Stochastic Lipschitz Q-Learning

Xu Zhu$^1$ and David Dunson$^2$

$^1$xz163@duke.edu, Department of Mathematics, Duke University
$^2$dunson@duke.edu, Departments of Statistical Science and Mathematics, Duke University

April 25, 2019

Abstract

In an episodic Markov Decision Process (MDP) problem, an online algorithm chooses from a set of actions in a sequence of $H$ trials, where $H$ is the episode length, in order to maximize the total payoff of the chosen actions. Q-learning, as the most popular model-free reinforcement learning (RL) algorithm, directly parameterizes and updates value functions without explicitly modeling the environment. Recently, [12] studies the sample complexity of Q-learning with finite states and actions. Their algorithm achieves nearly optimal regret, which shows that Q-learning can be made sample efficient. However, MDPs with large discrete states and actions [21] or continuous spaces [19] cannot learn efficiently in this way. Hence, it is critical to develop new algorithms to solve this dilemma with provable guarantee on the sample complexity. With this motivation, we propose a novel algorithm that works for MDPs with a more general setting, which has infinitely many states and actions and assumes that the payoff function and transition kernel are Lipschitz continuous. We also provide corresponding theory justification for our algorithm. It achieves the regret $\tilde{O}(K^{d+1}d^2\sqrt{H})$, where $K$ denotes the number of episodes and $d$ denotes the dimension of the joint space. To the best of our knowledge, this is the first analysis in the model-free setting whose established regret matches the lower bound up to a logarithmic factor.

1 Introduction

Reinforcement learning (RL) is about an agent interacting with the environment, learning an optimal policy by sequential trials to maximize cumulative rewards [25]. RL has a wide range of applications including health care [17], business management [16], artificial intelligence [21] etc. There are two main approaches to RL: model-based and model-free. Model-based algorithms [26][15][10][27][8] leverage a model representation for the environment and form a control policy based on the learned model. These approaches learn the value function and the policy in a data-efficient way, however, they may suffer from sensitivity to the model specification. Most state-of-the-art RL has been proposed in the model-free paradigm such as DQN [19], A3C [18], TRPO [20] etc. Model-free approaches directly update the value function and the policy, while allowing the dynamical system for the environment to be unknown. This robustness to model assumptions can come at the price of requiring a large number of samples, which may be costly or prohibitive to obtain for real physical systems [20][7]. Recent work has tried to improve the sample efficiency of model-free algorithms by combining them with model-based approaches. For example, [5] uses a model as the baseline.
while [9] uses roll-outs from the model as experience for acceleration. However, there is little theory to support such blending, which requires a precise quantitative understanding of relative sample complexities.

The theoretical question of “whether model-free algorithms can be made sample efficient” remained elusive until the very recent work [12]. In their paper, the authors consider episodic Markov Decision Process (MDP) dynamics, where the agent aims to maximize the total reward over multiple episodes. They leverage Q-learning with an UCB exploration policy that incorporates estimates of the confidence of Q values and exploration bonuses. The algorithm achieves total regret $\tilde{O}(\sqrt{H^3S\text{AT}})$, where $S$ is the number of states, $A$ is the number of actions, $H$ is the number of steps per episode and $T = HK$ is the total number of steps. However, MDPs with huge discrete states and actions [21] or continuous spaces [19] cannot learn efficiently in this way, though there is stronger demand for algorithms that can manage these tasks. Hence, it is critical to develop new algorithms to solve this dilemma with provable guarantee on the sample complexity. With this motivation, we propose a novel algorithm that works for MDPs within a more general setting, which deals with infinitely many states and actions, assuming that the payoff function and transition kernel are both Lipschitz continuous. We also provide corresponding theory justification for our algorithm. It achieves the regret bound of $\tilde{O}(K^{d+1}d^2 \sqrt{H^3})$, where $K$ denotes the number of episodes and $d$ denotes the dimension of the joint space. To the best of our knowledge, it is the first analysis in the model-free setting whose established regret matches the minimax lower bound up to a logarithmic factor.

2 Related Work

The definition of “model-free” is given in existing literature [24][25]:

**Definition 1.** A reinforcement learning algorithm is model-free if its space complexity is always sublinear relative to the space required to store an MDP.

- **Model-Free MDP** We do not assume access to a “simulator” and the agent is not allowed to reset within each episode. Under this setting, the standard Q-learning of incorporating $\epsilon$-greedy exploration appears to take exponentially many episodes to learn [13]. [24] introduces delayed Q-learning, where the Q-value for each state-action pair is updated only once every $m = \tilde{O}(1/\epsilon^2)$ times this pair is visited. When translated to this setting, this gives $\tilde{O}(T^{\frac{3}{2}})$ total regret. [12] proposes two algorithms, UCB-Hoeffding and UCB-Berstein. They achieve regret bounds of $O(\sqrt{H^3\text{SAT}})$ and $O(\sqrt{H^3\text{SAT}})$, respectively. The UCB exploration instead of $\epsilon$-greedy exploration in the model-free setting allows for better treatment of uncertainties for different states and actions.

- **Continuous Bandit** Bandits with infinitely many arms are practically significant, and it can be regarded as a special case of episodic MDP with $H = 1$. Model-based algorithms include linear payoff [1][2], Gaussian process payoff [6][23] etc. Another group of model-free algorithms assume that the expected payoff is a Lipschitz continuous function of the arms [4][14][22]. These algorithms achieve regret $\tilde{O}(T^{\frac{d+1}{d+2}})$, which matches the minimax lower bound, as a result of successfully managing the trade-off between exploration and exploitation, which motivates our study.
3 Preliminaries

