Overcoming the Long Horizon Barrier for Sample-Efficient Reinforcement Learning with Latent Low-Rank Structure

Tyler Sam[1], Yudong Chen[2], and Christina Lee Yu[1]

[1]School of Operations Research and Information Engineering, Cornell University
[2]Department of Computer Sciences, University of Wisconsin-Madison

June 27, 2023

Abstract

The practicality of reinforcement learning algorithms has been limited due to poor scaling with respect to the problem size, as the sample complexity of learning an ϵ-optimal policy is $\tilde{\Omega}\left(|S||A|H^3/\epsilon^2\right)$ over worst case instances of an MDP with state space S, action space A, and horizon H. We consider a class of MDPs for which the associated optimal Q^* function is low rank, where the latent features are unknown. While one would hope to achieve linear sample complexity in $|S|$ and $|A|$ due to the low rank structure, we show that without imposing further assumptions beyond low rank of Q^*, if one is constrained to estimate the Q function using only observations from a subset of entries, there is a worst case instance in which one must incur a sample complexity exponential in the horizon H to learn a near optimal policy. We subsequently show that under stronger low rank structural assumptions, given access to a generative model, Low Rank Monte Carlo Policy Iteration (LR-MCPI) and Low Rank Empirical Value Iteration (LR-EVI) achieve the desired sample complexity of $\tilde{O}\left((|S|+|A|)\text{poly}(d,H)/\epsilon^2\right)$ for a rank d setting, which is minimax optimal with respect to the scaling of $|S|, |A|$, and ϵ. In contrast to literature on linear and low-rank MDPs, we do not require a known feature mapping, our algorithm is computationally simple, and our results hold for long time horizons. Our results provide insights on the minimal low-rank structural assumptions required on the MDP with respect to the transition kernel versus the optimal action-value function.

Contents

1 Introduction

Reinforcement learning (RL) methods have been increasingly popular in sequential decision making tasks due to their empirical success, e.g., Atari Games [31], StarCraft II [46], and robotics [27]. RL algorithms can be applied to any sequential decision making problem which can be modeled by a Markov decision process (MDP) defined over a state space S and an action space A. The agent interacts with the environment across a horizon of length H. In each step of the horizon, the agent observes the current state of the environment and takes an action. In response the environment returns an instantaneous reward and transitions to the next state. The key Markov property that the dynamics of an MDP must satisfy is that the distribution of the instantaneous reward and the next state is only a function of the current state and action. As a result it is sufficient for the agent to only consider policies that define a distribution over the actions given the current state of the environment. The goal of the agent is to find an optimal policy which maximizes its cumulative expected reward over the horizon. When the dynamics and reward function of the MDP are known in advance, that can be solved directly using dynamic programming. Reinforcement learning considers the setting in which the MDP dynamics are unknown and thus the algorithm must query from the MDP to both learn the model as well as find an optimal policy.

Despite the empirical success and popularity of RL, its usage in practical applications is limited by the high data sampling costs in the training process, resulting from poor scaling of RL algorithms with respect to the size of the state and action spaces. Given a finite-horizon homogeneous MDP with state space S, action space A, and horizon H, one needs $\tilde{\Omega}\left(|S||A|H^3/\epsilon^2\right)$ samples given a generative model to learn an optimal policy [38]. The required number of samples is often too large as many real-world problems when modeled as a Markov decision process (MDP) have very large state and action spaces. For example, the n-city Capacitated Vehicle Routing Problem (CVRP), a classical combinatorial problem from operations research, involves a state space $\{0,1\}^n$ and an action space being all partial permutations of $n-1$ cities [13].

A key function that is used in the course of solving for an optimal policy is the Q^π function, which is also referred to as the action-value function of policy π. It is defined over steps $h \in [H]$, states $s \in S$, and actions $a \in A$. $Q_h^\pi(s,a)$ represents the expected cumulative reward that an agent would collect if it were at state s at step h, took action a, and subsequently followed the policy π for all future steps until the end of the horizon. When the state and action space are finite, the Q_h^π function can be represented as a $|S| \times |A|$ matrix. The Q function associated to the optimal policy is denoted by Q^*. Given Q_h^*, the optimal policy at step h is trivial to find as it would follow from simply choosing the action that optimizes Q_h^* for each state. Many RL algorithms rely on estimating the Q_h^* functions across all state action pairs in order to find a near optimal policy, resulting in the $|S||A|$ sample complexity dependence. Furthermore, the tight lower bound also suggests one may need to estimate the full Q_h^* function to find the optimal policy in worst case MDPs.

MDPs with Low Rank Structures. A glaring limitation of general purpose RL algorithms is that they do not exploit application dependent structure that may be known in advance. Many real-world systems in fact have additional structure that if exploited should improve computational and statistical efficiency. The critical question becomes what structure is reasonable to assume, and how to design new algorithms and analyses to efficiently exploit it. In this work, we focus on the subclass of MDPs that exhibit latent low-dimensional structure with respect to the relationship between states and actions, e.g., Q_h^* is low rank when viewed as a $|S|$-by-$|A|$ matrix. A sufficient but not necessary condition that would result in such a property is that the transition kernel has low Tucker rank when viewed as a $|S|$-by-$|S|$-by-$|A|$ tensor of state transition probabilities, and the

expected instantaneous reward function can be represented as a $|S|$-by-$|A|$ low rank matrix.

While low rank structure has been extensively used in the matrix and tensor estimation literature, it has not been widely studied in the RL literature, except for the theoretical results from [37] and empirical results from [56, 34, 33]. However, we will give examples at the end of Section 5 to illustrate that this property is in fact quite widespread and common in many real world systems. While the sample complexity under the fully general model scales as $|S||A|$, we would expect that the sample complexity under a rank-d model would scale as $d(|S|+|A|)$, as the low rank assumption on a matrix reduces the degrees of freedom of Q_h^* from $|S||A|$ to $d(|S|+|A|)$. Even though this intuition holds true in the classical low rank matrix estimation setting, the additional dynamics of the MDP introduce complex dependencies that may amplify the error for long horizons. The work in [37] proposes an algorithm that learns an ϵ-optimal Q function under a low rank assumption on Q_h^*, resulting in a sample complexity of $\tilde{O}(\text{poly}(d)(|S|+|A|)\exp(H)/\epsilon^2)$ in the general finite-horizon MDP setting. While they do achieve the reduction from $|S||A|$ to $\text{poly}(d)(|S|+|A|)$, they have an exponential dependence on the horizon that arises from an amplification of the estimation error due to the MDP dynamics and nonlinearity of low rank matrix estimation. A key contribution of this work is to characterize conditions under which we are able to achieve both linear sample complexity on $|S|$ and $|A|$ along with polynomial dependence on H.

The term "low rank" has been used to describe other types of low dimensional models in the MDP/RL literature, especially in the context of linear function approximation, and we would like to clarify up front that these models are significantly different. In particular, the typical use of "low rank MDPs" refers to an assumption that the transition kernel when viewed as a tensor is low rank with respect to the relationship between the originating state action pair (s, a) and the destination state s'. This implies that the relationship across time for a given trajectory exhibits a latent low dimensional structure in that the relationship between the future state and the previous state and action pair is mediated through low dimensional dynamics. However, this assumption does not imply that the Q function is low rank when viewed as a matrix, which would imply a low dimensional relationship between the current state and the action taken at that state. Another assumption which is easily confused with ours is the assumption that Q^* is linearly-realizable. This implies that Q^* can be written as a linear combination of d matrices $\{\phi_\ell\}_{\ell\in[d]}$ (each of size $|S|$ by $|A|$). While this implies that the set of plausible Q^* lives in a low dimensional space parameterized by $\{\phi_\ell\}_{\ell\in[d]}$, this does not imply that Q^* is low rank with respect to the relationship between S and A. The guarantees for RL algorithms under low rank MDP and linearly-realizable Q^* structure either require prior knowledge of the feature representation as given by $\{\phi_\ell\}_{\ell\in[d]}$, or otherwise do not admit polynomial time algorithms. While assuming a priori knowledge of the feature representation is often restrictive and unlikely in real applications, this assumption enables a reduction to supervised learning such that the sample complexity no longer depends on the size of the state and action space, but only on the dimension of the representation. The low rank structure we assume in this work does not require any knowledge of the latent low dimensional representation, but as a result the optimal sample complexity necessarily must still scale linearly with the size of the state and action space.

Our Contributions. We identify sufficient low-rank structural assumptions that allow for computationally and statistically efficient learning, reducing the sample complexity bounds to scale only linearly in $|S|, |A|$ and *polynomially* in H (as opposed to exponential in H in [37] or $|S||A|$ in the general tabular MDP setting). First, we show that there are additional complexities that arise from MDPs with *long horizons*; we provide an example where the optimal action-value function Q^* is low rank, yet the learner must observe an exponential (in H) number of samples to learn a near optimal policy when exploiting the low-rank structure of Q^*. This lower bound illustrates that

exploiting low rank structure in RL is significantly more involved than classical matrix estimation. We propose a new computationally simple model-free algorithm, referred to as Low Rank Monte Carlo Policy Iteration (LR-MCPI). Under the assumption that Q^* is low rank, by additionally assuming a constant suboptimality gap, we prove that LR-MCPI achieves the desired sample complexity, avoiding the exponential error amplification in the horizon. Additionally we prove that LR-MCPI also achieves the desired sample complexity when all ϵ-optimal policies π have low rank Q^π functions. Under the stronger assumption that the transition kernel and reward function have low rank, we show that the model-free algorithm in [37], which we refer to as Low Rank Empirical Value Iteration (LR-EVI), also achieves the desired sample complexity. Table 1 summarizes our sample complexity bounds in their corresponding settings, and compares them with existing results from literature in the tabular finite-horizon MDP setting; here d refers to the rank parameter.[1]

MDP Assumptions	Sample Complexity
Low-rank Q_h^* & suboptimality gap $\Delta_{\min} > 0$ (Theorem 7)	$\tilde{O}\left(\frac{d^3(\lvert S\rvert+\lvert A\rvert)H^4}{\Delta_{\min}^2}\right)$
ϵ-optimal policies have low-rank Q_h^π (Theorem 8)	$\tilde{O}\left(\frac{d^3(\lvert S\rvert+\lvert A\rvert)H^6}{\epsilon^2}\right)$
Transition kernels and rewards are low-rank (Theorem 9)	$\tilde{O}\left(\frac{d^3(\lvert S\rvert+\lvert A\rvert)H^5}{\epsilon^2}\right)$
Low-rank Q_h^* & constant horizon [37]	$\tilde{O}\left(\frac{d^5(\lvert S\rvert+\lvert A\rvert)}{\epsilon^2}\right)$
Tabular MDP with homogeneous rewards [38]	$\tilde{\Theta}\left(\frac{\lvert S\rvert\lvert A\rvert H^3}{\epsilon^2}\right)$

Table 1: Our sample complexity bounds alongside results from the literature, where d denotes the rank.

We extend our results to approximately low-rank MDPs, for which we show that our algorithm learns action-value functions with error $\epsilon + O(H^2\xi)$, where ξ is the rank-d approximation error, with an efficient number of samples. Furthermore, we empirically validate the improved efficiency of our low-rank algorithms. In the appendix, we show that our algorithm learns near-optimal action-value functions in a sample-efficient manner in the continuous setting, similar to the results in the table above. Finally, we prove that using existing convex program based matrix estimation methods instead of the one in [37] also achieves the desired reduction in sample complexity.

2 Related Work

Tabular Reinforcement Learning. Sample complexity bounds for reinforcement learning algorithms in the tabular MDP setting have been studied extensively, e.g., [3, 53, 11, 26]. Even with a generative model, $\Omega\left(|S||A|/\epsilon^2(1-\gamma)^3\right)$ samples are necessary to estimate an ϵ-optimal action-value function [6]. The work [38] presents an algorithm and associated analysis that achieves a matching upper bound on the sample complexity (up to logarithmic factors), proving that the lower bound is tight. Our work focuses on decreasing the sample complexity's dependence on $|S|$ and $|A|$ from $|S||A|$ to $|S| + |A|$ under models with a low-rank structure.

Complexity Measures for RL with General Function Approximation. The search for the most general types of structure that allow for sample-efficient reinforcement learning has resulted in many different complexity measures, including Bellman rank [22, 15], witness rank [41], Bellman

[1]The sample complexity bounds of Theorems 7, 8, and 9 presented in the table hide terms that are properties of the matrix, which are constant under common regularity assumptions (and will be discussed in later sections) and terms independent of $|S|$ or $|A|$.

Eluder dimension [23], and Bilinear Class [16]. For these classes of MDPs with rank d, finding an ϵ-optimal policy requires $\tilde{O}\left(\text{poly}(d, H)/\epsilon^2\right)$ samples. Unfortunately, these complexity measures are so broad that the resulting algorithms that achieve sample efficiency are often not polynomial time computable, and they rely on strong optimization oracles in general, e.g., assuming that we can solve a high dimensional non-convex optimization problem. We remark that our settings, including those under our strongest assumptions, cannot be easily incorporated into those frameworks.

Linear Function Approximation - Linear Realizability and Low Rank MDPs. To combat the curse of dimensionality, there is an active literature that combines linear function approximation with RL algorithms. As mentioned in the introduction, although these models are referred to as "low rank", they are significantly different than the type of low rank structure that we consider in our model. Most notably, the resulting Q^* matrix may not be low rank. As a result we only provide a brief overview of the results in this literature, largely to illustrate the types of properties that one would hope to study for our type of low rank model. One model class in this literature assumes that Q^* is linearly-realizable with respect to a known low dimensional feature representation, given by a known feature extractor $\phi : S \times A \to \mathbb{R}^d$ for $d \ll |S|, |A|$. [50, 51] show that an exponential number of samples in the minimum of the dimension d or the time horizon H may still be required under linear realizability, implying that additionally assumptions are required. These results highlight an interesting phenomenon that the dynamics of the MDP introduce additional complexities for linear function approximation in RL settings that are not present in supervised learning.

A more restrictive model class, sometimes referred to as *Linear/Low-rank MDPs*, imposes linearity on the dynamics of the MDP itself, i.e. the transition kernels and reward functions are linear with respect to a known low dimensional feature extractor ϕ [24, 55, 54, 48, 19]. As this does not impose structure on the relationship between s and a, the resulting Q functions may not be low rank. When the feature extractor is known, there are algorithms that achieve sample complexity or regret bounds that are polynomial in d with no dependence on $|S|$ or $|A|$. There have been attempts to extend these results to a setting where the feature mapping is not known [2, 32, 45], however the resulting algorithms are not polynomial time, as they require access to a strong nonconvex optimization oracle. Furthermore they restrict to a finite class of latent representation functions.

Low Rank Structure with respect to States and Actions. There is a limited set of works which consider a model class similar to ours, in which there is low rank structure with respect to the interaction between the states and actions and hence their interaction decomposes. This structure could be imposed on either the transition kernel, or only on the optimal Q^* function. [56, 34, 33] provide empirical results showing that Q^* and near-optimal Q functions for common stochastic control tasks have low rank. Their numerical experiments demonstrate that the performance of standard RL algorithms, e.g., value iteration and TD learning, can be significantly improved in combination with low-rank matrix/tensor estimation methods. The theoretical work [37] considers the weakest assumption that only imposes low rankness on Q^*. They develop an algorithm that combines a novel matrix estimation method with value iteration to find an ϵ-optimal action-value function with $\tilde{O}\left(d^5(|S| + |A|)/\epsilon^2\right)$ samples for infinite-horizon γ-discounted MDPs assuming that Q^* has rank d. While this is a significant improvement over the tabular lower bound $\tilde{\Omega}\left(|S||A|/((1-\gamma)^3\epsilon^2)\right)$ [6], their results require strict assumptions. The primary limitation is that they require the discount factor γ to be bounded from above by a small constant, which effectively limits their results to short, *constant* horizons. Lifting this limitation is left as an open question in their paper. In this work, we provide a concrete example that illustrates why long horizons may pose a challenge for using matrix estimation in RL. Subsequently we show that this long horizon barrier can be overcome by imposing additional structural assumptions. The algorithm in [37] also relies on prior knowledge of special anchor states and actions that span the entire space. We will

show that under standard regularity conditions, randomly sampling states and actions will suffice.

Matrix Estimation. Low-rank matrix estimation methods focus on recovering the missing entries of a partially observed low-rank matrix with noise. The field has been studied extensively with provable recovery guarantees; see the surveys [9, 12]. However, the majority of recovery guarantees of matrix estimation are in the Frobenius norm instead of an entry-wise/ℓ_∞ error bound, whereas a majority of common analyses for reinforcement learning algorithms rely upon constructing entrywise confidence sets for the estimated values. Matrix estimation methods with entry-wise error bounds are given in [10, 14, 1], but all require strict distributional assumptions on the noise, e.g., independent, mean-zero sub-Gaussian/Gaussian error. The matrix estimation method proposed in [37] provides entry-wise error guarantees for arbitrary bounded noise settings and is the method we use in our algorithm in order to aid our analysis.

3 Preliminaries

We consider a standard finite-horizon MDP given by (S, A, P, R, H) [42]. Here S and A are the finite state and action spaces, respectively. $H \in \mathbb{Z}_+$ is the time horizon. $P = \{P_h\}_{h\in[H]}$ is the transition kernel, where $P_h(s'|s, a)$ is the probability of transitioning to state s' when taking action a in state s at step h. $R = \{R_h\}_{h\in[H]}$ is the reward function, where $R_h : S \times A \to \Delta([0, 1])$ is the distribution of the reward for taking action a in state s at step h. We use $r_h(s, a) := \mathbb{E}_{r\sim R_h(s,a)}[r]$ as the mean reward. A stochastic, time-dependent policy of an agent has the form $\pi = \{\pi_h\}_{h\in[H]}$ with $\pi_h : S \to \Delta(A)$, where the agent selects an action according to the distribution $\pi_h(s)$ at time step h when at state s.

