big\},$$
$$\mathcal{G}(K, H, \delta) \stackrel{\text{def}}{=} \bigcap_{k \leq K} \bigcap_{h \leq H} \big(\mathcal{G}_h^k(\xi, \delta) \cap \mathcal{G}_h^k(\epsilon, \delta) \big).$$

Definition D.5 (Model prediction error)**.** Finally, for all $(k, h) \in [K] \times [H]$, we recall the model prediction error associated with the reward r_h^k,

$$l_h^k(s, a) = r_h^k(s, a) + \mathbb{P}_h V_{h+1}^k(s, a) - Q_h^k(s, a).$$

This depicts the prediction error using V_{h+1}^k instead of $V_{h+1}^{\pi^k}$ in the Bellman equations (A.1).

E. Concentration Bounds

In this section we provide some key lemmas that are crucial for the regret analysis of Theorem 4.1. Our concentration results are conditioned on the good event $\mathcal{G}(K, H, \delta)$ which is defined in Definition D.4. Moreover, our high probability bounds involve functions $\alpha(\delta)$, $\beta(\delta)$, $\nu(\delta)$ and $\gamma(\delta)$ which are specified in Definition D.2.

E.1. Proof of Lemma 4.7

Lemma 4.7 characterizes the concentration behavior of the estimated value function. The proof closely follows that of Lemma D.1 in (Cai et al., 2019b). However, one major difference is that we consider the filtrations $\mathcal{F}_{h,1}^k$ and $\mathcal{F}_{h,2}^k$ defined in Appendix D.1 that include the noise sampled from Gaussian distributions as opposed to $\mathcal{F}_{k,h,1}$ and $\mathcal{F}_{k,h,2}$ defined in Section 4.1 in (Cai et al., 2019b). Before providing the proof for Lemma 4.7, we recall an important result on the concentration properties of self-normalizing processes.

Lemma E.1 (Concentration of Self-Normalized Processes (Abbasi-Yadkori et al., 2011))**.** Let $\{\varepsilon_t\}_{t=1}^{\infty}$ be a real-valued stochastic process with corresponding filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$. Let for any $t \geq 1$, $\varepsilon_t \,|\, \mathcal{F}_{t-1}$ be a zero-mean and σ- sub-Gaussian random variable, that is $\mathbb{E}[\varepsilon_t \,|\, \mathcal{F}_{t-1}] = 0$, and for all $\lambda \in \mathbb{R}$,

$$\mathbb{E}[\exp(\lambda \varepsilon_t) \,|\, \mathcal{F}_{t-1}] \leq \exp(\lambda^2 \sigma^2 / 2). \tag{E.1}$$

Let $\{\phi\}_{t=0}^{\infty}$ be an $\mathbb{R}^d$-valued stochastic process where $\phi_t \in \mathcal{F}_{t-1}$. Moreover, let $\Lambda_0 \in \mathbb{R}^{d \times d}$ be a positive definite matrix and let $\Lambda_t = \Lambda_0 + \sum_{s=1}^{t} \phi_s \phi_s^\top$. Then for any $\delta > 0$, with probability at least $1 - \delta$, it holds for all $t \geq 0$ that

$$\Big\| \sum_{s=1}^{t} \phi_s \varepsilon_s \Big\|_{\Lambda_t^{-1}}^2 \leq 2\sigma^2 \log\Big(\frac{\det(\Lambda_t)^{1/2} \det(\Lambda_0)^{-1/2}}{\delta} \Big). \tag{E.2}$$

Proof of Lemma 4.7. Using the Markov property and the definition of the filtration $\mathcal{F}_{h,1}^k$, we have

$$\mathbb{E}\left[V_{h+1}^k(s_{h+1}^k) \,|\, \mathcal{F}_{h,1}^k \right] = (\mathbb{P}_h V_{h+1}^k)(s_h^k, a_h^k). \tag{E.3}$$

Note that the function V_{h+1}^k is determined by the function Q_{h+1}^k and the policy π_{h+1}^k. Both Q_{h+1}^k and π_{h+1}^k are determined by the historical data in the filtration $\mathcal{F}_{h,1}^k$. Thus, conditioning on the filtration $\mathcal{F}_{h,1}^k$, V_{h+1}^k is a deterministic function and the only source of randomness in (E.3) is s_{h+1}^k.

We define the one-step environment noise with respect to V_{h+1}^k as

$$\eta_h^k = V_{h+1}^k(s_{h+1}^k) - (\mathbb{P}_h V_{h+1}^k)(s_h^k, a_h^k). \tag{E.4}$$

Note that conditioning on $\mathcal{F}_{h,1}^k$, η_h^k is a random variable with mean zero. Moreover, $V_{h+1}^k \in [0, H]$. Thus, using Hoeffding's lemma and the definition of sub-Gaussian random variable in (E.1), conditioning on the filtration $\mathcal{F}_{h,1}^k$, η_h^k is an $H/2$-sub-Gaussian random variable. Lastly, we note that η_h^k is $\mathcal{F}_{h,2}^k$-measurable. We are now ready to apply Lemma E.1. First, we

recall the definition of Σ_h^k.

$$\Sigma_h^k = \sum_{i=1}^{k-1} \phi_h^i(s_h^i, a_h^i)\phi_h^i(s_h^i, a_h^i)^\top + \lambda I. \tag{E.5}$$

Now using Lemma E.1, for any fixed $h \in [H]$, it holds that with probability at least $1 - \delta/H$,

$$\left\|\sum_{i=1}^{k-1} \phi_h^i(s_h^i, a_h^i) \cdot \left(V_{h+1}^i(s_{h+1}^i) - (\mathbb{P}_h V_{h+1}^i)(s_h^i, a_h^i)\right)\right\|^2_{(\Sigma_h^k)^{-1}} \leq H^2/2 \cdot \log\left(\frac{H \det(\Sigma_h^k)^{1/2} \det(\lambda I)^{-1/2}}{\delta}\right), \tag{E.6}$$

for any $k \in [K]$. We will now find an upper bound for $\det(\Sigma_h^k)$.