We consider the setting of a tabular episodic Markov Decision Process (MDP) \((\mathcal{S}, \mathcal{A}, H, \mathbb{P}, M)\), where \(\mathcal{S}\) is the state space, \(\mathcal{A}\) is the action space, \(H\) is the number of steps in each episode, \(\mathbb{P}\) is the transition matrix so that \(\mathbb{P}_h(\cdot|s,a)\) gives the distribution over the next states if action \(a \in \mathcal{A}\) is taken for state \(s \in \mathcal{S}\), and \(M\) is the mapping from \(\mathcal{S} \times \mathcal{A}\) to the space of probability measures over the real line. We denote the distribution assigned to \((s,a)\) by \(M_{s,a}\). We require that for each \(x\), the distribution \(M_{s,a}\) is integrable and the mean reward function

\[ f(s,a) = \int y \, dM_{s,a}(y) \]

is measurable. In this paper, we focus on stochastic Lipschitz bandit optimization. First, the actual reward at step \(h\) for any state-action pair \((s,a)\) follows the distribution \(\mathcal{N}(f_h(s,a), \sigma^2)\), where \(f_h(\cdot, \cdot) \in [0,1]\) is the bounded mean-payoff satisfying the Lipschitz condition

\[ |f_h(s,a) - f_h(s',a')| \leq L_f(||s - s'||_\infty + ||a - a'||_\infty), \forall h \in [H] \]

and \(\sigma^2\) is a variance. The transition kernel is also Lipschitz continuous

\[ ||\mathbb{P}_h(\cdot|s,a) - \mathbb{P}_h(\cdot|s',a')||_{L_\infty} \leq L_P(||s - s'||_\infty + ||a - a'||_\infty), \forall h \in [H]. \]

Furthermore, the metric spaces \(\mathcal{S}\) and \(\mathcal{A}\) considered in this paper are hyper-rectangles with \(\mathcal{S} \subset \mathbb{R}^{d_S}\), and \(\mathcal{A} \subset \mathbb{R}^{d_A}\). We let \(d = d_S + d_A\) be the dimension of the joint space. Our theory generalizes to arbitrary compact spaces by embedding such spaces within a hyper-rectangle.

A policy \(\pi\) of an agent is a collection of \([H]\) functions \(\{\pi_h : \mathcal{S} \rightarrow \mathcal{A}\}_{h \in [H]}\). We use \(V^\pi_h : \mathcal{S} \rightarrow \mathbb{R}\) to denote the value function at step \(h\) under policy \(\pi\), so that \(V^\pi_h(s)\) gives the expected cumulative rewards received under policy \(\pi\), starting from \(s_h = s\). Formally,

\[ V^\pi_h(s) = \mathbb{E}_\pi\left[ \sum_{h'=h}^{H} f_h(s_{h'}, \pi_{h'}(a_{h'})) \mid s_h = s \right]. \]

Accordingly, we also define \(Q^\pi_h : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}\) to denote the Q-value function at step \(h\) under policy \(\pi\), which follows

\[ Q^\pi_h(s,a) = f_h(s,a) + \mathbb{E}_\pi\left[ \sum_{h'=h}^{H} f_h(s_{h'}, \pi_{h'}(a_{h'})) \mid s_h = s, a_h = a \right]. \]

Since the episode is finite, there always exists a policy \(\pi^*\) with \(V^*_h(s) = \sup_\pi V^\pi_h(s)\) for all \(s \in \mathcal{S}\) and \(h \in [H]\). For simplicity, we denote \(\mathbb{P}_h V^\pi_{h+1}(s,a) = \mathbb{E}_{s' \sim \mathbb{P}_h(\cdot|s,a)} V^\pi_{h+1}(s')\). The Bellman equation can be expressed as:

\[
\begin{align*}
V^\pi_h(s) &= Q^\pi_h(s, \pi_h(s)) \\
Q^\pi_h(s, a) &= f_h(s, a) + \mathbb{P}_h V^\pi_{h+1}(s, a) \\
V^\pi_{H+1}(s) &= 0 \quad \forall s \in \mathcal{S}
\end{align*}
\]

\[
\begin{align*}
V^*_h(s) &= \max_{a \in \mathcal{A}} Q^*_h(s, a) \\
Q^*_h(s, a) &= f_h(s, a) + \mathbb{P}_h V^*_{h+1}(s, a) \\
V^*_{H+1}(s) &= 0 \quad \forall s \in \mathcal{S}
\end{align*}
\]

With the previous assumptions and the Bellman optimality equation, it is not difficult to obtain the Lipschitz continuity of \(V^*_h\) and \(Q^*_h\).
Theorem 1. For any \( h \in [H], \) \( V_h^* \) and \( Q_h^* \) are Lipschitz continuous with respect to some coefficient \( L(H) \).

**Proof.** Since \( H \) is finite, it is sufficient to show that for each fixed \( h \), \( V_h^* \) and \( Q_h^* \) are Lipschitz continuous. Bellman equation for \( h = H \) gives

\[
Q_H^*(s, a) = f_H(s, a), \quad V_H^*(s) = \max_{a \in A} f_H(s, a).
\]

Hence \( Q_H^* \) and \( V_H^* \) are Lipschitz continuous. For \( h < H \),

\[
Q_h^*(s, a) = f_h(s, a) + \int_{s'} V_{h+1}^*(s') P_h(s'|s, a) \, ds'.
\]

Therefore,

\[
|Q_h^*(s_1, a_1) - Q_h^*(s_2, a_2)| \leq f_h(s_1, a_1) - f_h(s_2, a_2) + \int_{s' \in S} V_{h+1}^*(s') |P_h(s'|s_1, a_1) - P_h(s'|s_2, a_2)| \, ds'
\]

\[
\leq (L_f + L_p H m(S)) (\|s_1 - s_2\|_\infty + \|a_1 - a_2\|_\infty),
\]

where \( m(S) \) denotes the measure of space \( S \). Define \( L(H) = L_f + L_p H m(S) \) thus \( Q_h^* \) is Lipschitz continuous with respect to \( L(H) \). \( V_h^* \) satisfies

\[
V_h^*(s) = \max_{a \in A} Q_h^*(s, a),
\]

so it is also Lipschitz continuous with respect to \( L(H) \). \( \Box \)

We randomly initialize the state \( s_1^k \) for each episode \( k \) to let the agent play the game for \( K \) episodes. The performance of the policy \( \pi_k \) for \( k = 1, \ldots, K \) is measured by the total regret

\[
R_K = \sum_{k=1}^{K} [V_1^*(s_1^k) - V_{\pi_k}^*(s_1^k)].
\]

Our goal is to design a policy \( \pi \) that minimizes the regret bound. The following definition will be used to bound this regret.