For each policy π, the value function and action-value function of π represent the expected total future reward obtained from following policy π given a starting state or state-action pair at step h,

$$V_h^\pi(s) := \mathbb{E}\left[\sum_{t=h}^H r_t(s_t, a_t) \;\middle|\; s_h = s\right], \tag{1}$$

$$Q_h^\pi(s, a) := \mathbb{E}\left[\sum_{t=h}^H r_t(s_t, a_t) \;\middle|\; s_h = s, a_h = a\right], \tag{2}$$

where $a_t \sim \pi_t(s_t)$ and $s_{t+1} \sim P_t(\cdot|s_t, a_t)$. The optimal value and action-value functions are given by $V_h^*(s) := \sup_\pi V_h^\pi(s)$ and $Q_h^*(s, a) := \sup_\pi Q_h^\pi(s, a)$, respectively, for all $s \in S, h \in [H]$. These functions satisfy the Bellman equations

$$V_h^*(s) = \max_{a\in A} Q_h^*(s, a), \quad Q_h^*(s, a) = r_h(s, a) + \mathbb{E}_{s'\sim P_h(\cdot|s,a)}[V_{h+1}^*(s')], \quad \forall s, a, h \tag{3}$$

with $V_{H+1}^*(s) = 0$. For an MDP with finite spaces and horizon, there always exists an optimal policy π^* that satisfies $V_h^{\pi^*}(s) = V_h^*(s)$ for all s, h.

The primary goal in this work is to find a near-optimal policy or action-value function. For $\epsilon > 0$, π is an ϵ-optimal policy if $|V_h^*(s) - V_h^\pi(s)| \le \epsilon, \forall(s, h) \in S \times [H]$. Similarly, $Q = \{Q_h\}_{h\in[H]}$ is called an ϵ-optimal action-value function if $|Q_h^*(s, a) - Q_h(s, a)| \le \epsilon, \forall(s, a, h) \in S \times A \times [H]$. We will view Q_h^*, Q_h^π and r_h as $|S|$-by-$|A|$ matrices and $P_h(\cdot|\cdot, \cdot)$ as an $|S|$-by-$|S|$-by-$|A|$ tensor, for which various low-rank assumptions are considered. For a given function $V : S \to \mathbb{R}$, we sometimes use the shorthand $[P_h V](s, a) := \mathbb{E}_{s'\sim P_h(\cdot|s,a)}[V(s')]$ for the conditional expectation under P_h.

Throughout this paper, we assume access to a simulator (a.k.a. the generative model framework, introduced by [25]), which takes as input a tuple $(s, a, h) \in S \times A \times [H]$ and outputs independent samples $s' \sim P_h(\cdot|s, a)$ and $r \sim R_h(s, a)$. This assumption is one of the stronger assumptions in reinforcement learning literature, but common in the line of work that studies sample complexity without directly addressing the issue of exploration, e.g., [38, 3].

Notation. Let $a \wedge b := \min(a, b)$, $a \vee b := \max(a, b)$, δ_a denote the distribution over A that puts probability 1 on action a, $\sigma_i(M)$ denote the i-th largest singular value of a matrix M, and M_i denote the i-th row. The n-by-n identity matrix is denoted by $I_{n\times n}$, and $[H] := \{1, \dots, H\}$. We use several vector and matrix norms: Euclidean/ℓ_2 norm $\|\cdot\|_2$, spectral norm $\|M\|_{op} = \sigma_1(M)$, nuclear norm $\|M\|_*$, entrywise ℓ_∞ norm $\|M\|_\infty$ (largest absolute value of entries), and Frobenius norm $\|M\|_F$. We define the condition number of a rank-d matrix M as $\kappa_M := \frac{\sigma_1(M)}{\sigma_d(M)}$.

4 Information Theoretic Lower Bound

While one may hope to learn the optimal action-value functions when only assuming that Q_h^* is low rank, we argue that the problem is more nuanced. Specifically, we present two similar MDPs with rank-one Q^*, where the learner has complete knowledge of the MDP except for one state-action pair at each time step. As the learner is restricted from querying that specified state-action pair, in order to distinguish between the two MDPs and learn the optimal policy, the learner must use the low-rank structure to estimate the unknown entry. We then show that doing so requires a exponential number of observations in the horizon H.

Consider MDPs $M^\theta = (S, A, P, R^\theta, H)$ indexed by a real number θ, where $S = A = \{1, 2\}$. At $h = 1$, $r_1^\theta(s_1, a) = 0$, $P_1(\cdot|s_0, a) = \delta_a$ for all $a \in A$, and the starting state s_1 is deterministic. For $h > 1$,

$$r_H^\theta = \begin{pmatrix} \frac{1}{2} \\ 1 + 2\theta \end{pmatrix}, \qquad r_h^\theta = \begin{pmatrix} -\frac{1}{4} & 0 \\ -\frac{1}{2} & 2^{H-h}\theta \end{pmatrix}, \qquad \text{and } P_h(\cdot|s, a) = \delta_s, \qquad \forall s, a, \forall h \in \{2, \dots, H-1\},$$

where δ_s denotes the Dirac delta distribution at s. The rewards are deterministic except for the terminal reward at state 2, where the reward distribution $R_H^\theta(2)$ is such that the reward takes value 2 with probability $\frac{1}{2} + \theta$, and takes value 0 otherwise.

If action $a = 1$ (resp., 2) is taken at the initial step $h = 1$, the MDP will transition to state 1 (resp., 2) and then stay at this state in all subsequent steps. Thus learning the optimal policy only depends on determining the optimal action in step 1. Let θ take one of two possible values: $\theta_1 = -\frac{3}{4\cdot 2^{H-1}}$ and $\theta_2 = \frac{3}{4\cdot 2^{H-1}}$. To determine the correct action at the initial step, one must correctly identify θ. We will show that identifying θ takes an exponential number of samples in the horizon H.

Lemma 1. *The optimal policy for the above MDP (for both values of θ) for steps $h \geq 1$ is $\pi_h^*(1) = \pi_h^*(2) = 2$ for all $h \in \{2, \dots, H-1\}$. Furthermore,*

$$Q_h^{*,\theta} = \begin{pmatrix} \frac{1}{4} \\ \frac{1}{2} + 2^{H-h}\theta, & 1 + 2^{H-h+1}\theta \end{pmatrix}, \qquad V_h^{*,\theta} = \begin{pmatrix} \frac{1}{2} \\ 1 + 2^{H-h+1}\theta \end{pmatrix}, \qquad \forall h \in \{2, \dots, H-1\}.$$

Lemma 1, proved in Appendix E, shows that Q_h^* is rank one. We will calculate the optimal Q function and policy at step $h = 1$ in the proof of Theorem 2 after introducing the observation model.

Observation model: The learner has exact knowledge of $r_H^\theta(1)$, $r_h^\theta(s, a)$, P_h, r_1^θ, P_1 for all $(s, a) \in \Omega := \{(1, 1), (1, 2), (2, 1)\}$ and $h \in [H-1]$. Note that these known rewards and transitions are independent of θ. In addition, the learner is given n iid samples from $R_H^\theta(2)$.

One interpretation of this observation model is that the learner has infinitely many samples of the form (s, a, s'), $s' \sim P_h(\cdot|s, a)$ for each (s, a), so P_h can be estimated with zero error. Similarly, the learner has infinitely many samples from $r_H^\theta(1)$ and $r_h^\theta(s, a)$ for $(s, a) \in \Omega$. However, the learner cannot observe $r_h^\theta(2, 2)$ and hence must estimate $Q_h^\theta(2, 2)$ using the low-rank structure. Finally, n noisy observations of the terminal reward $r_H^\theta(2)$ at state 2 are given.

Theorem 2. *Consider the above class of MDPs and observation model. To learn a* $1/8$*-optimal policy with probability at least 0.9, the learner must observe* $n = \Omega(4^H)$ *samples from* $R_H^\theta(2)$.

Proof. From Lemma 1, we have $V_2^{*,\theta_1} = \begin{pmatrix} 1/2 \\ 1/4 \end{pmatrix}$ and $V_2^{*,\theta_2} = \begin{pmatrix} 1/2 \\ 7/4 \end{pmatrix}$. Hence, at $h = 1$ the optimal action is $\pi_1^*(s_0) = 1$ for $\theta = \theta_1$ and $\pi_1^*(s_0) = 2$ for $\theta = \theta_2$. If $\theta = \theta_1$ and action 2 is taken instead, $\pi_1(s_0) = 2$, then this action incurs a $1/4$ penalty in value relative to the optimal action, i.e., $Q_1^{*,\theta_1}(s_0, 2) \leq Q_1^{*,\theta_1}(s_0, 1) - \frac{1}{4}$. If $\theta = \theta_2$ and action 1 is taken, $\pi_1(s_0) = 1$, then this action incurs a $5/4$ penalty relative to the optimal action. Therefore, to learn an ϵ-optimal policy for $\epsilon < 1/4$, e.g., $\epsilon = 1/8$ as stated in the theorem, the learner must correctly determine whether $\theta = \theta_1$ or $\theta = \theta_2$. It is well known from existing literature, see e.g., [49, 4, 29], that one needs $\Omega\left(1/(2\theta_1 - 2\theta_2)^2\right)$ samples to distinguish two (scaled) Bernoulli distributions with mean $1/2 + \theta, \theta \in \{\theta_1, \theta_2\}$ with probability at least 0.9. Substituting $(2\theta_1 - 2\theta_2)^2 > \frac{4^{H-1}}{9}$ proves the result. □

Consider the following operational interpretation of the above example. The learner can use the rank-one structure to estimate $Q_h^*(2,2)$ given $Q_h^*(s,a), (s,a) \in \Omega$ as follows: $Q_h^*(2,2) = Q_h^*(1,2)Q_h^*(2,1)/Q_h^*(1,1)$, coinciding with the matrix estimation algorithm in [37]. Lemma 1 shows that an $\varepsilon = 2^{H-h}\theta$ error in $Q_h^*(2,1)$ leads to a $2 \cdot \varepsilon = 2^{H-h+1}\theta$ error in $Q_h^*(2,2)$ and $Q_{h-1}^*(2,1)$. As such, the error is amplified exponentially when propagating backwards through the horizon, showing that this low-rank based procedure is inherently unstable.

This example illustrates that reinforcement learning with low-rank structure is more nuanced than low-rank estimation without dynamics, and that the constant horizon assumption in [37] is not merely an artifact of their analysis. Furthermore, as the entries of Q_h^* are similar in magnitude, the blow up in error is not due to the missing entry containing most of the signal. This motivates us to consider additional assumptions beyond Q_h^* being low rank. In the above example, the optimal state-action pair $(2,2)$ is not observed, and the reward r_h and transition kernel P_h are not low-rank. To achieve stable and sample-efficient learning with long horizons, we will consider when additional structures in the MDP dynamics can be exploited to identify and sample from the optimal action.

5 Assumptions

In this section, we present three low rank settings that enable sample-efficient reinforcement learning, with each setting increasing in the strength of the low rank structural assumption.

Assumption 1 (Low-rank Q_h^*)**.** *For all* $h \in [H]$*, the rank of the matrix* Q_h^* *is* d*. Consequently,* Q_h^* *can be represented via its singular value decomposition* $Q_h^* = U^{(h)}\Sigma^{(h)}(V^{(h)})^\top$*, for a* $|S| \times d$ *orthonormal matrix* $U^{(h)}$*, a* $|A| \times d$ *orthonormal matrix* $V^{(h)}$*, and a* $d \times d$ *diagonal matrix* $\Sigma^{(h)}$*.*

Assumption 1 imposes that the action-value function of the optimal policy is low rank. This assumption can be contrasted with another common structural assumption in the literature, namely linearly-realizable Q^*, meaning that $Q_h^*(s,a) = w_h^\top \phi(s,a)$ for some weight vector $w_h \in \mathbb{R}^d$ and a *known* feature mapping $\phi : S \times A \to \mathbb{R}^d$ [50, 51]. In comparison, Assumption 1 decomposes ϕ into the product of separate feature mappings on the state space $U^{(h)}$ and the action space $V^{(h)}$. Hence, linearly-realizable Q^* does not imply low-rank Q_h^*. Furthermore, we assume the latent factors $U^{(h)}$ and $V^{(h)}$ are completely unknown, whereas the linear function approximation literature typically assumes ϕ is known or approximately known.

Assumption 1 only imposes low-rankness on Q_h^*, allowing for the Q^π function associated to non-optimal policies π to be full rank. Assumption 1 is likely too weak, as Theorem 2 illustrates a doubly exponential growth in policy evaluation error under only this assumption. Below we

present three additional assumptions. *Each* of these assumptions enable our algorithms to achieve the desired sample complexity when coupled with Assumption 1.

Assumption 2 (Suboptimality Gap)**.** *For each $(s,a) \in S \times A$, the suboptimality gap is defined as $\Delta_h(s,a) := V_h^*(s) - Q_h^*(s,a)$. Assume that there exists an $\Delta_{\min} > 0$ such that*

$$\min_{h\in[H],s\in S,a\in A} \{\Delta_h(s,a) : \Delta_h(s,a) > 0\} \geq \Delta_{\min}.$$

Assumption 2 stipulates the existence of a suboptimality gap bounded away from zero. In the finite setting with $|S|, |A|, H < \infty$, there always exists a $\Delta_{\min} > 0$ for any non-trivial MDP in which there is at least one suboptimal action. This is an assumption commonly used in bandit and reinforcement learning literature.

Assumption 3 (ϵ-optimal Policies have Low-rank Q Functions)**.** *For all ϵ-optimal policies π, the associated Q_h^π matrices are rank-d for all $h \in [H]$, i.e., Q_h^π can be represented via $Q_h^\pi = U^{(h)}\Sigma^{(h)}(V^{(h)})^\top$ for some $|S| \times d$ matrix $U^{(h)}$, $|A| \times d$ matrix $V^{(h)}$, and $d \times d$ diagonal matrix $\Sigma^{(h)}$.*

Assumption 3 imposes that all ϵ-optimal policies π have low-rank Q_h^π. We have not seen this assumption in existing literature. It is implied by the stronger assumption that *all policies* have low-rank Q_h^π; see Appendix B for an MDP that satisfies Assumption 3 but fails the stronger assumption. The stronger assumption is analogous to the property that Q^π is linear in the feature map ϕ for *all* policies, which is commonly used in work on linear function approximation and linear MDPs.

To state our strongest low-rank assumption, we first recall the definition of tensor Tucker rank.

Definition 3 (Tucker Rank [28])**.** *The Tucker rank of a tensor $X \in \mathbb{R}^{n_1\times n_2\times n3}$ is the smallest (d_1, d_2, d_3) such that there exists a core tensor $G \in \mathbb{R}^{d_1\times d_2\times d_3}$ and orthonormal latent factor matrices $A_i \in \mathbb{R}^{n_i\times d_i}$ for $i \in [3]$ such that for all $(a,b,c) \in [n_1] \times [n_2] \times [n_3]$,*

$$X(a,b,c) = \sum_{\ell_1\in[d_1]} \sum_{\ell_2\in[d_2]} \sum_{\ell_3\in[d_3]} G(\ell_1,\ell_2,\ell_3)A_1(a,\ell_1)A_2(b,\ell_2)A_3(c,\ell_3).$$

Our strongest low-rank assumption imposes that the expected reward functions are low rank, and the transition kernels have low Tucker rank *along one dimension*.

Assumption 4 (Low-rank Transition Kernels and Reward Functions)**.** *The expected reward function has rank d, and the transition kernel P_h has Tucker rank $(|S|, |S|, d)$ or $(|S|, d, |A|)$, with shared latent factors. For the Tucker rank $(|S|, |S|, d)$ case, this means that for each $h \in [H]$, there exists a $|S| \times |S| \times d$ tensor $U^{(h)}$, an $|A| \times d$ matrix $V^{(h)}$, and an $|S| \times d$ matrix $W^{(h)}$ such that*

$$P_h(s'|s,a) = \textstyle\sum_{i=1}^d U^{(h)}(s',s,i)V^{(h)}(a,i) \quad \text{and} \quad r_h(s,a) = \sum_{i=1}^d W^{(h)}(s,i)V^{(h)}(a,i).$$

For the Tucker rank $(|S|, d, |A|)$ case, this means that for each $h \in [H]$, there exists a $|S| \times |A| \times d$ tensor $V^{(h)}$, an $|S| \times d$ matrix $U^{(h)}$, and an $|A| \times d$ matrix $W^{(h)}$ such that

$$P_h(s'|s,a) = \textstyle\sum_{i=1}^d U^{(h)}(s,i)V^{(h)}(s',a,i) \quad \text{and} \quad r_h(s,a) = \sum_{i=1}^d U^{(h)}(s,i)W^{(h)}(a,i).$$

Assumption 4 is our strongest low-rank structural assumption as it implies that the Q_h^π functions associated with *any* policy π are low rank, which subsequently implies both Assumptions 3 and 1. In fact, Assumption 4 implies that for any value function estimate $\hat{V}_h$, the matrix $r_h + [P_h\hat{V}_{h+1}]$ is low rank, as stated in the following proposition.

Proposition 4. *If the transition kernel has Tucker rank* $(|S|, |S|, d)$ *or* $(|S|, d, |A|)$ *and the expected reward function has rank* d *with shared latent factors, i.e., Assumption 4 holds, then the matrix* $r_h + [P_h \hat{V}_{h+1}]$ *has rank at most* d *for any* $\hat{V}_{h+1} \in \mathbb{R}^{|S|}$.

See Appendix F for the proof. Proposition 4 results from the fact that for any fixed h, the matrices corresponding to r_h and $P_h(s'|\cdot, \cdot)$ for all s' share either the same column or row space, which is critically used in the analysis of our Low Rank Empirical Value Iteration algorithm.