Using triangle inequality, we bound the spectral norm of Σ_h^k as

$$\begin{aligned}\|\Sigma_h^k\|_2 &\leq \lambda + \sum_{i=1}^{k-1} \|\phi_h^i(s_h^i, a_h^i)\|_2^2 \\ &\leq \lambda + dH^2K.\end{aligned}$$

The last inequality above follows from Assumption 2.1 which says for any $(s, a) \in \mathcal{S} \times \mathcal{A}$ and $V : \mathcal{S} \to [0, H]$,

$$\left\|\int_S \psi(s, a, s') \cdot V(s')ds'\right\|_2 \leq \sqrt{d}H.$$

Using the definition of spectral norm of a matrix, we deduce,

$$\begin{aligned}\det(\Sigma_h^k) &\leq \|\Sigma_h^k\|_2^d \\ &\leq (\lambda + dH^2K)^d.\end{aligned} \tag{E.7}$$

Finally, setting $\lambda = 1$, combining (E.6) and (E.7), and applying union bound we get that for any $h \in [H]$, with probability at least $1 - \delta$,

$$\begin{aligned}\left\|\sum_{i=1}^{k-1} \phi_h^i(s_h^i, a_h^i) \cdot \left(V_{h+1}^i(s_{h+1}^i) - (\mathbb{P}_h V_{h+1}^i)(s_h^i, a_h^i)\right)\right\|^2_{(\Sigma_h^k)^{-1}} &\leq H^2/2 \cdot \log\left(\frac{H(\lambda + dH^2K)^{d/2} \det(\lambda I)^{-1/2}}{\delta}\right) \\ &\leq C_1^2 dH^2 \log(dT/\delta),\end{aligned} \tag{E.8}$$

for any $(k, h) \in [K] \times [H]$, where $C_1 > 0$ is an absolute constant. Taking square root of both sides of (E.8) completes the proof.

□

E.2. Proof of Lemma 4.8

Proof. Applying Lemma E.2, for all $(s, a, h, k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]$, we have,

$$\left|r_h(s, a) + \mathbb{P}_h V_{h+1}^k(s, a) - \varphi(s, a)^\top \widehat{w}_h^k - \phi_h^k(s, a)^\top \widehat{\theta}_h^k\right| \leq \sqrt{d}\|\varphi(s, a)\|_{(\Lambda_h^k)^{-1}} + \sqrt{\nu(\delta)}\|\phi(s, a)_h^k\|_{(\Sigma_h^k)^{-1}}, \tag{E.9}$$

with probability at least $1 - \delta$.

As we are conditioning on the event $\mathcal{G}(K, H, \delta)$, using Cauchy-Schwarz inequality, we have

$$\begin{aligned}\max_{j\in[M]} \left|\phi_h^k(s, a)^\top \xi_h^{k,j}\right| &\leq \max_{j\in[M]} \left\|\phi_h^k(s, a)\right\|_{(\Sigma_h^k)^{-1}} \left\|\xi_h^{k,j}\right\|_{(\Sigma_h^k)^{-1}} \\ &\leq \sqrt{\gamma(\delta)}\left\|\phi_h^k(s, a)\right\|_{(\Sigma_h^k)^{-1}},\end{aligned} \tag{E.10}$$

for all $(s,a,h,k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]$.

Similarly, we have

$$\max_{j\in[M]} \left|\varphi(s,a)^\top \epsilon_h^{k,j}\right| \leq \sqrt{\alpha(\delta)} \left\|\varphi(s,a)\right\|_{(\Lambda_h^k)^{-1}}, \tag{E.11}$$

for all $(s,a,h,k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]$.

Now from the definition of model prediction error, using (E.9) and (E.10), we obtain,

$$\begin{aligned}
-l_h^k(s,a) &= Q_h^k(s,a) - r_h^k(s,a) - \mathbb{P}_h V_{h+1}^k(s,a) \\
&= \min\{\max_{j\in[M]} \widetilde{r}_h^{k,j}(s,a) + \max_{j\in[M]} \widetilde{P}_h \widetilde{V}_{h+1}^{k,j}(s,a), H-h+1\}^+ - r_h^k(s,a) - \mathbb{P}_h V_{h+1}^k(s,a) \\
&\leq \max_{j\in[M]} \widetilde{r}_h^{k,j}(s,a) + \max_{j\in[M]} \widetilde{P}_h \widetilde{V}_{h+1}^{k,j}(s,a) - r_h^k(s,a) - \mathbb{P}_h V_{h+1}^k(s,a) \\
&= \max_{j\in[M]} \varphi(s,a)^\top (\widehat{w}_h^k + \epsilon_h^{k,j}) + \max_{j\in[M]} \phi_h^k(s,a)^\top (\widehat{\theta}_h^k + \xi_h^{k,j}) - r_h^k(s,a) - \mathbb{P}_h V_{h+1}^k(s,a) \\
&\leq \max_{j\in[M]} \left|\varphi(s,a)^\top \epsilon_h^{k,j}\right| + \max_{j\in[M]} \left|\phi_h^k(s,a)^\top \xi_h^{k,j}\right| + \left|\varphi(s,a)^\top \widehat{w}_h^k + \phi_h^k(s,a)^\top \widehat{\theta}_h^k - r_h^k(s,a) - \mathbb{P}_h V_{h+1}^k(s,a)\right| \\
&\leq \sqrt{\alpha(\delta)} \left\|\varphi(s,a)\right\|_{(\Lambda_h^k)^{-1}} + \sqrt{\gamma(\delta)} \left\|\phi_h^k(s,a)\right\|_{(\Sigma_h^k)^{-1}} + \sqrt{d} \left\|\varphi(s,a)\right\|_{(\Lambda_h^k)^{-1}} + \sqrt{\nu(\delta)} \left\|\phi(s,a)_h^k\right\|_{(\Sigma_h^k)^{-1}} \\
&\leq \left(\sqrt{\alpha(\delta)} + \sqrt{d}\right) \left\|\varphi(s,a)\right\|_{(\Lambda_h^k)^{-1}} + \left(\sqrt{\gamma(\delta)} + \sqrt{\nu(\delta)}\right) \left\|\phi_h^k(s,a)\right\|_{(\Sigma_h^k)^{-1}},
\end{aligned} \tag{E.12}$$