**Definition 2 (Diameter).** Given the metric \( d(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \) over the space \( \mathcal{X} \), the diameter of \( \mathcal{X} \) is defined by

\[
D(\mathcal{X}) = \sup_{x, x' \in \mathcal{X}} d(x, x').
\]

In our setting, the metric \( d(\cdot, \cdot) \) is taken as \( \|x - x'\|_\infty \) to adapt to the Lipschitz continuity.
4 Main Results

In this section, we present our result on the total regret with partition-based UCB exploration. We also provide the minimax lower bound regret for our problem as a corollary of the lower bound regret for MDP with finite states and actions [12, 3]. This shows that our regret bound is tight up to a logarithmic factor.

Unlike the UCB exploration strategy for Q-learning [12] with only finite states and actions, which maintains the value function $Q_h(s, a)$ for each $(s, a, h) \in S \times A \times [H]$, it is intractable to estimate the value function for each state-action pair individually under our setting, since the number is discounted. Our strategy is to maintain $H$ separate partitions for $S \times A$ and then restrict the estimated $Q_h$ and $V_h$ to have the same value within the same partition. As we observe more feedback from interacting with the environment, the partition gets finer around regions with high payoff. At each time step $(h, k) \in [H] \times [K]$, given the state $s \in S$, the algorithm takes the action $a \in A$ that maximizes the current estimate $Q_h$ and observes immediate reward $y_h^k$ and the next state $s_{h+1}^k$. Ties are broken randomly in the specific partition that maximizes $Q_h$. We refer to this partition as the target partition. The Q-value is then updated in a partition-wise manner:

$$Q_h(s, a) \leftarrow (1 - \alpha_t)Q_h(s, a) + \alpha_t[y_h^k + V_{h+1}(s_{h+1}^k) + b_t], \quad \forall (s, a) \in P_h(s_h^k, a_h^k),$$

where $t$ is the counter of how many times $(s, a)$ is contained in target partitions (or, how many times that $Q_h(s, a)$ gets updated). As mentioned in [12], the learning rate $\alpha_t$ is chosen as

$$\alpha_t = \frac{H + 1}{H + t}$$

instead of $\frac{1}{t}$ in order to obtain regret that is not exponential in $H$. Moreover, we define $b_t$ at the $k$-th episode as:

$$b_t = \sqrt{\frac{AH^3 \log(4Hk^2/p)}{t}} + \sqrt{\frac{AH^3 \log(4Hk^2/p)}{t}} + \frac{2L(H)(D(S) + D(A))}{\sqrt{t}},$$

where $1 - p$ denotes our confidence, and $b_t$ is the amount added to the average payoff, where the first two terms account for the uncertainty arising from the randomness of the actual reward and the last term accounts for the variation of the mean-payoff function over the target partition.

Differently from [4], we gradually decrease the rate of splitting the target partition as its size gets smaller through a guided counter $L_h(s, a)$, which denotes the number of times that target partitions containing $(s, a)$ have been split. This adaptation is crucial to bound the partition size to control the regret. This also helps control the computational complexity if we store the value functions with a tree structure.

**Theorem 2.** For any $p > 0$, with probability $1 - p$, the total regret bound of Q-learning with partition-based UCB-Hoeffding (see Algorithm 1) is at most $\tilde{O}(K \frac{4H^3}{p} \sqrt{H^3})$.

To demonstrate the sharpness of our results, we also derive the theoretical lower bound for the episodic MDP studied in this paper.

**Theorem 3.** For fixed state space $S$, action space $A$, constants $L_r$, $L_p$ and arbitrarily large $K$, there exists an episodic MDP $(X_S, X_A, H, P, f)$ with $L_r$-Lipschitz reward function $f$ and $L_p$-Lipschitz transition probability $P$ such that $X_S \in S$, $X_A \in A$ and the total regret for $K$ episodes of any algorithm must be at least $\Omega(K \frac{4H^3}{p} \sqrt{H^3})$. 

5
In addition, we obtain:

\( \sum \) We let \( Q_h(s,a) \) and observe \( s_h \) for all \((s,a,h) \in S \times A \times [H]. \)

for episode \( k = 1, 2, \ldots \) do

for step \( h = 1, \ldots, H \) do

\( P \)

1. take action \( a_h^k \in \text{argmax}_a Q_h(s_h^k,a) \);

2. receive immediate reward \( y_h^k \) and observe \( s_{h+1}^k \);

3. \( n_h^k = N_h(s_h^k,a_h^k) + 1; \)

4. \( l_h^k = L_h(s_h^k,a_h^k) \);

5. \( Q_h(s,a) \leftarrow (1 - \alpha_n^k) Q_h(s,a) + \alpha_n^k [y_h^k + V_{h+1}(s_{h+1}^k) + b_{n_h}], \forall (s,a) \in \mathcal{P}_h(s_h^k,a_h^k) \);

6. \( V_h(s) \leftarrow \min \{ H, \max_{a \in A} Q_h(s_h^k,a) \} \);

7. \( N_h(s,a) \leftarrow N_h(s,a) + 1, \forall (s,a) \in \mathcal{P}_h(s_h^k,a_h^k) \);

8. if \( n_h \geq 4^{th} \) then

9. \( L_h(s,a) \leftarrow L_h(s,a) + 1, \forall (s,a) \in \mathcal{P}_h(s_h^k,a_h^k) \);

10. split \( \mathcal{P}_h(s_h^k,a_h^k) \) into \( 2^d \) sub-partitions \( \{ S_{h,i}^k \times A_{h,i}^k \}_{i=1}^{2^d} \) along the middle of each dimension;