Next we present several definitions used to characterize the error guarantees of the matrix estimation algorithm. It is commonly understood in the matrix estimation literature that other properties of the matrix beyond low rank, such as its incoherence or condition number, govern how efficiently a matrix can be estimated. Consider a trivial rank-1 MDP where $H = 1$ and the reward is a sparse matrix with only d nonzero entries taking value 1. Since the locations of the nonzero entries are unknown, we will likely observe only zeros upon sampling any small subset of entries. Estimation using a small number of samples would be possible, however, if an expert were to provide knowledge of a special set of rows and columns, which have been referred to as *anchor* states and actions in [37]. For some sets $S_h^\# \subseteq S$ and $A_h^\# \subseteq A$, we use $Q_h(S_h^\#, A_h^\#)$ to denote the submatrix obtained by restricting Q_h to state-action pairs from $S_h^\# \times A_h^\#$.

Definition 5 ((k, α)-Anchor States and Actions)**.** *A set of states* $S_h^\# \subset S$ *and a set of actions* $A_h^\# \subset A$ *are* (k, α)*-anchor states and actions for a rank-*d *matrix* Q_h *if* $|S_h^\#|, |A_h^\#| \leq k$*, the submatrix* $Q_h(S_h^\#, A_h^\#)$ *has rank* d*, and* $\|Q_h\|_\infty / \sigma_d(Q_h(S_h^\#, A_h^\#)) \leq \alpha$.

Any set of valid anchor states and anchor actions must have at least size d in order for the associated anchor submatrix to be rank d. As the full matrix Q_h has rank d, this also implies that all rows (resp., columns) of Q_h can be written as a linear combination of the rows associated to states $S_h^\#$ (resp., columns associated to actions $A_h^\#$). The parameter α depends on the quality of the anchor sets; sub-matrices that are close to being singular result in large α. We remark that assuming knowledge of a minimal set of anchor states and actions is common in literature, i.e., anchor-based topic modelling [5, 8] and linear feature-based RL [48, 55]. Furthermore, Shah et al. [37] posit that it suffices empirically to choose states and actions that are far from each other as anchor states and actions. However, in the worst case, finding valid anchor states and actions may require significant a priori knowledge about the unknown matrix.

Alternately, anchor states and actions can be randomly constructed for matrices that satisfy standard regularity conditions such as incoherence, commonly used in matrix estimation [7].

Definition 6 (Incoherence)**.** *Let* $Q_h \in \mathbb{R}^{|S| \times |A|}$ *be a rank-*d *matrix with singular value decomposition* $Q_h = U \Sigma V^\top$ *with* $U \in \mathbb{R}^{|S| \times d}$ *and* $V \in \mathbb{R}^{|A| \times d}$*.* Q_h *is* μ*-incoherent if* $\max_{i \in [|S|]} \|U_i\|_2 \leq \sqrt{\mu d / |S|}$ *and* $\max_{j \in [|A|]} \|V_j\|_2 \leq \sqrt{\mu d / |A|}$*, where* U_i *denotes the* i*-th row of a matrix* U.

A small incoherence parameter μ ensures that the masses of U and V are not too concentrated in a couple of rows or columns. Consequently, a randomly sampled subset of rows (resp., columns) will span the row (resp., column) space, so these subsets of rows and columns contain sufficient information to reconstruct the entire matrix. Both μ and κ, the condition number of Q_h, will be used in the analysis to show that the entrywise error amplification from the matrix estimation method scales with μ, d, κ instead of the size of the state or action space, k, or α.

Discussion of Assumptions. While low rank structure with incoherence is widely accepted in the matrix and tensor estimation literature, we provide a few examples to illustrate how these properties could also naturally arise in MDPs. Consider a continuous MDP which is converted to a tabular MDP via discretization, which is a common approach for tackling continuous MDPs.

As the size of the discretization is artificial, the true complexity of the MDP is governed by the structure of the continuous MDP, which is independent of the discretization size. As long as the reward function and dynamics are sufficiently smooth with respect to the continuous MDP, the resulting tabular MDP will have approximate low-rank structure as d would be at most logarithmic with respect to $|S|, |A|$, due to a universal low rank property of smooth functions [44]. Additionally, the incoherence condition intuitively states that there cannot be a disproportionately small set of rows or columns that represent a disproportionately large amount of the signal. For MDPs that are derived from uniform discretizations of continuous MDPs with smoothness properties, incoherence also arises naturally as there will be a constant fraction of the rows or columns representing any fixed length interval of the continuous state space. Even in inherently discrete settings such as a recommendation system with users and movies, when the population is sufficiently large, one could view the discrete population of states/actions as representing a sample from an underlying continuous population with appropriate smoothness conditions. Finally, in many physical systems as relevant to most stochastic control tasks, there exist low dimensional feature representations that capture the "sufficient statistics" of the state, which fully govern the dynamics of the system.

6 Algorithm

Our algorithm follows from a natural synthesis of matrix estimation with empirical value iteration and Monte Carlo policy iteration. We first describe the vanilla approximate dynamic programming algorithms for the general tabular MDP settings. Empirical value iteration simply replaces the expectation in the Bellman update in Equation (3) with empirical samples [20]. Specifically, to estimate $Q^*_h(s,a)$, one collects N samples of one step transitions, which entails sampling a reward and next state from $R_h(s,a)$ and $P_h(\cdot|s,a)$. Let $\hat{r}_h(s,a)$ denote the empirical average reward of the N samples from $R_h(s,a)$. Let $\hat{P}_h(\cdot|s,a)$ denote the empirical distribution over N next states sampled from $P_h(\cdot|s,a)$. Given an estimate $\hat{V}_{h+1}$ for the optimal value function at step $h+1$, the empirical Bellman update equation is

$$\hat{Q}_h(s,a) = \hat{r}_h(s,a) + \mathbb{E}_{s' \sim \hat{P}_h(\cdot|s,a)}[\hat{V}_{h+1}(s')], \quad \text{and} \quad \hat{V}_h(s) = \max_{a \in A} \hat{Q}_h(s,a). \tag{4}$$

Evaluating $\hat{Q}_h$ and $\hat{V}_h$ requires collecting N samples for each of the $|S||A|$ state action pairs (s,a).

Monte Carlo policy iteration for tabular MDPs approximates $Q^\pi_h(s,a)$ for a policy π by replacing the expectation in the definition (2) of Q^π with empirical trajectory samples, which is similar to first-visit Monte Carlo policy evaluation except we use the generative model to start at a specified state-action pair and time step [42]. This involves sampling N independent trajectories starting from state-action pair (s,a) at step h and following a given policy π until the end of the horizon H. For a fixed policy π and state action pair (s,a), let the sequence of rewards along the i-th sampled trajectory be denoted $(r^i_h, r^i_{h+1}, \dots r^i_H)$. We will use $\hat{r}^{\text{cum}}_h(s,a)$ to denote the empirical average cumulative reward across the N trajectories, given by

$$\hat{r}^{\text{cum}}_h(s,a) := \tfrac{1}{N} \textstyle\sum_{i=1}^N \sum_{t=h}^H r^i_t. \tag{5}$$

Given an estimate of the optimal policy for steps greater than h, denoted by $(\hat{\pi}_{h+1}, \hat{\pi}_{h+2}, \dots \hat{\pi}_H)$, the Monte Carlo estimate for the optimal action-value function and optimal policy at step h are

$$\hat{Q}_h(s,a) = \hat{r}^{\text{cum}}_h(s,a), \quad \text{and} \quad \hat{\pi}_h(s) = \delta_a \ \text{ for } a = \operatorname*{argmax}_{a' \in A} \hat{Q}_h(s,a'), \tag{6}$$

where the trajectories used to compute $\hat{r}^{\text{cum}}_h(s,a)$ are sampled by following the policy $(\hat{\pi}_{h+1}, \hat{\pi}_{h+2}, \dots \hat{\pi}_H)$, and recall δ_a denotes the distribution that puts probability 1 on action a. Computing $\hat{Q}_h$ and $\hat{\pi}_h$

involves sampling $|S||A|N$ trajectories, which are each of length $H-h$, which results in a sample complexity of $|S||A|N(H-h)$ individual transitions from the MDP.

The dependence on $|S||A|$ in the sample complexity for both of the classical algorithms described above is due to using empirical samples to evaluate $\hat{Q}_h$ for every state-action pair $(s,a) \in S \times A$. The assumption that Q_h^* is at most rank d imposes constraints on the relationship between $Q_h^*(s,a)$ at different state-action pairs, such that by approximating Q_h^* using empirical samples at only $O(d|S|+d|A|)$ locations, we should intuitively be able to use the low rank constraint to predict the remaining entries. Let $\Omega_h \subset S \times A$ denote the subset of entries (s,a) for which we use empirical samples to approximate $\hat{Q}_h(s,a)$, computed via either (4) or (6). Given estimates of $\hat{Q}_h(s,a)$ at $(s,a) \in \Omega_h$, we can then use a low-rank matrix estimation subroutine to estimate the Q function for $(s,a) \notin \Omega$. This is the core concept of our algorithm, which we then combine with the two classical approaches of empirical value iteration and Monte Carlo policy iteration.

6.1 Formal Algorithm Statement

We present two Low Rank RL algorithms, which take as input any matrix estimation algorithm, $\texttt{ME}(\cdot)$, that takes in a subset of entries of the matrix and returns an estimate of the whole matrix, the sets $\{\Omega_h\}_{h\in[H]}$ that indicate the state action pairs for which data should be collected by querying the MDP generative model, and $\{N_{s,a,h}\}_{(s,a,h)\in S\times A\times H}$, which denotes how many samples to query at state-action pair (s,a) at timestep h. We use "Low Rank Empirical Value Iteration" (LR-EVI) to refer to the algorithm which uses option (a) for Step 1 below, and we use "Low Rank Monte Carlo Policy Iteration" (LR-MCPI) to refer to the algorithm which uses option (b) for Step 1.

Hyperparameters: $\{\Omega_h\}_{h\in[H]}$, $\{N_{s,a,h}\}_{(s,a,h)\in S\times A\times H}$, and $\texttt{ME}(\cdot)$

Initialize: Set $\hat{V}_{H+1}(s) = 0$ for all s, and let $\hat{\pi}^{H+1}$ be any arbitrary policy.

For each $h \in \{H, H-1, H-2, \dots 1\}$ in descending order,

- **Step 1:** For each $(s,a) \in \Omega_h$, compute $\hat{Q}_h(s,a)$ using empirical estimates according to either (a) empirical value iteration or (b) Monte Carlo policy evaluation.

  (a) **Empirical Value Iteration:** Collect $N_{s,a,h}$ samples of a single transition starting from state s and action a at step h. Use the samples to estimate $\hat{Q}_h(s,a)$ according to

  $$\hat{Q}_h(s,a) = \hat{r}_h(s,a) + \mathbb{E}_{s'\sim\hat{P}_h(\cdot|s,a)}[\hat{V}_{h+1}(s')],$$

  where $\hat{r}_h(s,a)$ denotes the empirical average reward of the $N_{s,a,h}$ samples from $R_h(s,a)$, and $\hat{P}_h(\cdot|s,a)$ denotes the empirical distribution over the $N_{s,a,h}$ states sampled from $P_h(\cdot|s,a)$.

  (b) **Monte Carlo Policy Evaluation:** Collect $N_{s,a,h}$ independent full trajectories starting from state s and action a at step h until the end of the horizon H, where actions are chosen according to the estimated policy $(\hat{\pi}_{h+1}, \hat{\pi}_{h+2}, \dots \hat{\pi}_H)$. Let $\hat{Q}_h(s,a) = \hat{r}_h^{\text{cum}}(s,a)$, where $\hat{r}_h^{\text{cum}}(s,a)$ denotes the empirical average cumulative reward across the $N_{s,a,h}$ trajectories starting from (s,a) at step h. If $(r_h^i, r_{h+1}^i, \dots r_H^i)$ denotes the sequence of rewards along the i-th sampled trajectory from (s,a) at step h, then

  $$\hat{Q}_h(s,a) = \hat{r}_h^{\text{cum}}(s,a) := \tfrac{1}{N_{s,a,h}} \textstyle\sum_{i=1}^{N_{s,a,h}} \sum_{t=h}^{H} r_t^i.$$

- **Step 2:** Predict the action-value function for all $(s,a) \in S \times A$ according to $\texttt{ME}(\cdot)$:

  $$\bar{Q}_h = \texttt{ME}\left(\{\hat{Q}_h(s,a)\}_{(s,a)\in\Omega_h}\right).$$

- **Step 3:** Compute the estimates of the value function and the optimal policy according to

$$\hat{V}_h(s) = \max_{a \in A} \bar{Q}_h(s,a) \quad \text{and} \quad \hat{\pi}_h(s) = \delta_{\operatorname{argmax} \bar{Q}_h(s,a)}.$$

The tabular MDP variant of the algorithm proposed in [37] is equivalent to LR-EVI where anchor states $S_h^\#$ and actions $A_h^\#$ are given and $\Omega_h = (S_h^\# \times A) \cup (S \times A_h^\#)$. Furthermore, LR-EVI is equivalent to a modification of the algorithm in [56] with a different choice for the matrix estimation algorithm used in Step 2 and the corresponding sample set Ω_h constructed in Step 1.

6.2 Matrix Estimation Subroutine

A critical piece to specify for the algorithm above is how to choose the subset Ω_h, and what matrix estimation subroutine $\texttt{ME}(\cdot)$ to use to predict the full Q_h function, where Q_h is Q_h^*, Q_h^π, or $Q_h' = r_h + P_h \hat{V}_{h+1}$ depending on the low-rank setting, given $\hat{Q}_h(s,a)$ for $(s,a) \in \Omega_h$. The performance of any matrix estimation algorithm will depend both on the selected subset Ω_h, as well as the entrywise noise distribution on $\hat{Q}_h(s,a)$ relative to the "ground truth" low-rank matrix. As a result, the subset Ω_h should be determined jointly with the choice of matrix estimation algorithm.

A limitation of a majority of results in the classical matrix estimation literature is that they do not admit entrywise bounds on the estimation error, and the analyses may be sensitive to the distribution of the observation error , i.e., require mean-zero sub-Gaussian noise. When estimating Q_h^*, the observations are biased unless one has learned the optimal policy at time steps $h+1$ to H. Since Q_h is low-rank under our assumptions, our estimates of the observations for the matrix estimation method are unbiased with bounded noise, therefore enabling us to relax the small discount factor requirement.

Many standard analyses of RL algorithms rely upon the construction of entrywise confidence sets for the estimates of the Q function. Our results and analyses rely on entrywise error bounds for the matrix estimation step that balance the worst case entrywise error amplification with the size of the observation set. As such, similar theoretical guarantees can be obtained for our algorithm under any matrix estimation method that admits suitable entrywise error bounds.

The majority of our theoretical results will be shown for the variant of the algorithm that uses a matrix estimation algorithm from [37], which is incidentally equivalent to exploiting a skeleton decomposition of a low rank matrix [18]. Their algorithm uses a specific sampling pattern, in which Ω_h is constructed according to $\Omega_h = (S^\# \times A) \cup (S \times A^\#)$, where $S_h^\#$ and $A_h^\#$ are assumed to be valid anchor states and actions for the matrix Q_h (cf. Definition 5). Given estimates $\hat{Q}_h(s,a)$ for all $(s,a) \in \Omega_h$, their algorithm estimates the Q function at all state action pairs according to

$$\bar{Q}_h(s,a) = \hat{Q}_h(s, A^\#) \left[\hat{Q}_h(S^\#, A^\#) \right]^\dagger \hat{Q}_h(S^\#, a), \tag{7}$$

where $M^\dagger$ denotes the pseudoinverse of M, and $\bar{Q}$ is the output of the matrix estimation algorithm. The simple explicit formula for the estimates enables direct entrywise error bounds. Instead of ensuring a uniform error bound over each state-action pair in $(S^\# \times A) \cup (S \times A^\#)$, we show that additional sampling of the anchor submatrix $\Omega_h^\# = S^\# \times A^\#$ yields a smaller error amplification compared to the method propsosed in [37]. In addition, we show that if Q_h is μ-incoherent, introduced in Definition 6, $\tilde{O}(\mu d, \kappa)$-anchor states and actions can be constructed randomly by including each state in $S^\#$ independently with probability $p_1 = \Theta(\mu d \log(|S|)/|S|)$ and including each action in $A^\#$ independently with probability $p_2 = \Theta(\mu d \log(|S|)/|S|)$. As a result, a priori knowledge of the anchor states and actions is not required under these regularity conditions.

In Section A.3, we show that our theoretical results also extend to the variation of our algorithm that uses soft nuclear norm minimization for matrix estimation alongside uniform Bernoulli sampling, utilizing entrywise guarantees shown in [10]. One matrix estimation algorithm that solves the soft nuclear norm minimization problem is `Soft-Impute` [30]. `Soft-Impute` proceeds by iteratively filling in the missing values by using a soft-thresholded singular value decomposition on the matrix of observed entries. In contrast to the sampling pattern used in the matrix estimation method given in Equation 13, the sampling pattern needed to ensure the entrywise guarantees from [10] assumes that each state-action pair is observed with probability p_{SI}.

We use LR-EVI (resp., LR-MCPI) + `SI` to refer to the algorithm that uses option (a) (resp., option (b)) for Step 1 and `Soft-Impute` as the matrix estimation method. In Section 8, we empirically compare LR-EVI and LR-MCPI for both variations of matrix estimation algorithms.

7 Main Results

In this section, we present the sample complexity, i.e., an upper bound on the number of observed samples of the reward and next state, guarantees for LR-MCPI and LR-EVI with the matrix estimation method presented in Section 6.2 under different low-rank assumptions, from the weakest to the strongest. For $(s,a) \notin \Omega_h^\# = S_h^\# \times A_h^\#$, we denote $N_{s,a,h} = N_h$, For $(s,a) \in \Omega_h^\# = S_h^\# \times A_h^\#$, we denote $N_{s,a,h} = N_h^\# = \alpha^2 k^2 N_h$, such that entries in the anchor submatrix get a factor of $\alpha^2 k^2$ more samples.