and,

$$\begin{aligned}
l_h^k(s,a) &= r_h^k(s,a) + \mathbb{P}_h V_{h+1}^k(s,a) - Q_h^k(s,a) \\
&= r_h^k(s,a) + \mathbb{P}_h V_{h+1}^k(s,a) - \min\{\max_{j\in[M]} \widetilde{r}_h^{k,j}(s,a) + \max_{j\in[M]} \widetilde{P}_h \widetilde{V}_{h+1}^{k,j}(s,a), H-h+1\}^+ \\
&\leq \max\big\{r_h^k(s,a) + \mathbb{P}_h V_{h+1}^k(s,a) - \max_{j\in[M]} \widetilde{r}_h^{k,j}(s,a) - \max_{j\in[M]} \widetilde{P}_h \widetilde{V}_{h+1}^{k,j}(s,a), r_h^k(s,a) + \mathbb{P}_h V_{h+1}^k(s,a) \\
&\quad - (H-h+1)\big\}^+ \\
&\leq \max\big\{r_h^k(s,a) - \varphi(s,a)^\top \widehat{w}_h^k + \mathbb{P}_h V_{h+1}^k(s,a) - \phi_h^k(s,a)^\top \widehat{\theta}_h^k - \max_{j\in[M]} \varphi(s,a)^\top \epsilon_h^{k,j} - \max_{j\in[M]} \phi_h^k(s,a)^\top \xi_h^{k,j}, 0\big\} \\
&\leq \max\big\{\sqrt{d}\left\|\varphi(s,a)\right\|_{(\Lambda_h^k)^{-1}} - \max_{j\in[M]} \varphi(s,a)^\top \epsilon_h^{k,j} + \sqrt{\nu(\delta)} \|\phi_h^k(s,a)\|_{(\Sigma_h^k)^{-1}} - \max_{j\in[M]} \phi_h^k(s,a)^\top \xi_h^{k,j}, 0\big\},
\end{aligned} \tag{E.13}$$

with probability at least $1-\delta$.

For all $j \in [M]$, we have,

$$\{\xi_h^{k,j}\} \sim N\big(0, \nu(\delta)(\Sigma_h^k)^{-1}\big).$$

Thus, for all $j \in [M]$ and for all $(s,a) \in \mathcal{S} \times \mathcal{A}$, we have

$$\big\{\phi_h^k(s,a)^\top \xi_h^{k,j}\big\} \sim N\Big(0, \nu(\delta) \|\phi_h^k(s,a)\|^2_{(\Sigma_h^k)^{-1}}\Big).$$

Now, for any $j \in [M]$ and for all $(s,a) \in \mathcal{S} \times \mathcal{A}$, we have

$$\mathbb{P}\Big(\phi_h^k(s,a)^\top \xi_h^{k,j} - \sqrt{\nu(\delta)} \|\phi_h^k(s,a)\|_{(\Sigma_h^k)^{-1}} \geq 0\Big) = \Phi(-1),$$

where we denote the cumulative distribution function of the standard Gaussian by $\Phi(\cdot)$.

Now we have,

$$\begin{aligned}
\mathbb{P}\Big(\max_{j\in[M]} \phi_h^k(s,a)^\top \xi_h^{k,j} - \sqrt{\nu(\delta)} \|\phi_h^k(s,a)\|_{(\Sigma_h^k)^{-1}} \geq 0\Big) &= 1 - (1-\Phi(-1))^M \\
&= 1 - \Phi(1)^M \\
&= 1 - c_0^M,
\end{aligned} \tag{E.14}$$

where we set $c_0 = \Phi(1)$.

Likewise, for all $j \in [M]$, we have,

$$\{\epsilon_h^{k,j}\} \sim N\big(0, d(\Lambda_h^k)^{-1}\big).$$

Following the same steps as above, we have

$$\mathbb{P}\Big(\max_{j\in[M]} \varphi_h^k(s,a)^\top \epsilon_h^{k,j} - \sqrt{d}\|\varphi_h^k(s,a)\|_{(\Lambda_h^k)^{-1}} \geq 0\Big) \geq 1 - c_0^M. \tag{E.15}$$

Combining (E.13), (E.14) and (E.15), with probability at least $1-(\delta+2c_0^M)$, we have,

$$l_h^k(s,a) \leq 0. \tag{E.16}$$

Combining (E.12) and (E.16), for all $(s,a,h,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times[K]$, we get

$$l_h^k(s,a) \leq 0,$$

and

$$-l_h^k(s,a) \leq \Big(\sqrt{\alpha(\delta)} + \sqrt{d}\Big)\big\|\varphi(s,a)\big\|_{(\Lambda_h^k)^{-1}} + \Big(\sqrt{\nu(\delta)} + \sqrt{\gamma(\delta)}\Big)\big\|\phi_h^k(s,a)\big\|_{(\Sigma_h^k)^{-1}},$$

with probability at least $1-(\delta+2c_0^M)$. □

Lemma E.2. Let $\lambda = 1$ in Algorithm 1. For any $\delta > 0$, conditioned on the event $\mathcal{G}(K,H,\delta)$, for all $(s,a,h,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times[K]$, we have

$$\Big|\varphi(s,a)^\top \widehat{w}_h^k + \phi_h^k(s,a)^\top \widehat{\theta}_h^k - r_h(s,a) - \mathbb{P}_h V_{h+1}^k(s,a)\Big| \leq \sqrt{d}\big\|\varphi(s,a)\big\|_{(\Lambda_h^k)^{-1}} + \sqrt{\nu(\delta)}\big\|\phi_h^k(s,a)\big\|_{(\Sigma_h^k)^{-1}},$$

with probability $1-\delta$.

Proof. Using triangle inequality, we obtain

$$\begin{aligned}
&\Big|\varphi(s,a)^\top \widehat{w}_h^k + \phi_h^k(s,a)^\top \widehat{\theta}_h^k - r_h(s,a) - \mathbb{P}_h V_{h+1}^k(s,a)\Big| \\
&\quad\leq \underbrace{\big|\varphi(s,a)^\top \widehat{w}_h^k - r_h(s,a)\big|}_{\text{(i)}} + \underbrace{\big|\phi_h^k(s,a)^\top \widehat{\theta}_h^k - \mathbb{P}_h V_{h+1}^k(s,a)\big|}_{\text{(ii)}}.
\end{aligned} \tag{E.17}$$

In the following we will analyze term (i) and (ii) in (E.17) separately and derive an upper bound for each of them.

Term (i). Note that

$$\begin{aligned}
&\big|\varphi(s,a)^\top \widehat{w}_h^k - r_h(s,a)\big| \\
&= \big|\varphi(s,a)^\top \widehat{w}_h^k - \varphi(s,a)^\top w_h\big| \\
&= \Big|\varphi(s,a)^\top (\Lambda_h^k)^{-1}\Big(\sum_{i=1}^{k-1} r_h(s_h^i,a_h^i)\varphi(s_h^i,a_h^i) - \sum_{i=1}^{k-1}\varphi(s_h^i,a_h^i)\varphi(s_h^i,a_h^i)^\top w_h - \lambda w_h\Big)\Big| \\
&= \Big|\varphi(s,a)^\top (\Lambda_h^k)^{-1}\Big(\sum_{i=1}^{k-1} \varphi(s_h^i,a_h^i)\big(r_h(s_h^i,a_h^i) - \varphi(s_h^i,a_h^i)^\top w_h\big) - \lambda w_h\Big)\Big| \\
&= \Big|\varphi(s,a)^\top (\Lambda_h^k)^{-1}\Big(\sum_{i=1}^{k-1} \varphi(s_h^i,a_h^i)\big(\varphi(s_h^i,a_h^i)^\top w_h - \varphi(s_h^i,a_h^i)^\top w_h\big) - \lambda w_h\Big)\Big| \\
&= \lambda|\varphi(s,a)^\top (\Lambda_h^k)^{-1} w_h|,
\end{aligned} \tag{E.18}$$