11. \( \mathcal{P}_h(s,a) \leftarrow S_{h,i}^k \times A_{h,i}^k, \forall (s,a) \in S_{h,i}^k \times A_{h,i}^k , \text{ for } i = 1, \ldots, 2^d \); end end

end

end

end

5 Proofs of the Theorems

We let \( 1_A \) be the indicator function for event \( A \). We denote by \( Q_h^k, V_h^k, N_h^k \) respectively, the \( Q_h, V_h, N_h \) functions at the beginning of the \( k \)-th episode. We also denote by \( S_h^k \times A_h^k \) the target partition \( \mathcal{P}_h(s_h^k,a_h^k) \) of the \( k \)-th episode. Under these notations, the update equation at the \( k \)-th episode can be rewritten as follows, for every \( h \in [H] : \)

\[
Q_h^k(s,a) = \begin{cases} 
(1 - \alpha_n^k)Q_h^k(s,a) + \alpha_n^k [y_h^k + V_{h+1}(s_{h+1}^k) + b_{n_h}], & (s,a) \in S_h^k \times A_h^k \\
Q_h^k(s,a), & \text{otherwise}
\end{cases}
\]

Accordingly,

\[
V_h^k(s) = \min \{ H, \max_{a \in A} Q_h^k(s,a') \}, \forall s \in S_h^k.
\]

For notational convenience, we also introduce the following related quantities:

\[
\alpha_i^0 = \prod_{j=1}^{t} (1 - \alpha_j), \quad \alpha_i^t = \alpha_i \prod_{j=i+1}^{t} (1 - \alpha_j).
\]

It is easy to verify that (1) \( \sum_{i=1}^{t} \alpha_i^t = 1 \) and \( \alpha_i^0 = 0 \) for \( t \geq 1 \); (2) \( \sum_{i=1}^{t} \alpha_i^t = 0 \) and \( \alpha_i^0 = 1 \) for \( t = 0 \).

In addition, we obtain:

Lemma 1. The following properties hold for \( \alpha_i^t \):

6
\[
\frac{1}{\sqrt{t}} \leq \sum_{i=1}^{t} \frac{a_i^i}{\sqrt{t}} \leq \frac{2}{\sqrt{t}} \text{ for every } t \geq 1.
\]

- \[\max_{i \in [t]} a_i^i \leq \frac{2H}{t} \text{ and } \sum_{i=1}^{t} (a_i^i)^2 \leq \frac{2H}{t} \text{ for every } t \geq 1.\]
- \[\sum_{i=1}^{\infty} a_i^i = 1 + \frac{1}{t} \text{ for every } i \geq 1.\]

Refer to [12] for the proof of Lemma 1. At any \((s, a, h, k) \in S \times A \times [H] \times [K]\), let \(t = N_h^k(s, a)\) and denote the episodes \(k_1 < \ldots < k_t < k\) such that \((s, a) \in \mathcal{P}_h^{k_i}\) for every \(1 \leq i \leq t\). \(k_i\) is the episode at which \((s, a)\) is contained in the target partition for the \(i\)-th time. We have

\[
Q_h^k(s, a) = a_i^0H + \sum_{i=1}^{t} a_i^i[y_h^{k_i} + V_{h+1}^{k_i}(s_{h+1}^{k_i}) + b_i].
\]

From the Bellman optimality equation, \(Q_h^*(s, a) = f_h(s, a) + \mathbb{P}_h V_{h+1}(s, a)\) and the fact that \(\sum_{i=0}^{t} a_i^i = 1\), we have

\[
Q_h^*(s, a) = a_i^0Q_h^*(s, a) + \sum_{i=1}^{t} a_i^i[f_h(s, a) + \mathbb{P}_h V_{h+1}^*(s, a)].
\]

Next, we will show that \(Q_h^k(s, a)\) is an upper bound for \(Q_h^*(s, a)\) for arbitrary \((s, a, h, k)\). Furthermore, their difference can be bounded by the difference of \(V^{k_i}\) and \(V^*\) at the next step.

**Lemma 2.** Let \(\beta_t = 2 \sum_{i=1}^{t} a_i^i b_i \in [2b_t, 4b_t]\). With probability \(1 - 2\beta\), it holds for any \((s, a, h, k) \in S \times A \times [H] \times [K]\) that

\[
0 \leq Q_h^k(s, a) - Q_h^*(s, a) \leq a_i^0H + \sum_{i=1}^{t} a_i^i(V_{h+1}^{k_i}(s_{h+1}^{k_i}) - V_{h+1}^*(s_{h+1}^{k_i})) + \beta_t,
\]

where \(k_i = \min\{k' < k | k' > k_{i-1} \wedge (s, a) \in \mathcal{P}_h^{k'} \cup \{k\}\}\) with \(k_0 = 0\) and \(t = N_h^k(s, a)\).

**Proof.**

\[
Q_h^k(s, a) - Q_h^*(s, a) \\
\leq a_i^0H + \sum_{i=1}^{t} a_i^i[y_h^{k_i} + V_{h+1}^{k_i}(s_{h+1}^{k_i}) + b_i - f_h(s, a) - \mathbb{P}_h V_{h+1}^*(s, a)] \\
= a_i^0H + \sum_{i=1}^{t} a_i^i[y_h^{k_i} - f_h(s, a) + V_{h+1}^*(s_{h+1}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s, a)] \\
+ \sum_{i=1}^{t} a_i^i[V_{h+1}^{k_i}(s_{h+1}^{k_i}) - V_{h+1}^*(s_{h+1}^{k_i})] + \sum_{i=1}^{t} a_i^i b_i.
\]