Theorem 7. *Assume that Q_h^* is rank d and has suboptimality gap $\Delta_{\min}$ (Assumptions 1 and 2), and $S_h^\#, A_h^\#$ are (k,α)-anchor states and actions for Q_h^* for all $h \in [H]$. Let $N_h = \tilde{O}\left((H-h+1)^2\alpha^2k^2/\Delta_{\min}^2\right)$ and $N_h^\# = \alpha^2k^2N_h$. LR-MCPI returns an optimal policy with probability at least $1-\delta$ with a sample complexity of $\tilde{O}\left((|S|+|A|)\alpha^2k^3H^4/\Delta_{min}^2 + \alpha^4k^6H^4/\Delta_{\min}^2\right)$.*

The dependence on the rank d is not explicitly shown in the sample complexities stated in these theorems as it is captured by k, which we bound with Lemma 10 (presented later in this section). In the tabular setting, there always exists a $\Delta_{\min} > 0$. This sample complexity improves upon $|S||A|$ when $\Delta_{\min}$ is greater than $|S|^{-1/2} \wedge |A|^{-1/2}$. When $\Delta_{\min}$ is small, if stronger low-rank assumptions also hold, then the results in Theorems 8 and 9 below may provide stronger bounds.

Under the assumption that the Q_h^π function is low rank for all ϵ-optimal policies, Theorem 8 states that LR-MCPI learns an ϵ-optimal policy with a sample complexity independent of $\Delta_{\min}$.

Theorem 8. *Assume that for all ϵ-optimal policies π, Q_h^π is rank d (Assumption 3), and $S_h^\#, A_h^\#$ are (k,α)-anchor states and actions for $Q_h^{\hat{\pi}}$, where $\hat{\pi}$ is the learned policy from LR-MCPI for all $h \in [H]$. Let $N_h = \tilde{O}\left((H-h+1)^2\alpha^2k^2H^2/\epsilon^2\right)$ and $N_h^\# = \alpha^2k^2N_h$. Then, LR-MCPI returns an ϵ-optimal policy and action-value function with probability at least $1-\delta$ with a sample complexity of $\tilde{O}\left((|S|+|A|)\alpha^2k^3H^6/\epsilon^2 + \alpha^4k^6H^6/\epsilon^2\right)$.*

The strongest assumption that the transition kernel has low Tucker rank and the reward function is low rank, implies that Q_h^π for all policies π is low rank. As such, the result in Theorem 8 also implies an efficient sample complexity guarantee for LR-MCPI under Assumption 4. We can further remove a factor of H by using LR-EVI instead. Empirical value iteration (see Step 1(a)) reduces the sample complexity by a factor of H since it does not require sampling a full rollout of the policy to the end of the horizon, as required for the Monte Carlo estimates (see Step 1(b)).

Theorem 9. *Assume that for any ϵ-optimal value function V_{h+1}, the matrix corresponding to $Q'_h = [r_h + [P_hV_{h+1}]]$ is rank d (a consequence of Assumption 4), and $S_h^\#, A_h^\#$ are (k,α)-anchor*

states and actions for $\hat{Q}'_h = [r_h + [P_h \hat{V}_{h+1}]]$*, where* $\hat{V}_{h+1}$ *is the learned value function from LR-EVI for all* $h \in [H]$*. Let* $N_h = \tilde{O}\left((H-h+1)^2\alpha^2k^2H^2/\epsilon^2\right)$ *and* $N_h^{\#} = \alpha^2k^2N_h$*. Then, LR-EVI returns an* ϵ*-optimal Q function and policy with probability at least* $1-\delta$ *with a sample complexity of* $\tilde{O}\left((|S|+|A|)\alpha^2k^3H^5/\epsilon^2 + \alpha^4k^6H^5/\epsilon^2\right)$*.*

From Proposition 4, under Assumption 4 (low-rank transition kernel and expected rewards), the matrix corresponding to $[r_h + [P_h\hat{V}_{h+1}]]$ has rank d for any value function estimate $\hat{V}_{h+1}$. This is critical to the analysis of LR-EVI as it guarantees that the expectation of the matrix $\bar{Q}_h$ constructed from Empirical Value Iteration in Step 1(a) is low rank. This property is not satisfied by Assumptions 3 and 1, and as such the analysis for Theorem 9 does not extend to these weaker settings. Additionally, this property eliminates the need for constructing estimates with rollouts, which removes a factor of H in the sample complexity compared to LR-MCPI under Assumption 3.

Our sample complexity bounds depend on k, α, presuming that the algorithm uses some given set of (k,α)-anchor states and actions. When there may not be a domain expert to suggest anchor states and actions, we show in the next lemma that one can construct (k,α)-anchor states and actions with high probability by random sampling, where k and α scale with the incoherence and the bounded condition number of the target matrix.

Lemma 10. *Let* Q_h *be a rank* d*,* μ*-incoherent matrix with condition number* κ*. Let* $S^{\#}$ *and* $A^{\#}$ *be constructed randomly such that each state* s *is included in* $S^{\#}$ *with probability* $p_1 = \Theta(d\mu/\log(|S|))$*, and each action* a *is included in* $A^{\#}$ *with probability* $p_2 = \Theta(d\mu/\log(|A|))$*. With probability* $1 - O\left(H(|S| \wedge |A|)^{-10}\right)$*,* $S^{\#}$ *and* $A^{\#}$ *are* (k,α) *anchor states and actions for* Q_h *for* $k = \tilde{O}(\mu d)$ *and* $\alpha = O(\kappa)$*.*

Lemma 10 asserts that without a priori knowledge, one can find a set of $\tilde{O}(\mu d, \kappa)$-anchor states and actions using the sampling subroutine defined in Section 6.2, given that the corresponding matrix is μ-incoherent with condition number κ.

Comparison to Impossibility Result in Theorem 2. Recall that Theorem 2 establishes an exponential 4^H lower bound for learning a near-optimal policy in MDPs with low-rank Q^*. While the constructed MDP has a constant suboptimality gap, the lower bound does not contradict Theorem 7 which achieves a poly(H) sample complexity for LR-MCPI under a stronger generative model, i.e. after estimating the optimal action at step h, LR-MCPI can subsequently sample full trajectories from the estimate of the optimal policy, which would then include entry $(2,2)$, which was prohibited in the setup of Theorem 2. In contrast, LR-EVI does not admit an efficient sample complexity for the MDP constructed in Section 4, and one can show that it exhibits exponential blowup in the estimation error due to an amplification of the estimation error in the terminal step when propagating the estimates backwards via value iteration. The MDP does not have a low rank transition kernel violating Assumption 4, as needed for Theorem 9.

7.1 Discussion of Optimality

Theorems 7, 8 and 9 show that under our various low rank assumptions, LR-MCPI and LR-EVI learn near-optimal polices in a sample efficient manner, decreasing the dependence of sample complexity on S and A from $|S||A|$ to $|S|+|A|$. Furthermore in Lemma 11 we establish a $d(|S|+|A|)H^3/\epsilon^2$ sample complexity lower bound for MDPs with low rank reward and transition kernel in the sense of Assumption 4 via minor modifications of existing lower bounds for tabular MDPs. Since Assumption 4 implies the optimal Q^* function is low rank, the same lower bound holds for

the latter setting. Comparing our results to the lower bound, it follows that the dependence on $|S|, |A|$, and ϵ in our sample complexity upper bound is minimax optimal.

Lemma 11. *For any algorithm, there exists an MDP $M = (S, A, P, R, H)$ with rank d reward R_h and transition kernel P_h for all $h \in [H]$ such that $\Omega\left(d(|S| + |A|)H^3/\epsilon^2\right)$ samples are needed to learn an ϵ-optimal policy with high probability.*

Proof. Existing lower bounds from [38] prove the necessity of $\Omega(d|S|H^3/\epsilon^2)$ samples to learn an ϵ-optimal policy with high probability for a time-homogeneous MDP with $|S|$ states and d actions. Replicating each action $|A|/d$ times results in an MDP $|A|$ actions and rank d reward functions and transition kernels, and this MDP is at least as hard as the original MDP. Repeating this construction with an MDP with d states and $|A|$ actions proves an $\Omega(d|A|H^3/\epsilon^2)$ sample complexity lower bound. Combining these two lower bounds proves the lemma. □

As an aside we also point out that previously shown lower bounds for linearly-realizable MDPs [50, 51] are not directly applicable to our setting, as the constructed instances therein need not have low-rank Q^* or transition kernels, and the size of their state space scales exponentially in d.

Our sample complexity bounds depend on k and α, the size and quality of the (k, α)-anchor sets. As stated in Lemma 10, we can construct a set of $\tilde{O}(\mu d, \kappa)$-anchor states and actions for any μ-incoherent matrix with condition number κ simply by randomly sampling a subset of state and action. The results presented in the table in Section 1 are obtained by substituting $k = \tilde{O}(d\mu)$ and $\alpha = O(\kappa)$ into the sample complexity bounds in Theorems 7, 8 and 9 and treating μ and κ as constants, as is standard in the matrix estimation literature, e.g., [1].

In the event that there is a domain expert who provides a set of (k, α)-anchor states and actions, then the sample complexity bound may be better by using the given set rather than randomly sampling if μ and κ are large. Note that k must be minimally at least d, and the quality of a given set of anchor states and actions depends on the smallest singular value associated to the anchor submatrix as reflected in α, which for poorly chosen anchor state and actions could scale with H.

In Theorems 7, 8 and 9, the cubic dependence on d is likely suboptimal, but this results from the suboptimal dependence on d in the corresponding entrywise error bounds in the matrix estimation literature [37, 1, 10]. Without knowledge of good anchor states/actions from a domain expert, the dependence on μ that arises from randomly sampling anchor states and actions is not surprising, as it also commonly arises in the classical matrix estimation literature under uniform sampling models. Any improvements in the matrix estimation literature on the dependence on d, μ would directly translate into improved bounds via our results.

Our dependence on the horizon H is fairly standard as it matches the dependence on H for vanilla Q-value iteration. There is a gap between the dependence on H in our upper bounds and the H^3 lower bound in Lemma 11, which is given for homogeneous MDPs. Our upper bound results allow for nonhomogenous rewards and transition kernels, which would likely increase the lower bound to H^4. Reducing the upper bounds to H^4 would likely require using the total variance technique from [38], which requires estimates of the variance of the policy at a given state-action pair. One can show that the variance of the Bellman operator is low rank under the strongest assumption of a low Tucker rank transition kernel, but the corresponding rank of the matrix of variances is $O(d^2)$. Hence, while it may be possible to adapt this variance technique to achieve the optimal dependence on H in our low-rank settings, doing so may incur a significantly worse dependence on d, i.e., d^6.

7.2 Proof Sketch

The analysis of LR-MCPI and LR-EVI are fairly similar, and involves first showing that upon each application of the matrix estimation subroutine stated in (7), the amplification of the entrywise error is bounded, as stated below in Lemma 12.

Lemma 12. *Let $S^\#$ and $A^\#$ be (k,α)-anchor states and actions for matrix Q_h. For all $(s,a) \in \Omega_h = (S^\# \times A) \cup (S \times A^\#) \setminus (S^\# \times A^\#)$, assume that $\hat{Q}_h(s,a)$ satisfies $|\hat{Q}_h(s,a) - Q_h(s,a)| \le \eta$, and for all $(s,a) \in S^\# \times A^\#$, assume that $\hat{Q}_h(s,a)$ satisfies $|\hat{Q}_h(s,a) - Q_h(s,a)| \le \eta^\#$. Then, for all $(s,a) \in S \times A$, the estimates $\bar{Q}_h(s,a)$ computed via (7) satisfy*

$$\left|\bar{Q}_h(s,a) - Q_h(s,a)\right| = O(\alpha k \eta + \alpha^2 k^2 \eta^\#).$$

Proof Sketch for Lemma 12 As our algorithm constructs $\hat{Q}_h(s,a)$ for $(s,a) \in \Omega_h$ via averaging over samples from the MDP, the condition $|\hat{Q}_h(s,a) - Q_h(s,a)| \le \eta$ is satisfied with high probability for $\eta = O((H-h)/\sqrt{N_{s,a,h}})$, shown via a simple application of Hoeffding's inequality. To prove Lemma 12, we show that the error is bounded by

$$\begin{aligned}
\left|\bar{Q}_h(s,a) - Q_h(s,a)\right| &\lesssim \left\|[\hat{Q}_h(S^\#, A^\#)]^\dagger\right\|_{op} \cdot \left\|\hat{Q}_h(S^\#, a)\hat{Q}_h(s, A^\#) - Q_h(S^\#, a)Q_h(s, A^\#)\right\|_F \\
&+ \left\|\left[\hat{Q}_h(S^\#, A^\#)\right]^\dagger - \left[Q_h(S^\#, A^\#)\right]^\dagger\right\|_{op} \cdot \left\|Q_h(S^\#, a)Q_h(s, A^\#)\right\|_F \\
&\lesssim \left(\frac{1}{\sigma_d(Q_h(S^\#, A^\#))}\right) \cdot k\|Q_h\|_\infty(2\eta + \eta^2) \\
&+ \left(\frac{\eta^\# k}{(\sigma_d(Q_h(S^\#, A^\#)))^2}\right) \cdot \|Q_h\|_\infty^2 k = O(\alpha k \eta + \alpha^2 k^2 \eta^\#).
\end{aligned}$$

The first inequality comes from an application of the triangle inequality and the definition of the operator norm since for any rank d matrix Q with (k,α)-anchor states and actions, for all $(s,a) \in S \times A$, $Q(s,a) = Q(s,A^\#)[Q(S^\#,A^\#)]^\dagger Q(S^\#,a)$. The operator norm terms are bounded using Weyl's inequality and a classic result from the perturbation of pseudoinverses. The other two terms are bounded by our assumption on $\hat{Q}_h$ and that the reward functions are bounded by one.

As η is the dominant error term as $\eta^\#$ is the error on the small anchor sub-matrix, $\{\bar{Q}_h(s,a)\}_{(s,a)\in S^\# \times A^\#}$ with size $\tilde{O}(k) \times \tilde{O}(k)$, the critical insight from Lemma 12 is that the amplification of the error due to matrix estimation is only a factor of αk, which is constant for a good choice of anchor states and actions. We set $N_{s,a,h}$ for each $(s,a) \in \Omega_h$ to guarantee $\alpha k \eta$ and $\alpha^2 k^2 \eta^\#$ are sufficiently small for a subsequent induction argument that shows the algorithm maintains near optimal estimates of Q^* and π^*. For each of the Theorems 7, 8, and 9, we will apply Lemma 12 to different choices of Q_h, chosen to guarantee that $\hat{Q}_h(s,a)$ is an unbiased estimate of Q_h. For Theorem 7, we choose $Q_h = Q_h^*$. For Theorem 8, we choose $Q_h = Q_h^{\hat{\pi}}$, where $\hat{\pi}$ is an ϵ-optimal policy. For Theorem 9, we choose $Q_h = r_h + [P_h \hat{V}_{h+1}]$, where $\hat{V}_{h+1}$ is the value function estimate for step $h+1$.

Choosing Q_h to be potentially distinct from Q_h^* is a simple yet critical distinction between our analysis and [37]. The analysis in [37] applies a bound similar to Lemma 12 with a choice of $Q_h = Q_h^*$. However, as $\hat{Q}_h$ will not be unbiased estimates of Q_h^*, the initial error η will contain a bias term that is then amplified exponentially in H when combined with an inductive argument for LR-EVI.

Proof Sketch for Lemma 10 To prove that the random sampling method presented in Section 6.2 finds $\tilde{O}(\mu d, \kappa)$-anchor states and actions with high probability, let us denote the singular value

decomposition of matrix Q_h with $U\Sigma V^T$. For a randomly sampled set of anchor states and actions $S^\#$ and $A^\#$, let $\tilde{U}$ and $\tilde{V}$ denote the submatrices of U and V limited to $S^\#$ and $A^\#$, such that the anchor submatrix $Q_h(S^\#, A^\#)$ is given by $\tilde{U}\Sigma\tilde{V}^T$. By the matrix Bernstein inequality [43], when rows and columns are sampled uniformly with probability $p_1 = \Theta(d\mu/\log(|S|)), p_2 = \Theta(d\mu/\log(|A|))$, the columns of $\tilde{U}$ and $\tilde{V}$ are nearly orthogonal. In particular, with high probability

$$\|p_1^{-1}\tilde{U}^T\tilde{U} - I_{d\times d}\|_{op} \leq \frac{1}{2} \quad \text{and} \quad \|p_2^{-1}\tilde{V}^T\tilde{V} - I_{d\times d}\|_{op} \leq \frac{1}{2},$$

implying that the anchor submatrix is rank d. By an application of the singular value version of the Courant-Fischer minimax theorem [21], we can relate $\sigma_d(Q_h(S^\#, A^\#))$ to $\sigma_d(Q_h)$ to show that

$$\alpha = \max_h \|Q_h\|_\infty/\sigma_d(Q_h(S_h^\#, A_h^\#)) = O(\kappa).$$

Inductive Argument for Main Theorems. The final step is to use the error analysis of each iteration in an inductive argument that argues the estimated policy at each step is near optimal. As the induction argument is similar across all three theorems, we present the inductive argument for Theorem 8, and refer readers to the Appendix for the full proofs of all the theorems. For Theorem 8, the induction step is that if $\hat{\pi}_{H-t+1}$ is $t\epsilon/H$-optimal, then for time step $H-t$, the policy found with LR-MCPI, $\hat{\pi}_{H-t}$, is $(t+1)\epsilon/H$-optimal. We then induct backwards across horizon, i.e. $t \in \{1, \dots H\}$.