where in the penultimate step, we used the fact $r_h(s,a) = \varphi(s,a)^\top w_h$ from Assumption 2.1. Applying Cauchy-Schwarz inequality we obtain,

$$\begin{aligned}\lambda|\varphi(s,a)^\top(\Lambda_h^k)^{-1}w_h| &\le \lambda\|\varphi(s,a)\|_{(\Lambda_h^k)^{-1}}\|w_h\|_{(\Lambda_h^k)^{-1}} \\ &\le \sqrt{\lambda}\|\varphi(s,a)\|_{(\Lambda_h^k)^{-1}}\|w_h\|_2 \\ &\le \sqrt{\lambda d}\|\varphi(s,a)\|_{(\Lambda_h^k)^{-1}}. \end{aligned} \tag{E.19}$$

Here the second inequality follows from observing that the smallest eigenvalue of Λ_h^k is at least λ and thus the largest eigenvalue of $(\Lambda_h^k)^{-1}$ is at most $1/\lambda$. The last inequality follows from Assumption 2.1 that $\|w_h\|_2 \le \sqrt{d}$. Combining (E.18) and (E.19) we get

$$\left|\varphi(s,a)^\top \widehat{w}_h^k - r_h(s,a)\right| \le \sqrt{\lambda d}\|\varphi(s,a)\|_{(\Sigma_h^k)^{-1}}. \tag{E.20}$$

Term (ii). Recall that

$$\phi_h^k(s,a) = \int_{\mathcal{S}} \psi(s,a,s')V_{h+1}^k(s')ds'.$$

Now,

$$\begin{aligned}\mathbb{P}_h V_{h+1}^k(s,a) &= \int_{\mathcal{S}} \psi(s,a,s')^\top \mu_h \cdot V_{h+1}^k(s')ds' \\ &= \phi_h^k(s,a)^\top \mu_h \\ &= \phi_h^k(s,a)^\top(\Sigma_h^k)^{-1}(\Sigma_h^k)\mu_h \\ &= \phi_h^k(s,a)^\top(\Sigma_h^k)^{-1}\Big(\sum_{i=1}^{k-1}\phi_h^i(s_h^i,a_h^i)\phi_h^i(s_h^i,a_h^i)^\top + \lambda I\Big)\mu_h \\ &= \phi_h^k(s,a)^\top(\Sigma_h^k)^{-1}\Big(\sum_{i=1}^{k-1}\phi_h^i(s_h^i,a_h^i)\phi_h^i(s_h^i,a_h^i)^\top\mu_h + \lambda\mu_h\Big) \\ &= \phi_h^k(s,a)^\top(\Sigma_h^k)^{-1}\Big(\sum_{i=1}^{k-1}\phi_h^i(s_h^i,a_h^i)\Big(\int_{\mathcal{S}}\psi(s_h^i,a_h^i,s')^\top V_{h+1}^i(s')ds'\Big)\mu_h + \lambda\mu_h\Big) \\ &= \phi_h^k(s,a)^\top(\Sigma_h^k)^{-1}\Big(\sum_{i=1}^{k-1}\phi_h^i(s_h^i,a_h^i)\Big(\int_{\mathcal{S}}\psi(s_h^i,a_h^i,s')^\top\mu_h \cdot V_{h+1}^i(s')ds'\Big) + \lambda\mu_h\Big) \\ &= \phi_h^k(s,a)^\top(\Sigma_h^k)^{-1}\Big(\sum_{i=1}^{k-1}\phi_h^i(s_h^i,a_h^i)(\mathbb{P}_h V_{h+1}^i)(s_h^i,a_h^i) + \lambda\mu_h\Big), \end{aligned} \tag{E.21}$$

where in the last step we invoke the definition of $\mathbb{P}_h$.

Since $(\Sigma_h^k)^{-1} \succ 0$, by Cauchy-Schwarz inequality and Lemma 4.7, with probability at least $1-\delta$, we have

$$
\begin{aligned}
&\big|\phi_h^k(s,a)^\top \widehat{\theta}_h^k - \mathbb{P}_h V_{h+1}^k(s,a)\big| \\
&= \Big|\phi_h^k(s,a)^\top (\Sigma_h^k)^{-1} \sum_{i=1}^{k-1} \phi_h^i(s_h^i,a_h^i)\cdot V_{h+1}^i \\
&\qquad - \phi_h^k(s,a)^\top (\Sigma_h^k)^{-1}\Big(\sum_{i=1}^{k-1} \phi_h^i(s_h^i,a_h^i)(\mathbb{P}_h V_{h+1}^i)(s_h^i,a_h^i) + \lambda\mu_h\Big)\Big| \\
&\le \Big|\phi_h^k(s,a)^\top (\Sigma_h^k)^{-1}\Big(\sum_{i=1}^{k-1} \phi_h^i(s_h^i,a_h^i)\cdot\big[\big(V_{h+1}^i - \mathbb{P}_h V_{h+1}^i\big)(s_h^i,a_h^i)\big]\Big)\Big| + \lambda\big|\phi_h^k(s,a)^\top (\Sigma_h^k)^{-1}\mu_h\big| \\
&\le \Big\|\sum_{i=1}^{k-1} \phi_h^i(s_h^i,a_h^i)\big[\big(V_{h+1}^k - \mathbb{P}_h V_{h+1}^k\big)(s_h^i,a_h^i)\big]\Big\|_{(\Sigma_h^k)^{-1}} \big\|\phi_h^k(s,a)\big\|_{(\Sigma_h^k)^{-1}} \\
&\qquad + \lambda\|\phi_h^k(s,a)\|_{(\Sigma_h^k)^{-1}}\|\mu_h\|_{(\Sigma_h^k)^{-1}} \\
&\le \big(\sqrt{\beta(\delta)} + \sqrt{\lambda d}\big)\big\|\phi_h^k(s,a)\big\|_{(\Sigma_h^k)^{-1}}. \qquad \text{(E.22)}
\end{aligned}
$$