Hence to prove the right hand side of Lemma 2, it is equivalent to show that with probability \(1 - 2\beta\),

\[
\sum_{i=1}^{t} a_i^i[y_h^{k_i} - f_h(s, a) + V_{h+1}^*(s_{h+1}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s, a)] \leq b_t \leq \frac{\beta_t}{2}.
\]
Using the Bellman optimality equation, we have
\[
\sum_{i=1}^{t} \alpha_i^i[V_{h+1}^*(s_{h+1}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s, a)]
\]
\[
= \sum_{i=1}^{t} \alpha_i^i[V_{h+1}^*(s_{h+1}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s_{h+1}^{k_i}, a_h^{k_i}) + \mathbb{P}_h V_{h+1}^*(s_{h+1}^{k_i}, a_h^{k_i}) - \mathbb{P}_h V_{h+1}^*(s, a)]
\]
\[
= \sum_{i=1}^{t} \alpha_i^i[V_{h+1}^*(s_{h+1}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s_{h+1}^{k_i}, a_h^{k_i}) + Q_h^*(s_{h+1}^{k_i}, a_h^{k_i}) - f_h(s_{h+1}^{k_i}, a_h^{k_i}) - Q_h^*(s, a) + f_h(s, a)].
\]
Therefore,
\[
\sum_{i=1}^{t} \alpha_i^i[y_h^{k_i} - f_h(s, a) + V_{h+1}^*(s_{h+1}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s, a)]
\]
\[
\leq \sum_{i=1}^{t} \alpha_i^i[V_{h+1}^*(s_{h+1}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s_{h+1}^{k_i}, a_h^{k_i})] + \sum_{i=1}^{t} \alpha_i^i[Q_h^*(s_{h+1}^{k_i}, a_h^{k_i}) - Q_h^*(s, a)]
\]
\[
+ \sum_{i=1}^{t} \alpha_i^i[f_h(s, a) - f_h(s_{h+1}^{k_i}, a_h^{k_i})].
\]

The three parts on the right hand side are bounded, respectively, in Lemma 3, Lemma 4 and Lemma 5. Equation 2 is thus proved. The left hand side of Lemma 2 follows from Equation 1 and Equation 2 and induction for $h$ from $H$ to 1.

\[\square\]

**Lemma 3.** For any $p \in (0, 1)$, with probability $1 - p$, it holds for any $(h, k) \in [H] \times [K]$ that
\[
\sum_{i=1}^{n_h^k} \alpha_i^i[V_{h+1}^*(s_{h+1}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s_{h+1}^{k_i}, a_h^{k_i})] \leq \sqrt{\frac{4H^3 \log(4Hk^2/p)}{n_h^k}},
\]
where $k_i = \min\{k' \leq k | k' > k_{i-1} \land (s_{h}^{k'}, a_h^{k'}) \in \mathcal{F}_h^{k'} \} \cup \{k\}$ with $k_0 = 0$.

**Proof.** Fixing $h \in H$, the random variable $k_i$ is clearly a stopping time. Let $\mathcal{F}_i$ be the $\sigma$-field generated by all the random variables $\{(s_{h}^{k'}, a_h^{k'}, y_{h}^{k'})\}_{1 \leq k \leq k_i}$ until episode $k_i$. Then $\{1_{k_i \leq k}[V_{h+1}^*(s_{h}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s_{h}^{k_i}, a_h^{k_i})]\}_{i=1}^{\tau}$ is a martingale difference sequence with respect to filtration $\{\mathcal{F}_i\}_{i \geq 0}$. By Azuma-Hoeffding inequality, we have that with probability at least $1 - \frac{p}{2Hk^2}$,
\[
\sum_{i=1}^{n_h^k} \alpha_i^i[V_{h+1}^*(s_{h+1}^{k_i}) - \mathbb{P}_h V_{h+1}^*(s_{h+1}^{k_i}, a_h^{k_i})] \leq H \left(-\sum_{i=1}^{n_h^k} 2(\alpha_i^i)^2 \log\left(\frac{p}{4Hk^2}\right)\right) \leq \sqrt{\frac{4H^3 \log(4Hk^2/p)}{n_h^k}}.
\]
Therefore, we apply union bound on $(h, k)$ and complete the proof of Lemma 3.

\[\square\]
Lemma 4. It holds for any \((h, k) \in [H] \times [K]\) that
\[
\sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i [Q_h^*(s_h^k, a_h^k) - Q_h^*(s, a)] \leq \frac{2L(H)(D(S) + D(A))}{\sqrt{n_h^k}},
\]
where \(k_i = \min\{k' \leq k | k' > k_{i-1} \land (s_h^k, a_h^k) \in \mathcal{P}_h^{k'} \} \cup \{k\}\) with \(k_0 = 0\).

Proof. It is sufficient to prove
\[
D(S_{h_i}^{k_i}) \leq \frac{D(S)}{\sqrt{i}} \quad \text{and} \quad D(A_{h_i}^{k_i}) \leq \frac{D(A)}{\sqrt{i}},
\]
and the result will directly follow from the fact that
\[
\sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \leq \frac{2}{\sqrt{n_h^k}}.
\]
Since the splitting is executed along the middle of each dimension, the diameters of the new partition elements are equal to half of the original one. Hence we have
\[
D(S_{h_i}^{k_i}) = \frac{D(S)}{2^{h_i}}.
\]
Since we split the target partition when \(n_h^k = 4^h\) thus it holds \(n_h^k \leq 4^h\). Therefore,
\[
D(S_{h_i}^{k_i}) = \frac{D(S)}{2^{h_i}} \leq \frac{D(S)}{\sqrt{n_h^k}} = \frac{D(S)}{\sqrt{i}}
\]
using the fact \(n_h^k = i\). Finally, it always holds that
\[
\sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i [Q_h^*(s_h^k, a_h^k) - Q_h^*(s, a)] = \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \frac{L(H)(D(S) + D(A))}{\sqrt{i}} \leq \frac{2L(H)(D(S) + D(A))}{\sqrt{n_h^k}}. \quad (3)
\]