To show the induction step, first we argue that by Hoeffding's inequality, for $(s,a) \in \Omega_{H-t}$, with high probability $|\hat{Q}_{H-t} - Q^{\hat{\pi}}_{H-t}| = O(\epsilon/\alpha^2k^2H)$ for $N_{H-t} = \tilde{O}\left((t+1)^2\alpha^2k^2H^2/\epsilon^2\right), N^\#_{H-t} = \alpha^2k^2N_{H-t}$. It is critical that $\hat{Q}_{H-t}$ are indeed unbiased estimates of $Q^{\hat{\pi}}_{H-t}$ as the estimate is constructed via Monte Carlo rollouts. By Assumption 3 and the inductive hypothesis, $Q^{\hat{\pi}}_{H-t}$ is low rank, such that by an application of Lemma 12, it follows that for all $(s,a) \in S \times A$, $|\bar{Q}_{H-t}(s,a) - Q^{\hat{\pi}}_{H-t}(s,a)| \leq \epsilon/2H$ for the appropriate choice of N_{H-t}. Finally we argue that assuming the inductive hypothesis, choosing greedily according to $\bar{Q}_{H-t}$ results in a $(t+1)\epsilon/H$-optimal policy. For some state s, we denote $a^* = \pi^*_{H-t}(s)$ and $\hat{a} = \hat{\pi}_{H-t}(s) = \max_a \bar{Q}_{H+t}(s,a)$. The final induction step is shown via

$$\begin{aligned}
V^*_{H-t}(s) - V^{\hat{\pi}}_{H-t}(s) &= Q^*_{H-t}(s,a^*) - \bar{Q}_{H-t}(s,\hat{a}) + \bar{Q}_{H-t}(s,\hat{a}) - Q^{\hat{\pi}}_{H-t}(s,\hat{a}) \\
&\leq |Q^*_{H-t}(s,a^*) - Q^{\hat{\pi}}_{H-t}(s,a^*)| + |Q^{\hat{\pi}}_{H-t}(s,a^*) - \bar{Q}_{H-t}(s,a^*)| + |\bar{Q}_{H-t}(s,\hat{a}) - Q^{\hat{\pi}}_{H-t}(s,\hat{a})| \\
&\leq \max_{s'}(V^*_{H-t+1}(s') - V^{\hat{\pi}}_{H-t+1}(s')) + \frac{\epsilon}{2H} + \frac{\epsilon}{2H},
\end{aligned}$$

where $\max_{s'}(V^*_{H-t+1}(s') - V^{\hat{\pi}}_{H-t+1}(s')) \leq t\epsilon/H$ from the induction hypothesis.

The proof of Theorem 7 involves a similar inductive argument except that given the stronger suboptimality gap assumption, we guarantee that $\hat{\pi}_h$ is an exactly optimal policy with high probability. This removes the linear growth in the error across the horizon that arises in Theorem 8, enabling us to reduce N_h by H^2. The proof of Theorem 9 also involves a similar inductive argument, but under Assumption 4, we additionally show that at each time step $Q'_{H-t} = r_{H-t} + [P_{H-t}\hat{V}_{H-t+1}]$, the expected value of $\hat{Q}_{H-t}$ for LR-EVI, is close to not only Q^*_{H-t} but also $Q^{\hat{\pi}}_{H-t}$, which ensures that LR-EVI not only recovers an ϵ-optimal Q function, but also an ϵ-optimal policy.

Sample Complexity Calculation. The sample complexity of LR-MCPI is given by $\sum_h (H-h)(N_h|\Omega_h| + N^\#_h k^2)$, and the sample complexity of LR-EVI is given by $\sum_h (N_h|\Omega_h| + N^\#_h k^2)$. The set $|\Omega_h|$ scales as $O(k|S| + k|A|)$, where $k = \tilde{O}(\mu d)$ when the anchor states and actions are sampled randomly. The final sample complexity bounds result from substituting the choices of N_h and $N^\#_h$ as specified in the statements of Theorems 7, 8, and 9 into the summation.

7.3 Extension to Approximately Low-Rank MDPs

Our sample complexity results rely on either Q_h^*, Q_h^π, or $[r_h + P_h\hat{V}_{h+1}]$ having rank d, which may only be approximately satisfied. Furthermore, our algorithms require knowledge of the rank of those matrices, which may not be feasible to assume in practice. Hence, we extend our results under the low-rank reward and low Tucker rank transition kernel setting (Assumption 4) to a (d, ξ_R, ξ_P)-approximate low-rank MDP.

Assumption 5 ((d, ξ_R, ξ_P)-Approximate Low-rank MDP)**.** *An MDP specified by (S, A, P, R, H) is a (d, ξ_R, ξ_P)-approximate low-rank MDP if for all $h \in [H]$, there exists a rank d matrix $r_{h,d}$ and a low Tucker rank transition kernel $P_{h,d}$ with Tucker rank either $(|S|, d, |A|)$ or $(|S|, |S|, d)$, such that $\forall h$,*

$$\max_{(s,a)\in S\times A} |r_h(s,a) - r_{h,d}(s,a)| \le \xi_R \qquad \text{and} \qquad \sup_{(s,a)\times A} 2d_{\mathrm{TV}}(P_h(\cdot|s,a), P_{h,d}(\cdot|s,a)) \le \xi_P, \tag{8}$$

where d_{TV} is the total variation distance.

Assumption 8 extends the exact low-rank Assumption 4, where ξ_R is the entrywise low-rank approximation error of the reward function, and ξ_P is the low-rank approximation error of the transition kernel in total variation distance. For small values of ξ_R and ξ_P, the MDP can be approximated well by a rank d MDP, and subsequently, it follows that for any estimate of the future value function, $r_h + [P_h\hat{V}_{h+1}]$ is close to a corresponding rank d approximation.

Proposition 13. *Consider a (d, ξ_R, ξ_P)-approximate low-rank MDP with ξ_R, $\xi_P, r_{h,d}$, and $P_{h,d}$ as defined in Assumption 8, with respect to the low rank approximation. For all $h \in [H]$ and any $\hat{V}_{h+1}$,*

$$\left\| [r_{h,d} + P_{h,d}\hat{V}_{h+1}] - [r_h + P_h\hat{V}_{h+1}] \right\|_\infty \le \xi_R + (H-h)\xi_P.$$

Theorem 14 shows that LR-EVI with the matrix estimation routine defined in Section 6.2 is robust with respect to the low rank approximation error.

Theorem 14. *Assume we have a (d, ξ_R, ξ_P)-approximate low-rank MDP (Assumption 8) where $r_{h,d}$ and $P_{h,d}$ refer to the corresponding low rank approximations for the reward function and transition kernel. For all $h \in [H]$, let $S_h^\#, A_h^\#$ be (k, α)-anchor states and actions for $Q'_{h,d} = [r_{h,d} + P_{h,d}\hat{V}_{h+1}]$, where $\hat{V}_{h+1}$ is the learned value function from Low Rank Empirical Value iteration. Let $N_{H-t} = \tilde{O}\left(k^2\alpha^2H^4/\epsilon^2\right), N_{H-t}^\# = \alpha^2k^2N_{H-t}$ for all $t \in \{0, \dots, H-1\}$. Then LR-EVI returns an $\left(\epsilon + \tilde{O}\left(k^2\alpha^2\left(\xi_R H + \xi_P H^2\right)\right)\right)$-optimal policy with probability at least $1-\delta$ with a sample complexity of $\tilde{O}\left(k^3\alpha^2(|S|+|A|)H^5/\epsilon^2 + k^6\alpha^4H^5/\epsilon^2\right)$.*

The proof of this theorem (see Appendix J) follows the same steps as the proof of Theorem 9 but additionally accounts for the low rank approximation error using applications of Proposition 13. Proposition 13 is first used to bound the error between $\hat{Q}_h(s,a)$ and $Q'_{h,d}(s,a)$ for $(s,a) \in \Omega_h$. Second, the proposition is used to bound the second term in the below inequality which controls the error of our estimate relative to Q_h^* and $Q_h^{\hat{\pi}}$:

$$|\hat{Q}_h(s,a) - Q_h^*(s,a)| \le |\hat{Q}_h(s,a) - Q'_{h,d}(s,a)| + |Q'_{h,d}(s,a) - Q'_h(s,a)| + |Q'_h(s,a) - Q_h^*(s,a)|$$

for all $(s,a) \in S \times A$ where $Q'_h = r_h + P_h\hat{V}_{h+1}$ and $Q'_{h,d} = r_{h,d} + P_{h,d}\hat{V}_{h+1}$.

Theorem 14 shows that in the approximate rank setting, the error of the policy our algorithm finds is additive with respect to the approximation error while remaining sample efficient. If $Q'_{h,d}$ is μ-incoherent with condition number κ, one can use the result of Lemma 10 to find $\tilde{O}(\mu d, \kappa)$-anchor states and actions without a priori/domain knowledge.

8 Experiments

We empirically compare the performance of combining low-rank matrix estimation with empirical value iteration and Monte Carlo policy iteration on a tabular version of the Oil Discovery problem [39]. Our results also join other empirical works that show the benefit of using low-rank variants of RL algorithms on stochastic control problems [56, 34, 33].

Experimental Setup: We formulate this problem as a finite-horizon MDP, where the state and action spaces are both $\{0, 1, \ldots, D\}$ for $D = 399$, and the horizon length $H = 10$. The learner's goal is to locate the oil deposits over a 1 dimensional space, where the target location $l_h = \texttt{round}(400(1 - \frac{1}{h}))$ changes with h to make the learning task more difficult. At step h the learner receives a reward $f_h(s)$ that depends on how close the learner is to the oil deposit at l_h, perturbed by a zero-mean Gausian noise with variance $\sigma_h^2(s, a) = (0.5 + a/400)^2/10$. The action a chosen indicates what state the learner attempts to move to next, and the learner additionally pays a transportation cost proportional to the distance between s and a, denoted by $c(s, a)$. As a result the reward function is

$$r_h(s, a) = f_h(s) - c(s, a) + \mathcal{N}(0, \sigma_h^2(s, a)),$$

where we choose $f_h(s)$ and $c(s, a)$ according to

$$f_h(s) = 1 - \frac{1}{4}\left\lceil \frac{4}{D} \max\left(0, |s - l_h| - 20\right)\right\rceil \quad \text{and} \quad c(s, a) = 0.01 \times \texttt{round}\left(\frac{|s - a|}{100}\right),$$

where $\texttt{round}(s)$ rounds s to the nearest integer. $c(s, a)$ is discretized to take on only 5 distinct values, but the level sets of $c(s, a)$ are diagonal bands, such that $c(s, a)$ is in fact full rank. However, the stable rank of $c(s, a)$, as defined by $\|c(s, a)\|_F^2/\|c(s, a)\|_*^2$ is only 1.46, which implies that $c(s, a)$ is close to a low-rank matrix [36]. See Appendix C.1 for further discussion about $c(s, a)$.

The learner's intended movements are perturbed, resulting in the following transition kernel:

$$\mathbb{P}_h(s'|s, a) = \max\{0, \min\{D, \delta_a + \text{Unif}(-C_h, C_h)\}\}$$

where $\text{Unif}(-C_h, C_h)$ denotes the discrete uniform distribution over $\{-C_h, -C_h + 1, \ldots, C_h\}$ and $C_h = 4(H - h + 1)$ determines the amount of noise in the transitions. Since $\mathbb{E}[\mathbb{P}_h(s'|s, a)]$ only depends on the time step h and action a, it follows that the rows of $\mathbb{E}[\mathbb{P}_h(s'| :, :)]$ are the same and the rank of $\mathbb{E}[\mathbb{P}_h(s'| :, :)]$ is one. Hence, the transition kernel has Tucker rank $(|S|, 1, |A|)$.

Because the reward function is approximately low-rank and the transition kernel has low Tucker rank, it follows that this MDP is approximately low rank (satisfying Assumption 5). See Appendix C.2 for a visualization of Q_1^* and more discussion on the rank of Q_h^*.

Algorithms: We compare LR-EVI, LR-EVI + `SI`, LR-MCPI, and LR-MCPI + `SI` with empirical value iteration (EVI) and Monte Carlo policy iteration (MCPI). Recall that LR-EVI + `SI` is essentially the same as LR-EVI but uses `Soft-Impute` from the `fancyimpute` package [35] for the matrix estimation method, whereas LR-EVI uses the matrix estimation algorithm presented in section 6.2. The observation set, i.e. Ω_h, that is used for `Soft-Impute` is a Bernoulli sampled subset of entries where the probability of including each entry is denoted p_{SI}. Equivalently LR-MCPI + `SI` is the same as LR-MCPI except that it uses `Soft-Impute` with Bernoulli sampled $|\Omega_h|$. The vanilla EVI (resp., MCPI) refers to our algorithm using option (a) (resp., option (b)) for Step 1 without the matrix estimation step, setting $\Omega_h = S \times A$ for all $h \in [H]$ and change Step 2 to be $\bar{Q}_h = \hat{Q}_h$.

To empirically validate the performance of the algorithms, for a fixed sample budget $\overline{N}$, we compare the max entrywise error of $\bar{Q}_1$ of all the algorithms. We test five different allocation

schemes on how to distribute the $\overline{N}$ samples across the time steps to determine $N_{s,a,h}$ and use the best one for each algorithm. We ensure that an equal number of samples are allocated to each state-action pair. As $\overline{N}$ may not be divisible by D^2, the true samples used is within D^2 of $\overline{N}$ due to rounding. We show that LR-EVI and LR-MCPI are robust to $p = p_S = p_A$ as both algorithms perform well for a range of values of p, and it suffices to choose p to be small. We perform a grid search to determine p_{SI} for each value of $\overline{N}$, choosing the best performing parameter for each. See Appendix C.3 for the details on how we chose and set the hyperparameters of the algorithms.

Results: For each value of $\overline{N} \in [10^6, 10^7, 10^8, 10^9]$, we run each of the above algorithms 10 times. Figure 1 shows the average ℓ_∞ error of $\bar{Q}_1$ across the 10 simulations, along with error bars whose height indicates one standard deviation above and below the mean. Note that for vanilla EVI to produce an estimate, it requires one sample per (s, a, h), which already requires 1.6×10^6 samples. For vanilla MCPI to produce an estimate, it requires one trajectory per (s, a, h) of length $H - h + 1$, which requires 8.8×10^6 one-step samples. As a result, there is no bar depicted for either EVI or MCPI for $\overline{N} = 10^6$, as both algorithms require more than 10^6 samples to even produce any estimate.

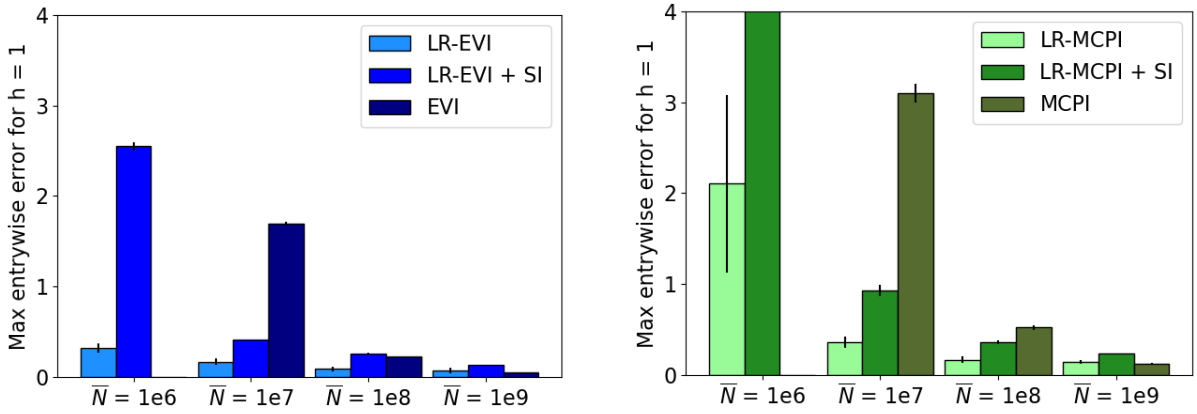


Figure 1: Max entrywise error of $\bar{Q}_1$ vs. sample budget for LR-EVI, LR-EVI + `Soft Impute`, empirical value iteration and LR-MCPI, LR-MCPI + `Soft-Impute`, Monte Carlo policy iteration at $h = 1$. Note that the optimal Q_1^* function ranges in value from roughly 8.3 to 9.6, such that 0.8 error would be roughly 10% error.

For $\overline{N} = 10^6$, the error bar for LR-MCPI + `SI` has a height of 11.5 but is trimmed to align the y-axis in both graphs. Figure 1 shows that when the sample budget is small ($\overline{N} = 10^6$), the low rank RL algorithms can still produce reasonable estimates even when there are not sufficient samples to even run the vanilla RL algorithms, i.e., less than 8.8×10^6 one-step samples. Our chosen MDP is also not strictly low rank, but only approximately low rank, thus our results validate that our algorithms are not sensitive to the exact rank, as they perform very well on this approximately low rank MDP as well. The Monte Carlo Policy Iteration variants seem to require more samples to achieve the same performance relative to Empirical Value Iteration variants. This is expected as the sample complexity of MCPI is multiplied by H due to sampling entire trajectories rather than one step samples. The MDP in this illustration is well-behaved for LR-EVI as it has a low rank transition kernel, but the practical benefit of LR-MCPI is that it is more robust to MDPs that may not have low rank structure in the transition kernel, as exhibited by the MDP constructed in Section 4.

We also compare the performance of EVI and MCPI and their low-rank variants on the Double Integrator, a stochastic control problem, see Appendix D for full details. The results from the Double Integrator simulations also show the benefit of the low-rank methods when the sample budget is

small; LR-MPCI produces a reasonable estimate of Q_1^* even when there are not sufficient samples to run MCPI. However, LR-EVI and LR-MPCI are sensitive to the choice of matrix estimation method, so in practice, one should carefully tune the matrix estimation methods' hyperparameters given computational limits on storage and runtime. When the sample budget is large, the low-rank methods lose their advantage and may even perform worse than tabular variants.