Similar to (E.19), applying Cauchy-Schwarz inequality, we get

$$
\begin{aligned}
-\lambda\phi(s,a)^\top(\Sigma_h^k)^{-1}\langle\mu_h, V_{h+1}^k\rangle_{\mathcal{S}} &\le \lambda\|\phi(s,a)\|_{(\Sigma_h^k)^{-1}}\|\langle\mu_h, V_{h+1}^k\rangle_{\mathcal{S}}\|_{(\Sigma_h^k)^{-1}} \\
&\le \sqrt{\lambda}\|\phi(s,a)\|_{(\Sigma_h^k)^{-1}}\|\langle\mu_h, V_{h+1}^k\rangle_{\mathcal{S}}\|_2 \\
&\le \sqrt{\lambda}\|\phi(s,a)\|_{(\Sigma_h^k)^{-1}}\Big(\sum_{i=1}^{d}\|\mu_h^i\|_1^2\Big)^{\frac{1}{2}}\|V_{h+1}^k\|_\infty \\
&\le H\sqrt{\lambda d}\|\phi(s,a)\|_{(\Sigma_h^k)^{-1}}. \qquad \text{(E.23)}
\end{aligned}
$$

Here the second inequality follows using the same observation we did for **term (i)**. The last inequality follows from $\sum_{i=1}^d \|\mu_h^i\|_1^2 \le d$ in Assumption 2.1 and the clipping operation performed in Line 1 of Algorithm 1. Now combining (E.22), (E.20) and (E.23), and letting $\lambda = 1$, we get,

$$
\big|r_h(s,a) + \mathbb{P}_h V_{h+1}^k(s,a) - \varphi(s,a)^\top \widehat{w}_h^k - \phi_h^k(s,a)^\top\widehat{\theta}_h^k\big| \le \sqrt{d}\big\|\varphi(s,a)\big\|_{(\Lambda_h^k)^{-1}} + \sqrt{\nu(\delta)}\big\|\phi(s,a)_h^k\big\|_{(\Sigma_h^k)^{-1}},
$$

with probability $1-\delta$. □

E.3. Proof of Lemma 4.5

Proof. Recall from Definition D.4 that,

$$
\mathcal{G}(K,H,\delta') = \bigcap_{k\le K}\bigcap_{h\le H}\big(\mathcal{G}_h^k(\xi,\delta')\cap\mathcal{G}_h^k(\epsilon,\delta')\big).
$$

By Lemma 4.6, we have, for any fixed t and k, the event $\mathcal{G}_h^k(\xi,\delta')$ occurs with probability at least $1-M\delta'$. Similarly, for any fixed t and k, the event $\mathcal{G}_h^k(\epsilon,\delta')$ occurs with probability at least $1-M\delta'$.

Now taking union bound over all $(t,k)\in[H]\times[K]$, we have

$$
\mathbb{P}\left(\bigcap_{k\le K}\bigcap_{h\le H}\big(\mathcal{G}_h^k(\xi,\delta')\cap\mathcal{G}_h^k(\epsilon,\delta')\big)\right) \ge 1 - 2MT\delta' = 1-\delta,
$$

which completes the proof.

□

F. Regret Bound

In this section we prove the main regret bound for Algorithm 1.

F.1. Regret Decomposition

We begin our analysis with the following regret decomposition which is independent of the linear setting in Assumption 2.1.

Lemma F.1 (Lemma 4.2 in (Cai et al., 2019b))**.** It holds that

$$\begin{aligned}\text{Regret}(T) &= \sum_{k=1}^{K}\Big(V_1^*(s_1^k) - V_1^{\pi^k}(s_1^k)\Big)\\ &= \underbrace{\sum_{k=1}^{K}\sum_{t=1}^{H}\mathbb{E}_{\pi^*}\Big[\big\langle Q_h^k(s_h,\cdot), \pi_h^*(\cdot\,|\,s_h) - \pi_h^k(\cdot\,|\,s_h)\big\rangle \,\Big|\, s_1 = s_1^k\Big]}_{\text{(i)}}\\ &\quad + \underbrace{\sum_{k=1}^{K}\sum_{t=1}^{H}\mathcal{D}_h^k}_{\text{(ii)}} + \underbrace{\sum_{k=1}^{K}\sum_{t=1}^{H}\mathcal{M}_h^k}_{\text{(iii)}}\\ &\quad + \underbrace{\sum_{k=1}^{K}\sum_{h=1}^{H}\big(\mathbb{E}_{\pi^*}\big[l_h^k(s_h,a_h)\,|\,s_1 = s_1^k\big] - l_h^k(s_h^k,a_h^k)\big)}_{\text{(iv)}},\end{aligned} \tag{F.1}$$

where

$$\mathcal{D}_h^k := \langle (Q_h^k - Q_h^{\pi^k})(s_h^k,\cdot), \pi_h^k(\cdot, s_h^k)\rangle - (Q_h^k - Q_h^{\pi^k})(s_h^k,a_h^k), \tag{F.2}$$

$$\mathcal{M}_h^k := (\mathbb{P}_h(V_{h+1}^k - V_{h+1}^{\pi^k}))(s_h^k,a_h^k) - (V_{h+1}^k - V_{h+1}^{\pi^k})(s_h^k). \tag{F.3}$$

In the remaining parts of the section, we will show how to upper bound term (i), (ii), (iii) and (iv) in (F.1).

F.2. Regret Upper Bound

We will first upper bound term (i) in (F.1). Note that term (i) corresponds to the performance difference bound in Lemma H.1 with the Q-function replaced by the estimation Q_h^k, which is derived from the policy evaluation step in Algorithm 1.

Lemma F.2. For the policy π_h^k at time-step k of episode h, it holds that

$$\sum_{k=1}^{K}\sum_{t=1}^{H}\mathbb{E}_{\pi^*}\Big[\big\langle Q_h^k(s_h,\cdot), \pi_h^*(\cdot\,|\,s_h) - \pi_h^k(\cdot\,|\,s_h)\big\rangle \,\Big|\, s_1 = s_1^k\Big] \le \sqrt{2H^3T\log|\mathcal{A}|}, \tag{F.4}$$

where $T = HK$.