Lemma 5. For any \(p \in (0, 1)\), with probability \(1 - p\), it holds for any \((h, k) \in [H] \times [K]\) that
\[
\left| \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i [y_h^{k_i} - f_h(s_h^k, a_h^k)] \right| \leq \sqrt{\frac{4H\sigma^2 \log(4Hk^2/p)}{n_h^k}},
\]
where \(k_i = \min\{k' \leq k | k' > k_{i-1} \land (s_h^k, a_h^k) \in \mathcal{P}_h^{k'} \} \cup \{k\}\) with \(k_0 = 0\).
Proof. According to our assumption, \( \{y_{hi} - f_h(s_{hi}, a_{hi})\}_{i \in [H]} \) \( \sim \text{N}(0, \sigma^2) \). Thus the weighted average \( \sum_{i=1}^{H} \alpha^i[y_{hi} - f_h(s_{hi}, a_{hi})] \sim \text{N}(0, \frac{1}{H} \sigma^2) \). By sub-Gaussian tail bound, we have with probability at least \( 1 - \frac{p}{H} \),
\[
\left| \sum_{i=1}^{H} \alpha^i[y_{hi} - f_h(s_{hi}, a_{hi})] \right| \leq \sqrt{\frac{4H \sigma^2 \log(4Hk^2/p)}{n_{hi}^k}}.
\]
Therefore, we apply union bound on \((h, k)\) and obtain Lemma 5.

**Lemma 6** (Bound the partition size). Denote by \( P_h^k \) the partition of \( S \times A \) for step \( h \) at the \( k \)-th episode, i.e. the range of \( P_h^k(s, a) \) at the beginning of the \( k \)-th episode. Denote by \( p_h^k \) the cardinality of \( P_h^k \). We have
\[
p_h^k = \mathcal{O}(k^{\frac{d}{a+2}})
\]
for any \( h \).

**Proof.** Fixing any \( h \in [H] \), every time Line 12 of Algorithm 1 is executed, the partition size will be increased by \( 2^d - 1 \). Without loss of generality, we denote \( \{k_i^h\}_{i=1}^{m_h^k} \) these episodes before \( k \), rearranged in the order such that \( l_{k_1^h} \leq l_{k_2^h} \leq \ldots \leq l_{k_{m_h^k}^h} \), where \( m_h^k = \Omega(p_h^k) \) is the quantity we want to bound. It follows that
\[
\sum_{i=1}^{m_h^k} (4^{l_{k_i^h}} - 4^{l_{k_i^h} - 1}) \leq k.
\]
The above formula relates \( m_h^k \) with \( l_{k_i^h} \) and \( k \). If we can further obtain a lower bound for \( l_{k_i^h} \), we will thereby get rid of \( l_{k_i^h} \) and bound \( m_h^k \) as a function of \( k \). Notice that the number of indices \( i \) satisfying
\[
l_{k_i^h} = l
\]
is at most \( 2^d \), it follows that
\[
l_{k_i^h} \geq \left\lceil \frac{\log_2(2^d - i + 1)}{d} \right\rceil - 1.
\]
Therefore,
\[
k = \Omega \left( \sum_{i=1}^{m_h^k} 4^{l_{k_i^h}} \right) = \Omega \left( \sum_{i=1}^{m_h^k} 4^{\log_2(2^d - i + 1)/d} \right) = \Omega \left( \sum_{i=1}^{m_h^k} i^{\frac{2}{d}} \right) = \Omega((m_h^k)^{\frac{d+2}{a}}).
\]
Rewriting the equation as \( m_h^k = \mathcal{O}(k^{\frac{d}{a+2}}) \) completes the proof.

**Proof of Theorem 2.** Let
\[
\delta_h^k = V_h^k(s_h^k) - V_{\pi_h^k}(s_h^k) \quad \text{and} \quad \phi_h^k = V_h^k(s_h^k) - V_h^*(s_h^k).
\]
By Lemma 2, it holds with probability $1 - 2p$ that $Q_h^k(s, a) \geq Q_h^k(s, a)$ for any $(s, a, h, k)$. Hence, we have
\[ V_1^k(s^k) = \max_{a \in A} Q_h^k(s^k, a) \geq \max_{a \in A} Q_1^k(s, a) = V_1^k(s^k) \]
where inequality (1) holds with probability $1 - 2p$. Thus, the total regret can be upper bounded:
\[ R_K = \sum_{k=1}^{K} [V_1^k(s^k) - V_1^{\pi_k}(s^k)] \leq \sum_{k=1}^{K} [V_1^k(s^k) - V_1^{\pi_k}(s^k)] = \delta_k. \]

The main idea of the rest of the proof is to upper bound $\sum_{k=1}^{K} \delta_h^k$ by the next step $\sum_{k=1}^{K} \delta_{h+1}^k$. For any fixed $(k, h) \in [K] \times [H]$, let $t = N_h^k(s^k_h, a^k_h)$ and denote by $k_1 < \ldots < k_t < k$ the episodes where $(s^k_h, a^k_h) \in S_h^k \times A_h^k$. Then we have,
\begin{align*}
\delta_h^k &= [V_h^k - V_h^{\pi_k}](s^k_h) \\
&\leq Q_h^k(s^k_h, a^k_h) - Q_h^{\pi_k}(s^k_h, a^k_h) \\
&= Q_h^k(s^k_h, a^k_h) - Q_h^*(s^k_h, a^k_h) + Q_h^*(s^k_h, a^k_h) + Q_h^{\pi_k}(s^k_h, a^k_h) \\
&\leq \alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i \phi_{h+1}^k + \beta_t + [\mathbb{P}_h(V_{h+1}^* - V_{h+1}^{\pi_k})](s^k_h, a^k_h) \\
&\leq \alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i \phi_{h+1}^k + \beta_t - \phi_{h+1}^k + \delta_{h+1}^k + \xi_{h+1}^k,
\end{align*}
where $\xi_{h+1}^k := [\mathbb{P}_h(V_{h+1}^* - V_{h+1}^{\pi_k})(s^k_h, a^k_h) - (V_{h+1}^*(s^k_h, a^k_h) - V_{h+1}^{\pi_k}(s^k_h, a^k_h))]$ is a martingale difference sequence. Inequality (1) holds because
\[ V_h^k(s^k_h) \leq \max_{a \in A} Q_h^k(s^k_h, a) = Q_h^k(s^k_h, a^k_h). \]