9 Conclusion

In this work, we prove novel sample complexity bounds using matrix estimation methods for MDPs with long time horizons without knowledge of special anchor states and actions, showing that incorporating matrix estimation methods into reinforcement learning algorithms can significantly improve the sample complexity of learning a near-optimal action-value function. Furthermore, we empirically verify the improved efficiency of incorporating the matrix estimation methods. We also provide a lower bound that highlights exploiting low rank structure in RL is significantly more challenging than the static matrix estimation counterpart without dynamics. While we show a gain from $|S||A|$ to $|S| + |A|$, the sample complexity may not be optimal with respect to d and H, which may be an interesting topic for future study. For example one could consider how to incorporate advanced techniques in existing tabular reinforcement learning literature that decrease the dependence on the time horizon into our low rank framework. While our results show the value of exploiting low-rank structure in reinforcement learning, the algorithms heavily rely on a generative model assumption, which may not always be realistic. Extensions to online reinforcement learning is an interesting and potentially impactful future direction.

References

[1] Emmanuel Abbe, Jianqing Fan, Kaizheng Wang, and Yiqiao Zhong. Entrywise eigenvector analysis of random matrices with low expected rank. *Annals of statistics*, 48(3):1452, 2020.

[2] Alekh Agarwal, Sham Kakade, Akshay Krishnamurthy, and Wen Sun. Flambe: Structural complexity and representation learning of low rank mdps. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 20095–20107. Curran Associates, Inc., 2020.

[3] Alekh Agarwal, Sham Kakade, and Lin F. Yang. Model-based reinforcement learning with a generative model is minimax optimal. In Jacob Abernethy and Shivani Agarwal, editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 67–83. PMLR, 09–12 Jul 2020.

[4] Martin Anthony and Peter L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, USA, 1st edition, 2009.

[5] Sanjeev Arora, Rong Ge, and Ankur Moitra. Learning topic models – going beyond svd. *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 1–10, 2012.

[6] Mohammad Gheshlaghi Azar, Rémi Munos, and Hilbert J. Kappen. On the sample complexity of reinforcement learning with a generative model. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, page 1707–1714, Madison, WI, USA, 2012. Omnipress.

[7] Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[8] George H. Chen and Jeremy C. Weiss. Survival-supervised topic modeling with anchor words: Characterizing pancreatitis outcomes, 2017.

[9] Yudong Chen and Yuejie Chi. Harnessing structures in big data via guaranteed low-rank matrix estimation: Recent theory and fast algorithms via convex and nonconvex optimization. *IEEE Signal Processing Magazine*, 35(4):14–31, 2018.

[10] Yuxin Chen, Yuejie Chi, Jianqing Fan, Cong Ma, and Yuling Yan. Noisy matrix completion: Understanding statistical guarantees for convex relaxation via nonconvex optimization. *SIAM Journal on Optimization*, 30:3098–3121, 01 2020.

[11] Christoph Dann and Emma Brunskill. Sample complexity of episodic fixed-horizon reinforcement learning. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[12] Mark A. Davenport and Justin Romberg. An overview of low-rank matrix recovery from incomplete observations. *IEEE Journal of Selected Topics in Signal Processing*, 10(4):608–622, 2016.

[13] Arthur Delarue, Ross Anderson, and Christian Tjandraatmadja. Reinforcement learning with combinatorial actions: An application to vehicle routing. In *NeurIPS*, 2020.

[14] Lijun Ding and Yudong Chen. Leave-one-out approach for matrix completion: Primal and dual analysis. *IEEE Transactions on Information Theory*, 66(11):7274–7301, 2020.

[15] Kefan Dong, Jian Peng, Yining Wang, and Yuan Zhou. Root-n-regret for learning in markov decision processes with function approximation and low bellman rank. In Jacob Abernethy and Shivani Agarwal, editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 1554–1557. PMLR, 09–12 Jul 2020.

[16] Simon Du, Sham Kakade, Jason Lee, Shachar Lovett, Gaurav Mahajan, Wen Sun, and Ruosong Wang. Bilinear classes: A structural framework for provable generalization in rl. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 2826–2836. PMLR, 18–24 Jul 2021.

[17] Tony Duan. Lightweight python library for in-memory matrix completion., 2020.

[18] S.A. Goreinov, E.E. Tyrtyshnikov, and N.L. Zamarashkin. A theory of pseudoskeleton approximations. *Linear Algebra and its Applications*, 261(1):1–21, 1997.

[19] Botao Hao, Yaqi Duan, Tor Lattimore, Csaba Szepesvari, and Mengdi Wang. Sparse feature selection makes batch reinforcement learning more sample efficient. In *ICML*, 2021.

[20] William B. Haskell, Rahul Jain, and Dileep Kalathil. Empirical dynamic programming. *Mathematics of Operations Research*, 41(2):402–429, 2016.

[21] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[22] Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E. Schapire. Contextual decision processes with low Bellman rank are PAC-learnable. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1704–1713. PMLR, 06–11 Aug 2017.

[23] Chi Jin, Qinghua Liu, and Sobhan Miryoosefi. Bellman eluder dimension: New rich classes of RL problems, and sample-efficient algorithms. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[24] Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. In Jacob Abernethy and Shivani Agarwal, editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 2137–2143. PMLR, 09–12 Jul 2020.

[25] Michael Kearns and Satinder Singh. Finite-sample convergence rates for q-learning and indirect algorithms. In M. Kearns, S. Solla, and D. Cohn, editors, *Advances in Neural Information Processing Systems*, volume 11. MIT Press, 1999.

[26] Gen Li, Changxiao Cai, Yuxin Chen, Yuantao Gu, Yuting Wei, and Yuejie Chi. Is q-learning minimax optimal? a tight sample complexity analysis. *arXiv preprint arXiv:2102.06548*, 2021.

[27] Yuxi Li. Reinforcement learning applications. *arXiv preprint arXiv:1908.06973*, 2019.

[28] Osman Asif Malik and Stephen Becker. Low-rank tucker decomposition of large tensors using tensorsketch. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[29] Shie Mannor and John N. Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *J. Mach. Learn. Res.*, 5:623–648, dec 2004.

[30] Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *Journal of Machine Learning Research*, 11(80):2287–2322, 2010.

[31] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charlie Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

[32] Aditya Modi, Jinglin Chen, Akshay Krishnamurthy, Nan Jiang, and Alekh Agarwal. Model-free representation learning and exploration in low-rank mdps. *CoRR*, abs/2102.07035, 2021.

[33] Sergio Rozada and Antonio G. Marques. Tensor and matrix low-rank value-function approximation in reinforcement learning, 2022.

[34] Sergio Rozada, Victor Tenorio, and Antonio G. Marques. Low-rank state-action value-function approximation. In *2021 29th European Signal Processing Conference (EUSIPCO)*, pages 1471–1475, 2021.

[35] Alex Rubinsteyn and Sergey Feldman. fancyimpute: An imputation library for python, 2016.

[36] Mark Rudelson and Roman Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *J. ACM*, 54(4):21–es, jul 2007.

[37] Devavrat Shah, Dogyoon Song, Zhi Xu, and Yuzhe Yang. Sample efficient reinforcement learning via low-rank matrix estimation. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12092–12103. Curran Associates, Inc., 2020.

[38] Aaron Sidford, Mengdi Wang, Xian Wu, Lin Yang, and Yinyu Ye. Near-optimal time and sample complexities for solving markov decision processes with a generative model. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[39] Sean R. Sinclair, Tianyu Wang, Gauri Jain, Siddhartha Banerjee, and Christina Lee Yu. Adaptive discretization for model-based reinforcement learning, 2020.

[40] Satinder Singh and Richard Yee. An upper bound on the loss from approximate optimal-value functions. *Machine Learning*, 16, 10 1996.

[41] Wen Sun, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, and John Langford. Model-based rl in contextual decision processes: Pac bounds and exponential improvements over model-free approaches. In *COLT*, 2019.

[42] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

[43] Joel A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, Aug 2011.

[44] Madeleine Udell and Alex Townsend. Why are big data matrices approximately low rank? *SIAM Journal on Mathematics of Data Science*, 1(1):144–160, 2019.

[45] Masatoshi Uehara, Xuezhou Zhang, and Wen Sun. Representation learning for online and offline rl in low-rank mdps. *arXiv preprint arXiv:2110.04652*, 2021.

[46] Oriol Vinyals, Igor Babuschkin, Wojciech M. Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H. Choi, Richard Powell, Timo Ewalds, Petko Georgiev, Junhyuk Oh, Dan Horgan, Manuel Kroiss, Ivo Danihelka, Aja Huang, L. Sifre, Trevor Cai, John P. Agapiou, Max Jaderberg, Alexander Sasha Vezhnevets, Rémi Leblond, Tobias Pohlen, Valentin Dalibard, David Budden, Yury Sulsky, James Molloy, Tom Le Paine, Caglar Gulcehre, Ziyun Wang, Tobias Pfaff, Yuhuai Wu, Roman Ring, Dani Yogatama, Dario Wünsch, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy P. Lillicrap, Koray Kavukcuoglu, Demis Hassabis, Chris Apps, and David Silver. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, pages 1–5, 2019.

[47] M.J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019.

[48] Bingyan Wang, Yuling Yan, and Jianqing Fan. Sample-efficient reinforcement learning for linearly-parameterized mdps with a generative model. *Advances in Neural Information Processing Systems*, 34, 2021.

[49] Ruosong Wang, Dean P. Foster, and Sham M. Kakade. What are the statistical limits of offline rl with linear function approximation?, 2020.

[50] Yuanhao Wang, Ruosong Wang, and Sham Kakade. An exponential lower bound for linearly realizable mdp with constant suboptimality gap. *Advances in Neural Information Processing Systems*, 34, 2021.

[51] Gellért Weisz, Philip Amortila, and Csaba Szepesvári. Exponential lower bounds for planning in mdps with linearly-realizable optimal action-value functions. In *Algorithmic Learning Theory*, pages 1237–1264. PMLR, 2021.

[52] Yihong Wu. Lecture notes on: Information-theoretic methods for high-dimensional statistics, 2020.

[53] Kunhe Yang, Lin F. Yang, and Simon Shaolei Du. Q-learning with logarithmic regret. In *AISTATS*, 2021.

[54] Lin Yang and Mengdi Wang. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 10746–10756. PMLR, 13–18 Jul 2020.

[55] Lin F. Yang and Mengdi Wang. Sample-optimal parametric q-learning using linearly additive features. In *ICML*, 2019.

[56] Yuzhe Yang, Guo Zhang, Zhi Xu, and Dina Katabi. Harnessing structures for value-based planning and reinforcement learning. In *International Conference on Learning Representations*, 2020.

A Extensions

We extend our results to the continuous MDP setting and infinite-horizon discounted MDP setting. We also discuss the use of alternative matrix estimation subroutines. Each of these extensions are fairly minor technically, but we include them to illustrate the wider implications of the low rank framework.

A.1 Continuous State and Action Spaces

Our results in Theorems 8 and 9 can be extended to the continuous MDP setting where S and A are both continuous spaces. In particular, the action-value function obtained when running LR-EVI and LR-MCPI on a discretized version of the continuous MDP can be used to construct an ϵ-optimal action-value function for the continuous MDP, similar to the reduction used in [37]. We assume the same regularity conditions on the continuous MDP as used in [37].

Assumption 6 (MDP Regularity for Continuous MDPs [37])**.** *Assume the MDP satisfies*

- *(Compact Domain):* $S = [0,1]^n$, $A = [0,1]^n$,
- *(Lipschitz):* Q_h^* *is L-Lipschitz with respect to the one-product metric:*

$$|Q_h^*(s,a) - Q_h^*(s',a')| \leq L(\|s - s'\|_2 + \|a - a'\|_2) \quad \forall \ \ h \in [h].$$

We follow the same steps as in [37] to discretize the state and action spaces into β-nets (S^β and A^β, respectively), i.e. S^β is a set such that for all $s \in S$, there exists an $s' \in S^\beta$ where $|s' - s|_2 \leq \beta$. We next define the discretized MDP to be $M^\beta = (S^\beta, A^\beta, P^\beta, r, H)$ where P_h^β is defined as follows:

$$P_h^\beta(s'|s,a) = \int_{\{s'' \in S : |s'' - s'|_2 \leq \beta\}} P_h(s''|s,a) \mathrm{d}s''.$$

After discretizing the state and action spaces, LR-MCPI or LR-EVI is run on the discretized MDP. Our approach differs from the one from [37] because we only discretize the continuous sets once and then run the tabular algorithm while their algorithm changes the discretization error β at each iteration. To run LR-MCPI or LR-EVI on the discretized MDP, one needs to be able to sample transitions/rollouts from P_h^β instead of P_h. See Appendix K for details on how we exploit the generative model to obtain transitions/rollouts on M^β. The following lemma shows how the optimal Q function on M^β can be used to approximate Q^* of the original MDP with small enough β.

Lemma 15. *Let MDP* $M^\beta = (S^\beta, A^\beta, P^\beta, R, H)$ *be the discretized approximation to MDP* $M = (S, A, P, R, H)$ *where* S^β *and* A^β *are* β*-nets of* S *and* A*, respectively. Let* Q^* *and* Q^β *be the optimal* Q *functions of* M *and* M^β*, respectively. For any* $s \in S, a \in A$ *and* $s' \in S^\beta, a' \in A^\beta$ *such that* $\|s - s'\|_2 \leq \beta, \|a - a'\|_2 \leq \beta$ *and for all* $h \in [H]$*,*

$$|Q_h^*(s,a) - Q_h^\beta(s',a')| \leq 2L(H - h + 1)\beta, \quad |V_h^*(s,a) - V_h^\beta(s',a')| \leq 2L(H - h + 1)\beta.$$

If the transition kernels and reward functions of M^β are low rank, satisfying Assumption 4, then LR-EVI finds an ϵ-optimal Q_h function with an efficient number of samples. If M^β only satisfies Assumption 3, then LR-MCPI finds an ϵ-optimal Q_h function with an efficient number of samples. For sake of brevity, we present only the sample complexity bound of LR-EVI under Assumption 4. See Appendix K for the analogous result with LR-MCPI.

Theorem 16. *Let $Q_h^\beta = [r_h + [P_h \hat{V}_{h+1}]]_{(s,a)\in S^\beta \times A^\beta}$ where $\hat{V}_{h+1}$ is the value function learned when running LR-EVI on the discretized MDP M^β. Let Assumption 4 hold on M^β, and $S_h^\#, A_h^\#$ be (k,α)-anchor states and actions for Q_h^β for all $h \in [H]$. Then, the learned $\bar{Q}_h$ from LR-EVI can be used to construct an ϵ-optimal Q function with probability at least $1-\delta$ when $\beta = \epsilon/4LH$ and $N_{H-t} = \tilde{O}\left((t+1)^2 k^2 \alpha^2 H^2/\epsilon^2\right), N_{H-t}^\# = \tilde{O}\left((t+1)^2 k^4 \alpha^4 H^2/\epsilon^2\right)$ for all $t \in \{0,\dots,H-1\}$ with a sample complexity of $\tilde{O}\left(k^3\alpha^2 H^{n+5}/\epsilon^{n+2} Vol(B)\right)$, where B is the unit norm ball in $\mathbb{R}^n$.*

Theorem 16 shows that if the low-rank and matrix estimation assumptions hold on the discretized MDP, then one can use the learned Q_h estimate from LR-EVI to construct an ϵ-optimal estimate of Q function. Both algorithms are sample efficient (with respect to the dimension of the state and action spaces) as the bounds have a $1/\epsilon^{n+2}$ dependence instead of $1/\epsilon^{2n+2}$, which is minimax optimal without the low-rank assumption. Furthermore, if Q_h^β is μ-incoherent with condition number κ, one can use the result of Lemma 10 to find $\tilde{O}(\mu d, \kappa)$-anchor states and actions without a priori/domain knowledge. Using the finite-horizon version of Corollary 2 from [40], we can construct an $O(\epsilon H)$-optimal policy by defining a policy greedily with respect to $\bar{Q}$.

The proof of Theorem 16 follows from combining Theorem 9 with a covering number lemma to upper bound the size of the β-nets. β is chosen carefully to account for the error amplification with respect to H from Lemma 15 while ensuring that the algorithms use an efficient number of samples.

A.2 Infinite-Horizon Discounted MDPs

We consider the standard setup for infinite-horizon tabular MDPs, (S, A, P, R, γ), where S and A denote the finite state and action spaces. $R : S \times A \to \Delta([0,1])$ denotes the reward distribution, and use $r_h(s,a) = \mathbb{E}_{r\sim R(s,a)}[r]$ to denote the expected reward. P denotes the transition kernel, and $0 < \gamma < 1$ denotes the discount factor. The value and action-value function of following the policy π are defined as:

$$V^\pi(s) \coloneqq \mathbb{E}\left[\sum_{t=0}^\infty \gamma^t R_t \middle| s_0 = s\right], \quad Q^\pi(s,a) \coloneqq \mathbb{E}\left[\sum_{t=0}^\infty \gamma^t R_t \middle| s_0 = s, a_0 = a\right],$$

for $R_t \sim R(s_t, a_t), a_t \sim \pi(s_t)$, and $s_t \sim P(\cdot|s_{t-1}, a_{t-1})$. We define the optimal value function as $V^*(s) = \sup_\pi V^\pi(s)$ for all $s \in S$ and the optimal action-value function as $Q^*(s,a) = r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}[V^*(s')]$. Since the reward function is bounded, for any policy π, $Q^\pi(s,a), V^\pi(s) \le \frac{1}{1-\gamma}$ for all $(s,a) \in S \times A$. To use matrix estimation methods, we require the transition kernel to have low Tucker rank and the reward function to have shared latent factors, which is our strongest low-rank assumption (Assumption 4).