Proof. Using Lemma H.2, we obtain

$$\begin{aligned}
&\sum_{k=1}^{K}\sum_{t=1}^{H}\mathbb{E}_{\pi^*}\left[\langle Q_h^k(s_h,\cdot),\pi_h^*(\cdot\,|\,s_h)-\pi_h^k(\cdot\,|\,s_h)\rangle\,\middle|\,s_1=s_1^k\right]\\
&\quad\le\sum_{k=1}^{K}\sum_{t=1}^{H}\Bigg(\frac{\alpha H^2}{2}+\frac{1}{\alpha}\mathbb{E}_{\pi^*}\Big[D_{KL}(\pi_h^*(\cdot\,|\,s_h)\,\|\,\pi_h^k(\cdot\,|\,s_h))\\
&\quad- D_{KL}(\pi_h^*(\cdot\,|\,s_h)\,\|\,\pi_h^{k+1}(\cdot\,|\,s_h))\,\Big|\,s_1=s_1^k\Big]\Bigg)\\
&\quad\le\frac{\alpha H^3K}{2}+\frac{1}{\alpha}\sum_{h=1}^{H}\mathbb{E}_{\pi^*}\left[D_{KL}(\pi_h^*(\cdot\,|\,s_h)\,\|\,\pi_h^1(\cdot\,|\,s_h))\,\middle|\,s_1=s_1^k\right]\\
&\quad\le\frac{\alpha H^3K}{2}+\frac{1}{\alpha}H\log|A|,
\end{aligned}$$

where the second last inequality holds by the non-negativity of the KL-divergence, and the last inequality follows from the initial values of the Q-function and policy at the beginning of Algorithm 1. Specifically, setting $\alpha=\sqrt{2\log|\mathcal{A}|/H^2K}$ in Line (1) of Algorithm 1 completes the proof.

□

We now focus on upper bounding term (ii) and (iii) in (F.1), which are two sequences of martingales.

Lemma F.3 (Bound on Martingale Difference Sequence)**.** For any $\delta>0$, it holds with probability $1-2\delta/3$ that

$$\sum_{k=1}^{K}\sum_{t=1}^{H}\mathcal{D}_h^k+\sum_{k=1}^{K}\sum_{t=1}^{H}\mathcal{M}_h^k\le 2\sqrt{2H^2T\log(3/\delta)}. \tag{F.5}$$

Proof. Recall that

$$\begin{aligned}
\mathcal{D}_h^k&:=\langle(Q_h^k-Q_h^{\pi^k})(s_h^k,\cdot),\pi_h^k(\cdot,s_h^k)\rangle-(Q_h^k-Q_h^{\pi^k})(s_h^k,a_h^k),\\
\mathcal{M}_h^k&:=\mathbb{P}_h((V_{h+1}^k-V_{h+1}^{\pi^k}))(s_h^k,a_h^k)-(V_{h+1}^k-V_{h+1}^{\pi^k})(s_h^k).
\end{aligned}$$

Note that in line 1 of Algorithm 1, we truncate Q_h^k to the range $[0,H-h]$. Thus for any $(k,t)\in[K]\times[H]$, we have, $|\mathcal{D}_h^k|\le 2H$. Moreover, since $\mathbb{E}[\mathcal{D}_h^k|\mathcal{F}_{h,1}^k]=0$, $\mathcal{D}_h^k$ is a martingale difference sequence. So, applying Azuma-Hoeffding inequality we have with probability at least $1-\delta/3$,

$$\sum_{k=1}^{K}\sum_{t=1}^{H}\mathcal{D}_h^k\le\sqrt{2H^2T\log(3/\delta)}, \tag{F.6}$$

where $T=KH$.

Similarly, $\mathcal{M}_h^k$ is a martingale difference sequence since for any $(k,t)\in[K]\times[H]$, $|\mathcal{M}_h^k|\le 2H$ and $\mathbb{E}[\mathcal{M}_h^k|\mathcal{F}_{h,1}^k]=0$. Applying Azuma-Hoeffding inequality we have with probability at least $1-\delta/3$,

$$\sum_{k=1}^{K}\sum_{t=1}^{H}\mathcal{M}_h^k\le\sqrt{2H^2T\log(3/\delta)}. \tag{F.7}$$

Applying union bound on (F.6) and (F.7) gives (F.5) and completes the proof.

□

We now upper bound the term (iv) in (F.1).

Lemma F.4. Let $\lambda = 1$ in Algorithm 1. For any $\delta > 0$, conditioned on the event $\mathcal{G}(K, H, \delta)$, we have,

$$\sum_{k=1}^{K}\sum_{h=1}^{H}\left(\mathbb{E}_{\pi^*}\left[l_h^k(s_h, a_h) \,\middle|\, s_1 = s_1^k\right] - l_h^k(s_h^k, a_h^k)\right) \le \left(\sqrt{\alpha(\delta)} + \sqrt{\nu(\delta)} + \sqrt{\gamma(\delta)} + \sqrt{d}\right)\sqrt{2dHT\log(1+K)}, \quad \text{(F.8)}$$

with probability $1 - (\delta + 2c_0^M)$.

Proof. By Lemma 4.8, with probability $1 - (\delta + 2c_0^M)$ it holds that

$$\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{E}_{\pi^*}\left[l_h^k(s_h, a_h) \,\middle|\, s_1 = s_1^k\right] \le 0, \quad \text{(F.9)}$$

and

$$\begin{aligned}
\sum_{k=1}^{K}\sum_{h=1}^{H} -l_h^k(s_h^k, a_h^k) &\le \sum_{k=1}^{K}\sum_{h=1}^{H}\left(\sqrt{\alpha(\delta)} + \sqrt{d}\right)\left\|\varphi(s,a)\right\|_{(\Lambda_h^k)^{-1}} + \sum_{k=1}^{K}\sum_{h=1}^{H}\left(\sqrt{\nu(\delta)} + \sqrt{\gamma(\delta)}\right)\left\|\phi_h^k(s,a)\right\|_{(\Sigma_h^k)^{-1}} \\
&= \left(\sqrt{\alpha(\delta)} + \sqrt{d}\right)\sum_{k=1}^{K}\sum_{h=1}^{H}\left\|\varphi(s,a)\right\|_{(\Lambda_h^k)^{-1}} + \left(\sqrt{\nu(\delta)} + \sqrt{\gamma(\delta)}\right)\sum_{k=1}^{K}\sum_{h=1}^{H}\left\|\phi_h^k(s,a)\right\|_{(\Sigma_h^k)^{-1}} \\
&\le \left(\sqrt{\alpha(\delta)} + \sqrt{d}\right)\sum_{h=1}^{H}\sqrt{K}\left(\sum_{k=1}^{K}\left\|\varphi(s_h^k, a_h^k)\right\|^2_{(\Lambda_h^k)^{-1}}\right)^{1/2} \\
&\quad + \left(\sqrt{\nu(\delta)} + \sqrt{\gamma(\delta)}\right)\sum_{h=1}^{H}\sqrt{K}\left(\sum_{k=1}^{K}\left\|\phi_h^k(s_h^k, a_h^k)\right\|^2_{(\Sigma_h^k)^{-1}}\right)^{1/2} \\
&\le \left(\sqrt{\alpha(\delta)} + \sqrt{d}\right)H\sqrt{2dK\log(1+K)} + \left(\sqrt{\nu(\delta)} + \sqrt{\gamma(\delta)}\right)H\sqrt{2dK\log(1+K)} \\
&= \left(\sqrt{\alpha(\delta)} + \sqrt{\nu(\delta)} + \sqrt{\gamma(\delta)} + \sqrt{d}\right)H\sqrt{2dK\log(1+K)} \\
&= \left(\sqrt{\alpha(\delta)} + \sqrt{\nu(\delta)} + \sqrt{\gamma(\delta)} + \sqrt{d}\right)\sqrt{2dHT\log(1+K)}.
\end{aligned} \quad \text{(F.10)}$$