Inequality (2) holds by Lemma 2 and Bellman equation. Finally, equality (3) holds by definition
\[ \delta_{h+1}^k - \phi_{h+1}^k = V_{h+1}^*(s^k_{h+1}) - V_{h+1}^{\pi_k}(s^k_{h+1}). \] We turn to computing the summation $\sum_{k=1}^{K} \delta_h^k$. Notice that $t = 0$ only when $k = 1$, hence,
\[ \sum_{k=1}^{K} \alpha_t^0 H = \sum_{k=1}^{K} H 1_{t=0} = H. \]

The key step is to upper bound the second term which is
\[ \sum_{k=1}^{K} \sum_{i=1}^{n_h^k - 1} \alpha_t^i \phi_{h+1}^k(s^k_h, a^k_h), \]
where $k_i(s^k_h, a^k_h)$ is the episode in which $(s^k_h, a^k_h)$ is contained in the target partition at step $h$ for the $i$-th time. We regroup the summands in a different way. For every $k' \in [K]$, the term $\phi_{h+1}^{k'}$ first appears in the summand when $k = n_{h'}^k$. Therefore,
where inequality uses $\sum_{t=1}^{\infty} \alpha_t^i = 1 + \frac{1}{H}$. Plugging these back in we have

$$\sum_{k=1}^{K} \delta_h^k \leq H + (1 + \frac{1}{H}) \sum_{k=1}^{K} \phi_{h+1}^k - K \sum_{k=1}^{K} \phi_{h+1}^k + \sum_{k=1}^{K} \delta_{h+1}^k + \sum_{k=1}^{K} (\beta_{n_h^k} + \xi_{h+1}^k),$$

where the final inequality uses $\phi_{h+1}^k \leq \delta_{h+1}^k$ owing to fact that $V^* \geq V^\pi_h$. Recursing the result for $h = 1, \ldots, H$, and using the fact $\delta_{H+1}^k = 0$, we have

$$\sum_{k=1}^{K} \delta_1^k = \mathcal{O}(H^2 + \sum_{h=1}^{H} \sum_{k=1}^{K} (\beta_{n_h^k} + \xi_{h+1}^k)).$$

As a result of Lemma 6, for any step $h \in [H]$, the partition size before the $K$-th episode is $\mathcal{O}(K^{d^3 + \frac{3}{2}})$. Fixing $h$, we denote the partition at $K$-th episode as as $\{\mathcal{P}_i^K\}_{i=1}^{p_K}$. It follows that

$$\sum_{k=1}^{K} \beta_{n_h^k} \leq \mathcal{O}(1) \cdot \sum_{k=1}^{K} \sqrt{\frac{H^3 \log KH}{n_h^k}}$$

$$= \mathcal{O}(1) \cdot \sum_{i=1}^{p_K} \sum_{k: (s_h^k, a_h^k) \in \mathcal{P}_i^K} \sqrt{\frac{H^3 \log KH}{n_h^k}}$$

$$\leq \mathcal{O}(1) \sqrt{H^3 \log KH} \sum_{i=1}^{p_K} \sqrt{|\{k: a_h^k \in \mathcal{P}_i^K\}|}$$

$$\leq \mathcal{O}(1) \sqrt{H^3 \log KH} \sum_{i=1}^{p_K} \sqrt{p_K}$$

$$\leq \mathcal{O}(\sqrt{H^3 \log KH} \cdot K^{d^3 + \frac{3}{2}}),$$

where inequality ① is true because the $n_h^k$ values for $k$ such that $(s_h^k, a_h^k) \in \mathcal{S}_h^K \times \mathcal{A}_h^K$ are distinct and inequality ② follows from Cauchy inequality. Also, by the Azuma-Hoeffding inequality, with probability $1 - p$, we have:

$$\sum_{h=1}^{H} \sum_{k=1}^{K} \xi_{h+1}^k = \mathbb{P}_h(V_{h+1}^* - V_{h+1}^\pi_h)(s_h^k, a_h^k) - (V_{h+1}^* (s_h^k)) - V_{h+1}^\pi_h(s_h^k)) \leq \mathcal{O}(\sqrt{H^3 K \log KH}).$$
This establishes $\sum_{k=1}^{K} \delta_{k}^{r} \leq O(H^{2} + K^{\frac{d+1}{d+2}} \sqrt{H^{3} \log HK})$. When $K \geq H$, we have $O(H^{2}) \leq O(K^{\frac{d+1}{d+2}} \sqrt{H^{3} \log HK})$, hence $\sum_{k=1}^{K} \delta_{k}^{r} \leq O(K^{\frac{d+1}{d+2}} \sqrt{H^{3} \log HK})$. When $H \geq K$, we have $\sum_{k=1}^{K} \delta_{k}^{r} \leq HK \leq O(K^{\frac{d+1}{d+2}} \sqrt{H^{3} \log HK})$. Therefore, $\sum_{k=1}^{K} \delta_{k}^{r} = O(K^{\frac{d+1}{d+2}} \sqrt{H^{3} \log HK})$. In sum, we have $\sum_{k=1}^{K} \delta_{k}^{r} \leq O(K^{\frac{d+1}{d+2}} \sqrt{H^{3}})$ hold with probability at least $1 - 3p$. Rescaling $p$ to $p/3$ finishes the proof.

To prove the lower bound, we utilize the results for finite MDP as stated in [12]. The original construction of the composite MDP leveraged in their proof is given by [11].

**Lemma 7.** For any algorithm there exists an $H$-episodic MDP with $S$ states and $A$ actions such that for any $T$, the algorithm’s regret is $\Omega(H \sqrt{SAT})$.

The key is to construct $X_{S}$ and $X_{A}$ with proper cardinality and to show that with the MDP defined in [11] based upon $X_{S}$ and $X_{A}$, the associated $r$ and $P$ satisfy the Lipschitz condition.