Assumption 7 (Low-rank Transition Kernels and Reward Functions (Infinite-horizon))**.** *The expected reward function has rank d, and the transition kernel P has Tucker rank $(|S|, |S|, d)$ or $(|S|, d, |A|)$, with shared latent factors. For the Tucker rank $(|S|, |S|, d)$ case, this means that there exists a $|S| \times |S| \times d$ tensor U, an $|A| \times d$ matrix V, and an $|S| \times d$ matrix W such that*

$$P(s'|s,a) = \textstyle\sum_{i=1}^d U(s', s, i) V(a, i) \quad \text{and} \quad r(s,a) = \textstyle\sum_{i=1}^d W(s,i) V(a,i).$$

For the Tucker rank $(|S|, d, |A|)$ case, this means that there exists a $|S| \times |A| \times d$ tensor V, an $|S| \times d$ matrix U, and an $|A| \times d$ matrix W such that

$$P(s'|s,a) = \textstyle\sum_{i=1}^d U(s, i) V(s', a, i) \quad \text{and} \quad r(s,a) = \textstyle\sum_{i=1}^d U(s,i) W(a,i).$$

Similar to the finite-horizon setting, this assumption implies that $r + \gamma[P\bar{V}]$ has low rank for any value function estimate.

Proposition 17. *For any MDP that satisfies Assumption 7, for any estimate of the value function, the rank of $r + \gamma[P\bar{V}]$ is upperbounded by d.*

The algorithm we consider that admits an efficient sample complexity is LR-EVI with the same matrix estimation method adapted for the infinite-horizon discounted setting, i.e., including the discount factor in the estimates and running Step 1, Step 2, and Step 3 for T iterations instead of recursing backwards through the horizon. We overload notation and let $\hat{Q}_i$ refer to the Q function estimated in the i-th iteration of the algorithm. The correctness result and sample complexity bound in this setting is as follows.

Theorem 18 (Correctness and Sample Complexity of LR-EVI under Assumption 7)**.** *Assume that for any ϵ-optimal value function $\bar{V}$, the matrix corresponding to $Q'_t = [r + [P\hat{V}_t]]$ has rank d (a consequence of Assumption 7) for all $t \in [T]$, and $S_t^{\#}, A_t^{\#}$ are (k, α)-anchor states and actions for $\hat{Q}'_t = [r + [P\hat{V}_t]]$, where $\hat{V}_t$ is the learned value function from LR-EVI at iteration t for all $t \in [T]$. Let $N_t = \tilde{O}\left(\alpha^2 k^2/\epsilon^2(1-\gamma)^4\right)$ and $N_t^{\#} = O(\alpha^2 k^2 N_t)$. Then, LR-EVI returns an ϵ-optimal Q function with probability at least $1-\delta$ with a sample complexity of $\tilde{O}\left(\frac{(|S|+|A|)\alpha^2 k^3}{\epsilon^2(1-\gamma)^4} + \frac{\alpha^4 k^6}{\epsilon^2(1-\gamma)^4}\right)$.*

Theorem 9 states that if the transition kernel has low Tucker rank, one can learn an ϵ-optimal Q function with sample complexity that scales with the sum of the sizes of the state and action space instead of the product. Furthermore, if Q'_t is μ-incoherent, then one can use Lemma 10 to find $\tilde{O}(\mu d, \kappa)$-anchor states and actions without domain knowledge, where κ is the condition number of Q'. To prove the correctness result in Theorem 9, we show that at each iteration the error of the Q function decreases with the following lemma.

Lemma 19. *Let $S_t^{\#}, A_t^{\#}$ and N_t be as defined as in Theorem 18, and let the estimate of the value function at step t satisfy $|\bar{V}_t - V^*|_\infty \leq B_t$. Ater one iteration of the algorithm, the resulting estimates of the value function and action-value function satisfy*

$$|\bar{Q}_{t+1} - Q^*|_\infty \leq \frac{(1+\gamma)B_t}{2}, \quad |\bar{V}_{t+1} - V^*|_\infty \leq \frac{(1+\gamma)B_t}{2}$$

with probability at least $1 - \frac{\delta}{T}$ for each $t \in [T]$.

Running the algorithm for a logarithmic number of times returns an ϵ-optimal Q function, which gives the sample complexity shown in Theorem 9.

A.3 Matrix Completion via Nuclear Norm Regularization

While all of the above results are stated for the variants of LR-MCPI and LR-EVI that use the matrix estimation algorithm as stated in Section 6.2, our results are not limited only to this specific choice of the matrix estimation algorithm. As briefly mentioned in Section 6.2, the analysis relies on using entrywise error bounds for the outputs of the matrix estimation algorithm. While the algorithm stated in Section 6.2 lends itself to explicit entrywise error bounds given its explicit form, it requires the non-standard sampling pattern associated to a set of anchor states and actions.

We show next that similar results can be derived for a different variation of LR-MCPI or LR-EVI that performs matrix estimation by solving the convex relaxation of the low-rank matrix completion problem. We utilize Theorem 1 from [10] to obtain entry-wise bounds on the matrix estimator.

Chen et al. [10] state that it is straightforward to extend their results to the rectangular matrix setting, but for ease of notation, they only consider square matrices. To directly apply the theorem, we let $|S| = |A| = n$ but note that it is easy to extend our results to $|S| \neq |A|$. Their analysis assumes data is gathered via a Bernoulli sampling model; i.e. each state-action pair is added to Ω_h with probability p, i.e., $\Omega_h = \{(s,a)|X_{(s,a)} = 1\}$ where $X_{(s,a)} \sim \text{Bernoulli}(p)$ for $(s,a) \in S \times A$.

The matrix estimator is the minimizer of a least-squares loss function with a nuclear norm regularizer, which is the convex relaxation of the low rank constraint. For the observed matrix $M_{ij} = M^*_{ij} + E_{ij}$, with M^* being the matrix we wish to recover and error matrix E, the formulation is

$$\min_{Z \in \mathbb{R}^{n \times n}} g(Z) \triangleq \frac{1}{2} \sum_{(i,j) \in \Omega_h} (Z_{ij} - M_{ij})^2 + \lambda \|Z\|_*, \tag{9}$$

with Ω_h constructed via Bernoulli sampling as mentioned above [10].

We next present the primary result that is needed from [10] for the readers' convenience. Assume that Ω is constructed with the Bernoulli sampling model and the error matrix $E = [E_{i,j}]$ is composed of i.i.d. zero-mean sub-Gaussian random variables with norm at most η.

Theorem 20 (Theorem 1 in [10])**.** *Let M^* have rank$-d$ and be μ-incoherent with condition number κ, where $d, \kappa \in O(1)$. Let $\lambda = C_\lambda n \sigma \sqrt{np}$ in Equation 9 for a large enough positive constant C_λ. Assume that $n^2 p \geq C \mu^2 n \log^3 n$ and $\sigma \leq c \sqrt{\frac{np}{\mu^3 \log n}} \|M^*\|_\infty$ for some sufficiently large constant $C > 0$ and small constant $c > 0$. Then with probability $1 - O(n^{-3})$, any minimizer Z_{cvx} of Equation 9 satisfies*

$$\|Z_{cvx} - M^*\|_\infty \leq C_{cvx} \frac{\sigma}{\sigma_d(M^*)} \sqrt{\frac{\mu n \log n}{p}} \|M^*\|_\infty$$

for some constant $C_{cvx} > 0$.

Applying Theorem 20 into our analyses for LR-MCPI and LR-EVI gives us the necessary error bounds to prove the desired linear $|S| + |A|$ sample complexities for LR-MCPI and LR-EVI with Ω_h generated according to the Bernoulli sampling model and

$$\texttt{ME}\left(\{\hat{Q}_h(s,a)\}_{(s,a) \in \Omega_h}\right) \leftarrow \text{CvxSolver}\left(\min_{Q \in \mathbb{R}^{|S| \times |A|}} g(Q) \triangleq \frac{1}{2} \sum_{(s,a) \in \Omega_h} (Q(s,a) - (\hat{Q}_h(s,a)))^2 + \lambda \|Q\|_*\right).$$

We state only the result for LR-EVI under Assumption 4 (low-rank reward function and low Tucker rank transition kernel). The modifications to the theorems and proofs to show the analogous result for LR-MCPI under Assumption 3 are essentially the same.

Theorem 21. *Let $p_h = \mu^3 d^2 \kappa^2 H^4 C^2_{cvx} \log(n)/\epsilon^2 n$. Assume that for any ϵ-optimal value function $\hat{V}_{h+1}$, $Q'_h = [r_h + [P_h \hat{V}_{h+1}]]$ has rank d (Assumption 4), is μ-incoherent, and has condition number κ for all $h \in [H]$. Then, the learned policy from the algorithm specified above is ϵ-optimal with probability at least $1 - O\left(Hn^{-3} + \exp\left(-\mu^3 d^2 \kappa^2 H^4 n \log(n)/\epsilon^2\right)\right)$. Furthermore, the number of samples used is upper bounded by $\tilde{O}\left(\mu^3 H^5 n/\epsilon^2\right)$ with the same probability.*

The proof of Theorem 21 follows the same argument as the proof of Theorem 9 but uses Theorem 20 to control the error amplification from the matrix estimation method. Similar to our main results, using this matrix estimation method as a subroutine reduces the sample complexity's dependence on $|S|$ and $|A|$ from $|S||A|$ to $|S|+|A|$. This theorem provides a potential explanation for the successful experimental results in [56], and answers an open question posed in [37]; it guarantees that using existing matrix estimation methods based on convex problems as a subroutine in traditional value iteration has significantly better sample complexity compared to vanilla value iteration when finding ϵ-optimal action-value functions.

B Example Illustrating Assumption 3 (Low Rank Q functions for Near Optimal Policies)

Assumption 3 states that the ϵ-optimal policies π have associated Q functions that are low rank. At first glance, it might be unclear if this Assumption can be satisfied without requiring the stronger conditions in Assumption 4 of low rank rewards and low Tucker rank transition kernels. In this section, we present an MDP (S, A, P, R, H) in the reward maximization setting with all ϵ-optimal policies π (ϵ-optimal π for all $s \in S$ and $h \in [H]$) having low-rank Q^π without the transition kernel having low Tucker rank. Specifically, we upperbound the rank of Q^π with a function of ϵ and the size of the state/action space. where Π_ϵ is the policy class containing all ϵ-optimal deterministic policies. With the following example, we show that there exists an MDP with a non-trivial relationship between d_ϵ and $\epsilon, |S|$, and $|A|$. We now present the H-step MDP that exhibits this property. Let $S = A = 0 \cup [m]$, and the reward function be $r_h(s, a) = 0$ for all $(s, a, h) \in S \times A \times [H-1]$ and $r_H(s, a) = 1 - \left(\frac{sa}{(m+1)^2}\right)^{1/2}$ for all $(s, a) \in S \times A$.

For all $h \in [H-1]$, the transition kernel is

$$P_h(0|s,a) = \begin{cases} 0, & \text{if } s = a, \\ 1, & \text{otherwise,} \end{cases} \qquad P_h(s'|s,a) = \begin{cases} 1, & \text{if } s' = s = a, \\ 0, & \text{otherwise,} \end{cases}$$

for $s' \in \{1, \ldots, m\}$. We note that

$$P_h(0|\cdot,\cdot) = \begin{bmatrix} \mathbf{0} & 1 & 1 & \ldots & 1 \\ 1 & \mathbf{0} & 1 & \ldots & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \ldots & \mathbf{0} & 1 \\ 1 & 1 & \ldots & 1 & \mathbf{0} \end{bmatrix},$$

and for $s' \in [m]$, $P_h(s'|s, a) = E^{s'}$ is an all-zero matrix with the s'-th diagonal entry equal to one, so the transition kernels do not have low Tucker rank. Next, we prove the main result of this section, which upper bounds the rank of the Q_h functions of ϵ-optimal policies.

We remark that at time step 2, selecting action 0 is always optimal, regardless of the state.

Lemma 22. *Let π be an ϵ-optimal policy, that is $V_h^*(s) - V_h^\pi(s) \le \epsilon$ for all $(s, h) \in S \times [H]$. Then, $rank(Q_H^\pi) = 2$, and $rank(Q_h^\pi) \le 1 + \lfloor \epsilon^2 (m+1)^2 \rfloor$ for all $h \in [H-1]$.*

Proof of Lemma 22. Let π be an ϵ-optimal policy. We first show that $Q_H^\pi = r_2$ is a matrix with rank 2. By construction, the first two rows of Q_H^π are:

$$(Q_2^\pi)_1 = [1, 1, 1, \ldots, 1], \quad (Q_2^\pi)_2 = \left[1, 1 - \left(\frac{1}{(m+1)^2}\right)^{1/2}, 1 - \left(\frac{2}{(m+1)^2}\right)^{1/2}, \ldots, 1 - \left(\frac{m+1}{(m+1)^2}\right)^{1/2}\right],$$

and for any $i \in \{3, \ldots m+1\}$, the i-th row of Q_2^π is

$$(Q_2^\pi)_i = \left[1, 1 - \left(\frac{i}{(m+1)^2}\right)^{1/2}, 1 - \left(\frac{2i}{(m+1)^2}\right)^{1/2}, \ldots, 1 - \left(\frac{(m+1)i}{(m+1)^2}\right)^{1/2}\right].$$

Hence, $(Q_2^\pi)_i = (1 - i^{1/2})(Q_2^\pi)_1 + i^{1/2}(Q_2^\pi)_2$, and $rank(Q_2^\pi) = 2$.

Let $h \in [H]$, to bound the rank of Q_h^π, we first note that for all $s \in S$, $|V_H^\pi(s) - 1| = \left(\frac{s\pi(s)}{(m+1)^2}\right)^{1/2} \leq \epsilon$ since π is ϵ-optimal. It follows that $\pi(s) < \frac{\epsilon^2(m+1)^2}{s}$, and if $\frac{\epsilon^2(m+1)^2}{s} < 1$, $\pi(s)$ must equal 0, which is the optimal action. Hence, there are at most $s \leq \epsilon^2(m+1)^2$ number of states in which π can deviate from the optimal policy. The value function of an ϵ-optimal policy is

$$V_H^\pi(s) = \begin{cases} 1, & \text{if } s = 0, \\ 1, & \text{if } s > \epsilon^2(m+1)^2, \\ 1 - \left(\frac{s\pi(s)}{(m+1)^2}\right)^{1/2}, & \text{otherwise.} \end{cases}$$

Since π is ϵ-optimal, we have $1 - V_{h+1}^\pi(s) \leq \epsilon$ for all $s \in S$. Due to the construction of the dynamics, if one starts at state s at time step h, one will be at either state s (choosing action $\pi(s) = s$ at each time step) or state 0 (taking any other sequence of actions). Thus, $V_{h+1}^\pi(s) = V_H^\pi(s)$ or $V_{h+1}^\pi(s) = 1$ depending on the sequence of action. It follows that $V_{h+1}^\pi(s) \geq V_H^\pi(s)$. It follows that if $s = 0$ or $s > \epsilon^2(m+1)^2$, $V_{h+1}^\pi(s) = 1$. Otherwise, $V_{h+1}^\pi(s) \geq \left(\frac{s\pi(s)}{(m+1)^2}\right)^{1/2}$ for $s \leq \lfloor \epsilon^2(m+1)^2 \rfloor$.

We next compute the Q function at the time step h to show that we can upperbound the rank of Q_h by the number of states that π_h deviates from the optimal policy. Specifically, for each $s \leq \lfloor \epsilon^2(m+1)^2 \rfloor$, let $\pi_h(s) = s$ for $h \in [H]$. It follows that

$$\begin{aligned} Q_h^\pi(s,a) &= r_h(s,a) + \sum_{s'=0}^{m} P_h(s'|s,a) V_{h+1}^\pi(s') \\ &= 0 + P_h(0|s,a) V_{h+1}^\pi(0) + \sum_{s'=\lfloor \epsilon^2(m+1)^2 \rfloor + 1}^{m} P_h(s'|s,a) V_{h+1}^\pi(s') + \sum_{s'=1}^{\lfloor \epsilon^2(m+1)^2 \rfloor} P_h(s'|s,a) V_{h+1}^\pi(s') \\ &= P_h(0|s,a) + \sum_{s'=\lfloor \epsilon^2(m+1)^2 \rfloor + 1}^{m} P_h(s'|s,a) + \sum_{s'=1}^{\lfloor \epsilon^2(m+1)^2 \rfloor} P_h(s'|s,a) \left(1 - \left(\frac{s\pi(s)}{(m+1)^2}\right)^{1/2}\right). \end{aligned}$$

In matrix form, it follows that

$$\begin{aligned} Q_h^\pi &\geq \begin{bmatrix} \mathbf{0} & 1 & 1 & \dots & 1 \\ 1 & \mathbf{0} & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \dots & \mathbf{0} & 1 \\ 1 & 1 & \dots & 1 & \mathbf{0} \end{bmatrix} + \sum_{s'=\lfloor \epsilon^2(m+1)^2 \rfloor + 1}^{m} E^{s'} + \sum_{s'=1}^{\lfloor \epsilon^2(m+1)^2 \rfloor} E^{s'} \left(1 - \left(\frac{s\pi(s)}{(m+1)^2}\right)^{1/2}\right) \\ &= J_{m \times m} - \sum_{s'=1}^{\lfloor \epsilon^2(m+1)^2 \rfloor} E^{s'} \left(\frac{s\pi(s)}{(m+1)^2}\right)^{1/2}. \end{aligned}$$

Thus, at most $\lfloor \epsilon^2(m+1)^2 \rfloor$-rows of Q_h^π are different from the all-ones row. It follows that $rank(Q_h^\pi) \leq 1 + \lfloor \epsilon^2(m+1)^2 \rfloor$, and each state $s \leq \lfloor \epsilon^2(m+1)^2 \rfloor$ that π_h performs optimally at tightens the above upperbound on the rank by one. Since the bound holds for arbitrary $h \in [H-1]$, it follows that it holds for all $h \in [H-1]$. □

C Experimental Details for Oil Discovery Problem

In this section we discuss the rank of the cost function, the rank of the Q^* function, and details of how we tuned the hyperparameters of our algorithms for the experiments presented in Section 8.