Here the second and the third inequality follow from the Cauchy-Schwarz inequality and Lemma H.4 respectively. Combining (F.9) and (F.10) completes the proof. □

F.3. Proof of Theorem 4.1

By conditioning our analysis on the good event as formalized in Lemma 4.5 and using concentration result for model prediction error from Lemma F.4, we are ready to prove our main theoretical result, Theorem 4.1.

Proof of Theorem 4.1. Let $\delta' = \delta/9$. From Lemma 4.5, the event $\mathcal{G}(K, H, \delta')$ happens with probability $1 - \delta'$. Combining Lemma F.4 and Lemma 4.5 we have that the event $\mathcal{G}(K, H, \delta')$ occurs and it holds that

$$\sum_{k=1}^{K}\sum_{h=1}^{H}\left(\mathbb{E}_{\pi^*}\left[l_h^k(s_h, a_h)|s_1 = s_1^k\right] - l_h^k(s_h^k, a_h^k)\right) \le \left(\sqrt{\alpha(\delta')} + \sqrt{\nu(\delta')} + \sqrt{\gamma(\delta')} + \sqrt{d}\right)\sqrt{2dHT\log(1+K)}, \quad \text{(F.11)}$$

with probability at least $(1-\delta')(1-(\delta'+2c_0^M))$. Note that $c_0^M = \delta'/2$ and $(1-\delta')(1-(\delta'+2c_0^M)) > 1-3\delta' = 1-\delta/3$. The martingale inequalities from Lemma F.3 happens with probability $1 - 2\delta/3$. Applying union bound on (F.4), (F.5) and (F.11) and using the fact that $\log|\mathcal{A}| = O(d^3[\log(dT/\delta)]^2)$, we get the final regret bound of $\widetilde{\mathcal{O}}(d^{3/2}H^{3/2}\sqrt{T})$. Therefore, we conclude the proof of Theorem 4.1. □

Remark F.5. When reduced to tabular setting, we just set $d = S^2A$. This directly leads to regret of $\widetilde{\mathcal{O}}(S^3A^{3/2}H^{3/2}\sqrt{T})$.

G. Bounds on Gaussian Noise

Here, we provide concentration inequalities for Gaussian random variables.

Lemma G.1 (Gaussian Concentration (Vershynin, 2018))**.** Consider a d-dimensional multivariate normal distribution $\eta \sim N(0, A\Sigma^{-1})$ where A is a scalar. For any $\delta > 0$, with probability $1 - \delta$,

$$\|\eta\|_\Sigma \leq c\sqrt{dA\log(d/\delta)},$$

where c is some absolute constant.

G.1. Proof of Lemma 4.6

Proof. From Lemma G.1, for a fixed $j \in [M]$, with probability at least $1 - \delta$ we would have

$$\|\eta\|_\Sigma \leq c\sqrt{dA\log(d/\delta)}.$$

Applying union bound over all M samples completes the proof. □

H. Auxiliary Lemmas

In this section we present several auxiliary lemmas.

H.1. Performance Difference Lemma and Proximal Descent Lemma

The following lemma shows how the difference between two policies can be stated in terms of the difference between their total rewards through the Q-function of one policy and the future state distribution of the other policy.

Lemma H.1 (Performance Difference Lemma; Lemma 3.2 in (Cai et al., 2019b))**.** Given two policies π and π', starting at state s_1, we have

$$V_1^{\pi'}(s_1) - V_1^{\pi}(s_1) = \mathbb{E}_{\pi'}\Big[\sum_{t=1}^{H}\big\langle Q_h^{\pi}(s_h,\cdot), \pi_h'(\cdot \mid s_h) - \pi_h(\cdot \mid s_h)\big\rangle \,\Big|\, s_1\Big].$$

As discussed in Section 3, for the policy improvement step in Algorithm 1, we solve an objective which is regularized by a KL-divergence term as in natural policy gradient (NPG) (Kakade, 2002) and our updated policy π^k takes the following closed form

$$\pi_h^k(\cdot \mid s) \propto \pi_h^{k-1}(\cdot \mid s) \cdot \exp\{\alpha Q_h^{k-1}(s,\cdot)\}.$$

We use the following lemma to characterize this closed form policy update.

Lemma H.2 (Proximal Descent Lemma; Lemma 3.3 in (Cai et al., 2019b))**.** For any distributions $p^*, p \in \Delta(\mathcal{A})$, state $s \in S$, and function $Q : \mathcal{S} \times \mathcal{A} \to [0, H]$, if $p' \in \Delta(\mathcal{A})$ is such that $p'(\cdot) \propto p(\cdot) \cdot \exp\{\alpha Q(s,\cdot)\}$, then

$$\big\langle Q(s,\cdot), p^*(\cdot) - p(\cdot)\big\rangle \leq \frac{\alpha H^2}{2} + \frac{1}{\alpha}\Big(D_{KL}\big(p^*(\cdot)\,\|\,p(\cdot)\big) - D_{KL}\big(p^*(\cdot)\,\|\,p'(\cdot)\big)\Big).$$