**Proof of Theorem 3.** Let $r = cK^{-\frac{1}{d+2}}$ for some constant $c$. Recall that $S$ and $A$ are both hyper-rectangles; hence we can find an $r$-packing $X_{S}$ and $X_{A}$, respectively, for $S$ and $A$ with $|X_{S}| = \Theta(K^{\frac{d\epsilon}{\delta+2}})$ and $|X_{A}| = \Theta(K^{\frac{d\epsilon}{\delta+2}})$. Denote by $[a, b]$ the range of the first dimension of $S$. We further require that

$$\{x \in X_{S} : x_1 \in [a, a + \frac{b-a}{4}]\} = \{x \in X_{S} : x_1 \in [b - \frac{b-a}{4}, b]\} = \frac{|X_{S}|}{2}.$$ 

Hence $X_{S}$ is separated into two parts, denoted by $X_{S}^{0}$ and $X_{S}^{1}$, respectively, and the distance between these two parts is at least $\frac{b-a}{2}$, which is invariant to $K$. Assign deterministic reward to our MDP such that independent of the taken action, states in $X_{S}^{0}$ always obtain reward 0 while states in $X_{S}^{1}$ always obtain reward $R$. Clearly the reward function satisfies the $L_{f}$-Lipschitz continuity for $R = \frac{(b-a)L_{f}}{2}$. Formally, for any $(s, a), (s', a') \in X_{S} \times X_{A},$

$$f(s, a) - f(s', a') \leq L_{f}\|s - s'\|_{\infty}.$$ 

We now compress the state space so that it only contains two elements $s_{0}$ and $s_{1}$. In addition, the action space is copied $\Theta(|X_{S}|)$ times, with $A'$ denoting the new action space. The compression does not change the optimal average reward[11]. The transition matrix is defined as $P(s_{1} | s_{0}, a^{*}) = \delta$ for some “good” action $a^{*}$ while for any other state-action pair, the state switches with probability $\delta$ while remaining the same with probability $1 - \delta$. For any algorithm, the optimal regret obtains the maximum if we choose $\epsilon = O(\sqrt{|X_{S}|}) = (K^{-\frac{1}{d+2}}) = O(r)$ as proved in [11]. The $L_{p}$-Lipschitz continuity of $P$ is thus satisfied for some proper $c$.

We have translated the MDP constructed in the proof of Lemma 7 into our setting. Therefore, the total regret for any algorithm on $(X_{S}, A, H, P, f)$ is at least $\Omega(H \sqrt{|X_{S}||X_{A}|T}) = O(K^{\frac{d+1}{d+2}} \sqrt{H^{3}})$. 

13
References

[1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In Advances in Neural Information Processing Systems, pages 2312–2320, 2011.

[2] Marc Abeille, Alessandro Lazaric, et al. Linear Thompson sampling revisited. Electronic Journal of Statistics, 11(2):5165–5197, 2017.

[3] Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pages 263–272. JMLR. org, 2017.

[4] Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. X-armed bandits. Journal of Machine Learning Research, 12(May):1655–1695, 2011.

[5] Yevgen Chebotar, Karol Hausman, Marvin Zhang, Gaurav Sukhatme, Stefan Schaal, and Sergey Levine. Combining model-based and model-free updates for trajectory-centric reinforcement learning. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pages 703–711. JMLR. org, 2017.

[6] Nando De Freitas, Alex J Smola, and Masrour Zoghi. Exponential regret bounds for Gaussian process bandits with deterministic observations. In Proceedings of the 29th International Coference on Machine Learning, pages 955–962. Omnipress, 2012.

[7] Marc Deisenroth and Carl E Rasmussen. Pilco: A model-based and data-efficient approach to policy search. In Proceedings of the 28th International Conference on machine learning (ICML-11), pages 465–472, 2011.

[8] Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In 2017 IEEE International Conference on Robotics and Automation (ICRA), pages 2786–2793. IEEE, 2017.

[9] Shixiang Gu, Ethan Holly, Timothy Lillicrap, and Sergey Levine. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In 2017 IEEE International Conference on Robotics and Automation (ICRA), pages 3389–3396. IEEE, 2017.

[10] Nicolas Heess, Gregory Wayne, David Silver, Timothy Lillicrap, Tom Erez, and Yuval Tassa. Learning continuous control policies by stochastic value gradients. In Advances in Neural Information Processing Systems, pages 2944–2952, 2015.

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

[12] Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is q-learning provably efficient? In Advances in Neural Information Processing Systems, pages 4868–4878, 2018.

[13] Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. Machine Learning, 49(2-3):209–232, 2002.
[14] Robert Kleinberg, Aleksandrs Slivkins, and Eli Upfal. Multi-armed bandits in metric spaces. In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, pages 681–690. ACM, 2008.

[15] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

[16] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, pages 661–670. ACM, 2010.

[17] Riccardo Miotto, Fei Wang, Shuang Wang, Xiaqian Jiang, and Joel T Dudley. Deep learning for healthcare: review, opportunities and challenges. *Briefings in Bioinformatics*, 19(6):1236–1246, 2017.

[18] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning*, pages 1928–1937, 2016.

[19] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[20] John Schulman, Sergey Levine, Pieter Abbeel, Michael I Jordan, and Philipp Moritz. Trust region policy optimization. In *ICML*, volume 37, pages 1889–1897, 2015.

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

[22] Aleksandrs Slivkins. Contextual bandits with similarity information. *The Journal of Machine Learning Research*, 15(1):2533–2568, 2014.

[23] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

[24] Alexander L Strehl, Lihong Li, Eric Wiewiora, John Langford, and Michael L Littman. Pac model-free reinforcement learning. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 881–888. ACM, 2006.

[25] Richard S Sutton, Andrew G Barto, et al. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

[26] Yuval Tassa, Tom Erez, and Emanuel Todorov. Synthesis and stabilization of complex behaviors through online trajectory optimization. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4906–4913. IEEE, 2012.
[27] Manuel Watter, Jost Springenberg, Joschka Boedecker, and Martin Riedmiller. Embed to control: A locally linear latent dynamics model for control from raw images. In Advances in Neural Information Processing Systems, pages 2746–2754, 2015.