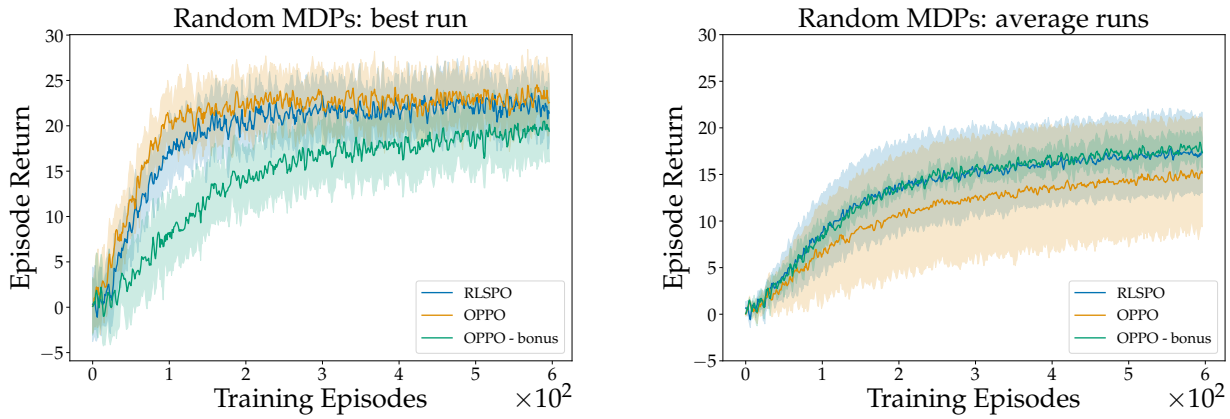
H.2. Some Useful Inequalities

Lemma H.3 (Hoeffding's Lemma (Vershynin, 2018))**.** Let X be any real-valued random variable with $\mathbb{E}[X] = \mu$, such that $a \leq X \leq b$ almost surely. Then, for all $\lambda \in \mathbb{R}$,

$$\mathbb{E}[\exp(\lambda X)] \leq \exp\left(\lambda\mu + \frac{\lambda^2(b-a)^2}{8}\right).$$

Lemma H.4 (Lemma 11 in (Abbasi-Yadkori et al., 2011))**.** Using the same notation as defined in this paper, we have

$$\sum_{k=1}^{K}\big\|\phi_h^k(s_h^k, a_h^k)\big\|^2_{(\Sigma_h^k)^{-1}} \leq 2d\log\Big(\frac{\lambda + K}{\lambda}\Big).$$



(a) Episode returns for best runs

(b) Episode returns for all runs

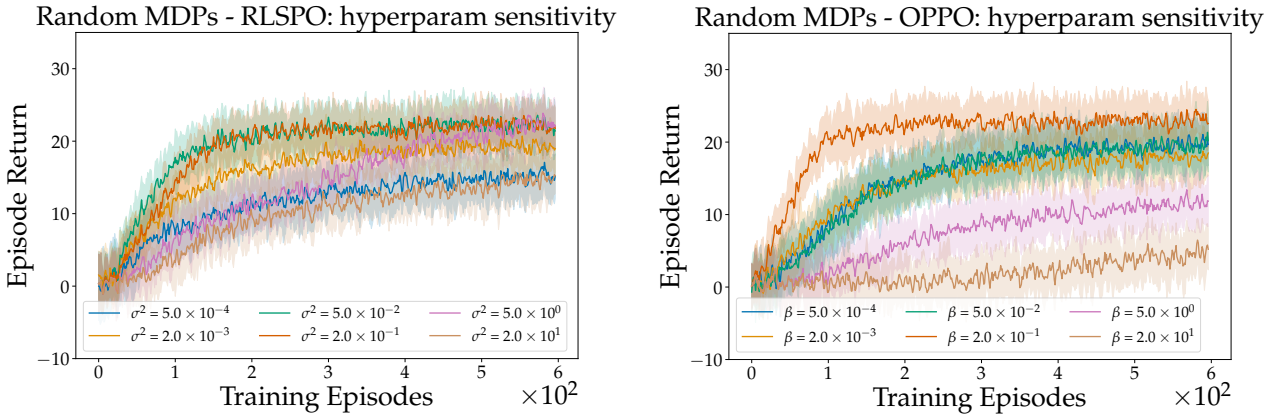
Figure 3: Comparison of RLSPO, OPPO and OPPO without bonus on the random MDP environment. a) Mean episode returns for best runs, after hyperparameter optimization. b) Mean episode returns, averaged over *all* hyperparameter combinations tested. The shaded areas represent one standard deviation. When well tuned to the task, OPPO can perform as well if not slightly better than RLSPO. However, RLSPO is much more robust to hyperparameter changes, as seen on the right, making it easier to tune.

I. Additional Details on Experiments

Our experimental results are demonstrated on two different environments, *RiverSwim* and *Random* MDP environments. Here, we provide additional details on the experimetnal setup and the results for both environments.

The RiverSwim Environment: Experimental results for the RiverSwim environment are provided in Fig. 1 and 2, where we compare the bandit feedback variant of OPPO (Cai et al., 2019a), OPPO without bonus, and our proposed algorithm RLSPO. Here, we briefly provide additional details describing the RiverSwim environment. The RiverSwim environment, as proposed in (Osband et al., 2013a) consists of six states as a chain MDP, where the agent starts from the leftmost state and chooses to swim left or right. This simulates an agent trying to swim against a water current to reach the land. The best policy is to swim right along the chain (against with the river current), while a bad policy would be to move left with the flow of current. Upon taking a right action, despite incurring a small negative reward, the environment stochastically transitions the agent to the right. The left action, although not incurring any immediate negative reward, almost surely transitions the agent to a left state. Maximal reward is obtained when the agent arrives at the right-end of the MDP. As introduced in (Osband et al., 2013a), such an environment requires efficient exploration to obtain an optimal policy, and is often considered a hard exploration task since the agent must figure out the optimal swimming action against the current flow, and the reward is given at the end of the chain. Agents that perform in this environment must demonstrate the ability to prioritize future potential rewards while not being locally optimal. Our experimental results show that the Thompson Sampling based RLSPO can solve the exploration task effectively while having higher robustness to hyperparameter tuning compared to the UCB based OPPO (Cai et al., 2019b).

Random MDP Experiments : We include additional results and ablation studies for randomly generated MDPs. For these sets of experiments, we use randomly generated non-stationary linearly parameterized MDPs with 10 states, 4 actions, an episode length of $H = 100$ and a sparse transition matrix. As a training setup, we use 4 random MDPs. For each MDP, we use 5 seeds for a total of 20 runs per hyperparameter combination. The experimental results in Fig. 3 compares our proposed RLSPO algorithm with OPPO (Cai et al., 2019a) (with and without bonus) across multiple runs. We further include our sensitivity analysis for the posterior inflation factor σ_1 for RLSPO and the bonus scaling factor β for OPPO as described in Fig. 4.



(a) RLSPO episode returns for different σ^2

(b) OPPO episode returns for different β

Figure 4: Algorithm sensitivity to their main hyperparameter, tested in random MDP. a) Mean RLSPO episode returns for different σ_1^2 values. b) Mean OPPO episode returns for different β values. In both figures, the baseline is the optimal policy and the shaded areas represent one standard deviation. OPPO performance is significantly more sensitive to β than RLSPO is to σ_1^2, even if α was optimized individually for each run.