C.1 Rank of $c(s, a)$

Recall that $c(s, a) = 0.01 \times \text{round}\left(\frac{|s-a|}{100}\right)$. Figure 2 displays a heat map and the singular values of the cost function.

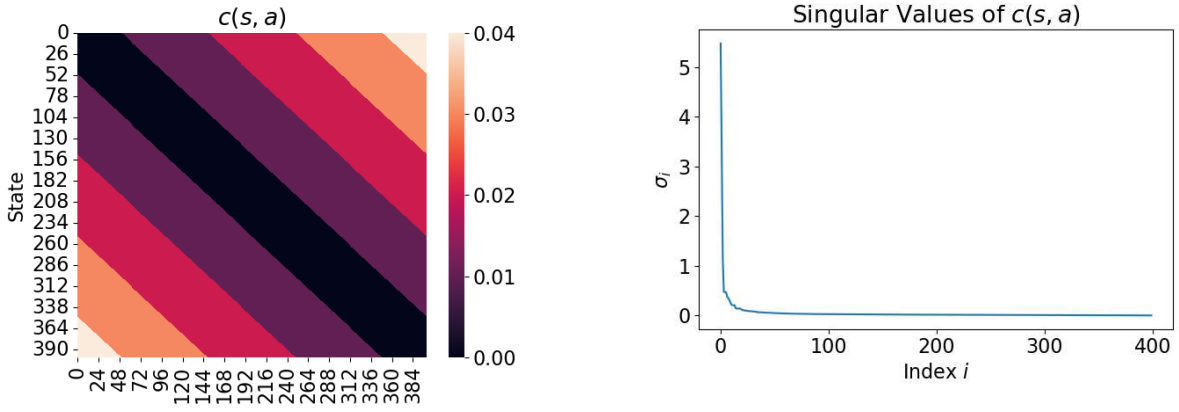


Figure 2: Heat map (left) and singular values (right) of $c(s, a)$.

Even though each entry of $c(s, a)$ can only be one of five values, the rank of $c(s, a)$ is 400 as all of the singular values are greater than zero. However, Figure 2 shows that the magnitude of the singular values decrease quickly. Furthermore, the stable rank of $c(s, a)$ is small $\|c(s, a)\|_F^2 / \|c(s, a)\|_*^2 = 1.46$. It follows that $c(s, a)$ is "approximately" low-rank.

C.2 Rank of Q^*

Figure 3 displays a heat map of Q_1^* and a plot of Q_1^* singular values from largest to smallest.

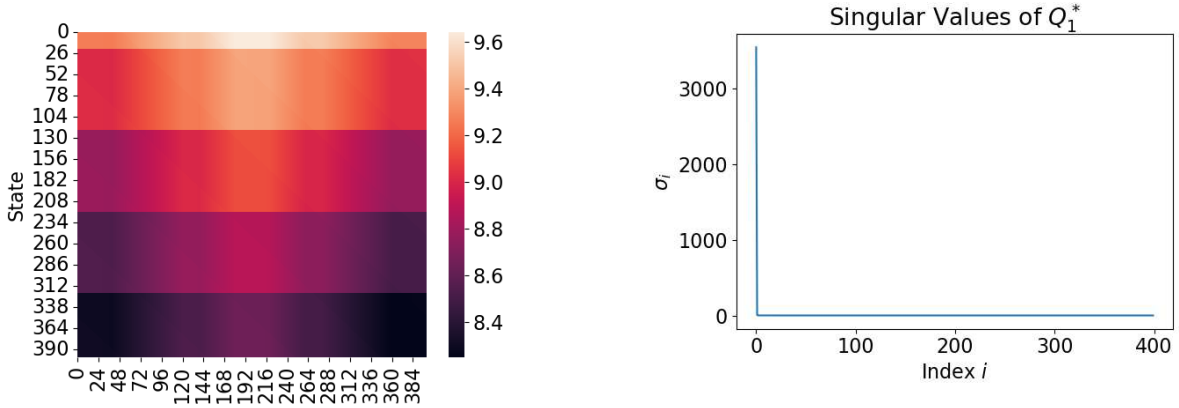


Figure 3: Heat map (left) and singular values (right) of Q_1^*.

While all of the singular values of Q_1^* are greater than zero ($rank(Q_1^*) = 400$), the magnitude of the first singular value is significantly larger than all the other singular values; $\sigma_1 / \sum_{i=1}^{400} \sigma_i = 0.995$. Table 2 displays the rank and stable rank of Q_h^* for $h \in [H]$.

	$h=1$	$h=2$	$h=3$	$h=4$	$h=5$	$h=6$	$h=7$	$h=8$	$h=9$	$h=10$
Rank	400	400	400	400	400	400	400	400	400	400
Stable Rank	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.004	1.000

Table 2: Rank and stable rank of Q_h^* for $h \in [H]$.

From the results in Table 2, it's clear that despite $rank(Q_h^*) = 400$ for all $h \in [H]$, Q_h^* is approximately low rank for all $h \in [H]$ because all the stable ranks are close to one.

C.3 Hyperparameter Tuning

In this section, we discuss the hyperparameters of our algorithms and our methodology on how to choose their value.

Allocation Scheme $N_{s,a,h}$: In the proof of our theorems for LR-EVI and LR-MCPI, the sample allocations $N_{s,a,h}$ are chosen to ensure that at each time step h, the algorithm takes enough samples so that that $\|\bar{Q}_h - Q_h^*\|_\infty \leq \epsilon(H-h+1)/H$. In practice, the algorithm does not have access to the optimal Q function and cannot use this condition as a criteria to choose $N_{s,a,h}$, so we instead empirically test a few different allocation schemes and choose the best. We choose $N_{s,a,h}$ to be uniform for all s, a, not distinguishing between (s, a) in the anchor submatrix or not.

To determine how to allocate samples across the ten time steps for each algorithm, we run a set of experiments benchmarking the algorithm's performance on a set of different allocation schedules. For an allocation scheme $\tau = \{\tau^i\}_{h\in H}$, where $\tau^i \geq 0, \sum_{h\in[H]} \tau^i = 1$, and total sample budget of $\overline{N}$, we will allocate roughly $\tau^i \overline{N}$ samples to the estimation of $\bar{Q}_h$. Essentially τ specifies the proportion of samples that are allocated to the estimates at each step, where there is some rounding involved as the number of samples must be integral. For some sequence of nonnegative numbers $\{a_1^i, a_2^i, \dots a_H^i\}$, the corresponding allocation scheme τ^i follows by simply normalizing according to $\tau_h^i = a_h^i / \sum_{h'\in[H]} a_{h'}^i$. The five different allocation schemes we consider are, $\tau^i = \{\tau_h^i\}_{h\in[H]}$, corresponding to

$$\begin{aligned} a_h^1 &= H-h+1, & a_h^4 &= \lfloor (h+1)/2 \rfloor, \\ a_h^2 &= (H-h+1)^2, & a_h^5 &= 1. \\ a_h^3 &= (H-h+1)^3 \end{aligned}$$

τ^5 is simply a constant allocation schedule, which evenly allocates samples across the steps. When there is an insufficient sample budget to implement other allocation schemes, we let the Empirical Value Iteration algorithms default to τ^5, evenly allocating one-step samples across steps h.

τ^1 is a linearly decreasing allocation schedule. Note that as Monte Carlo Policy Iteration requires samples of length $(H-h+1)$ trajectories to estimate Q_h, the allocation that would uniformly allocate trajectories across h for MCPI corresponds to τ^1. When there is an insufficient sample budget to implement other allocation schemes, we let the Monte Carlo Policy Iteration algorithms default to τ^1, evenly allocating trajectories across steps h.

τ^4 is also a linearly decreasing allocation schedule, but simply at a slower rate. τ^2 is a quadratically decreasing allocation schedule, which matches the allocation schedule chosen in our Theorem 9 for LR-EVI, as indicated by the number of one-step samples N_h scaling as $(H-h+1)^2$ in its dependence on h. τ^3 is a cubically decreasing allocation schedule, which matches the allocation schedule chosen in our Theorems 7 and 8 for LR-MCPI. In particular, the number of trajectories N_h scales as $(H-h+1)^2$ in its dependence on h, but the samples used need to be multiplied by the trajectory length $(H-h+1)$, resulting in a cubic relationship.

Table 3 displays the average entrywise error of $\bar{Q}_1$ of all the algorithms over ten trials for each of these allocation schemes, where we set the sample budget $\overline{N} = 10^8$, $p_S, p_A = 0.025$, and $p_{SI} = 0.2$. "—" refers to the algorithm not having enough samples to take even one sample for each state-action pair at one time step according to the specified allocation scheme.

	τ^1	τ^2	τ^3	τ^4	τ^5
LR-EVI	0.0803	0.0736	0.0948	0.0828	0.0825
LR-MCPI	0.1742	0.1419	0.1532	0.1795	0.2031
LR-EVI + `SI`	0.5638	0.4512	1.0065	0.5541	0.5358
LR-MCPI + `SI`	0.6126	0.6535	0.6868	0.6285	0.6406
EVI	0.2927	1.5558	—	0.255	0.2264
MCPI	0.52	0.5437	—	0.5498	0.6127

Table 3: Mean ℓ_∞ error of $\bar{Q}_1$ of LR-EVI, LR-MCPI, LR-EVI + `SI`, LR-MCPI + `SI`, EVI, and MCPI.

To calibrate these results, recall that entries in Q_1^* take values from roughly 8.3 to 9.6. While the performances are generally pretty similar, from the results in Table 3 the best allocation schemes for the algorithms are τ^2 for LR-EVI, τ^2 for LR-MCPI, τ^2 for LR-EVI + `SI`, τ^1 for LR-MCPI + `SI`, τ^5 for EVI, and τ^1 for MCPI. We use these allocation schemes for the experiments in Section 8.

Note that for our algorithms to run, they require minimally one sample for the value iteration algorithms or one trajectory for the policy iteration algorithms for each $(s, a) \in \Omega_h$. Hence, for smaller values of $\overline{N}$, e.g., $\overline{N} = 10^6$, there may be state-action pairs in Ω_h that do not receive even one sample/trajectory following the best allocation scheme chosen from the data in Table 3. Hence, if that problem occurs, we default to allocation scheme τ^5 for the value iteration algorithms and we default to τ^1 for the policy iteration algorithm. These allocations uniformly spread the samples/trajectories to ensure that the algorithm still produces an estimate when $\overline{N}$ may be small.

Finally we describe the details of the rounding that we implement to ensure that $N_{s,a,h}$ are integral, yet are distributed as close as possible to the desired allocation schedule $\tau = \{\tau^h\}$ with the sample budget of $\overline{N}$. For Empirical Value Iteration algorithms, $N_{s,a,h}$ denotes one-step samples at (s, a) used to construct the estimate for Q_h. Thus we compute initial values by $N_{s,a,h} = \lfloor \tau^h \overline{N} / |\Omega_h| \rfloor$, where the floor function is applied as the number of samples must be integral. Subsequently, we compute the number of excess samples, given by $N_\Delta = \overline{N} - \sum_{h \in [H]} \sum_{(s,a) \in \Omega_h} N_{s,a,h}$. Then, recursing forwards through the horizon, we add one sample to each state action pair in Ω_h, i.e., $N_{s,a,h} = N_{s,a,h} + 1$, provided that there is sufficient samples $N_\Delta \geq |\Omega_h|$. Then we recompute the number of extra samples, i.e., $N_\Delta = N_\Delta - |\Omega_h|$ and repeat continuing at $h+1$. With this rounding scheme, the final number of samples used by our algorithm will be within $[\overline{N} - D^2, \overline{N}]$, where $D^2 = 1.6 \times 10^5$.

For Monte Carlo Policy Iteration algorithms, $N_{s,a,h}$ denotes number of trajectories sampled starting from (s, a), that are used to construct the estimate for Q_h. Thus we compute initial values by $N_{s,a,h} = \lfloor \tau^h \overline{N} / |\Omega_h| (H - h + 1) \rfloor$, where the floor function is applied as the number of trajectories must be integral. Subsequently, we compute the number of excess samples, given by $N_\Delta = \overline{N} - \sum_{h \in [H]} \sum_{(s,a) \in \Omega_h} N_{s,a,h} (H - h + 1)$. Then, recursing forwards through the horizon, we add one trajectory to each state action pair in Ω_h, i.e., $N_{s,a,h} = N_{s,a,h} + 1$, provided that there is sufficient samples $N_\Delta \geq |\Omega_h| (H - h + 1)$. Then we recompute the number of extra samples, i.e., $N_\Delta = N_\Delta - |\Omega_h| (H - h + 1)$ and repeat continuing at $h + 1$. With this rounding scheme, the final number of samples used by our algorithm will be again within $[\overline{N} - D^2, \overline{N}]$.

Choosing $p = p_S = p_A$ for LR-EVI and LR-MCPI: While our theorems use knowledge of the rank and incoherence to choose p_S, p_A, and N_h, one cannot assume knowledge of these quantities in practice. However, many other matrix estimation methods face similar challenges. For example, in [56], to compute the Q function of the optimal policy, they need to choose p_{SI}, which depends on the rank of Q^*, for their algorithm, which combines value iteration and `Soft-Impute`. They show their algorithm is robust to the choice of p_{SI} by showing their algorithm performed similarly for multiple p_{SI} values. Similarly, we show LR-EVI and LR-MCPI are robust to the choice of $p = p_S = p_A$ (the only parameters in LR-EVI and LR-MPCI that depend on the rank and incoherence for a fixed allocation scheme and $\overline{N}$) in a similar manner. To show that LR-EVI and LR-MCPI are robust to $p = p_S = p_A$, we ran both LR-EVI and LR-MPCI with allocation scheme τ^2 and $\overline{N} = 10^8$ for each $p \in [0.025, 0.05, 0.075, 0.1]$, repeating each experiment 10 times. Since Q_h^* effectively has a rank of one, p should be minimally greater than or equal to $1/400 = 0.0025$; ideally even larger to ensure that with high probability there are a sufficient number of rows and columns sample. We set the smallest value of p to be 0.025, which results in an expected number of sampled rows/columns of 10 our of 400, which is already a fairly small number. Table 4 shows the average ℓ_∞ error of $\bar{Q}_1$ at time step $h = 1$ for different values of p.

	$p = 0.025$	$p = 0.05$	$p = 0.075$	$p = 0.1$
LR-EVI	0.077	0.084	0.09	0.129
LR-MCPI	0.152	0.179	0.196	0.216

Table 4: The mean $\|\bar{Q}_1\|_\infty$ error of LR-MCPI and LR-EVI for different values of p.

To calibrate these results, recall that entries in Q_1^* take values from roughly 8.3 to 9.6. The results show that for the different values of p, LR-EVI performs well and the errors are on the same order. Furthermore, the errors are less for smaller values of p. The same results hold for LR-MCPI for the different values of p. As a result, for the experiments in Section 8, we set $p = p_S = p_A = 0.025$. As our table suggests, the algorithm has decent performance for different values of p, so empirically one could choose p based on given computational and memory constraints. The tradeoff is that small values of p could reduce computation and memory usage, but it does assume the MDP satisfies the desired low rank conditions. By choosing p to be as larger, one may increase some robustness to the low rank conditions, as the guarantees would be able to tolerate MDPs with larger ranks.

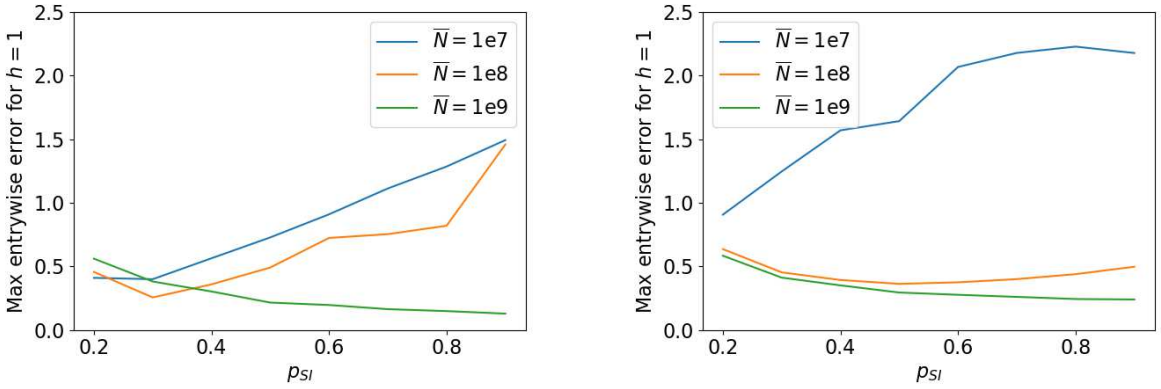
Choosing p_{SI}: In contrast to LR-EVI and LR-MPCI, for larger values of $\overline{N}$, p_{SI} needs to be increased as the gain from decreasing the noise is not as beneficial as observing more samples. For LR-EVI + `SI` and LR-MCPI + `SI`, we determine the best value of p_{SI} for the four different values of $\overline{N} \in [10^6, 10^7, 10^8, 10^9]$ used in our experiments in Section 8. We test eight values of $p_{SI} \in [0.2, 0.3, \ldots, 0.9]$ for the different $\overline{N}$.

For LR-EVI + `SI` to run, it minimally requires one sample for each $(s, a) \in \Omega_h$, which would mean at least $p_{SI} * 1.6 * 10^6$ total samples in expectation. Thus for the lowest sample budget of $\overline{N} = 10^6$, we set $p_{SI} = 0.2$ to ensure that LR-EVI + `SI` has sufficient samples to run successfully for all ten trials.

Similarly, LR-MCPI + `SI` requires at least one trajectory for each $(s, a) \in \Omega_h$ to run, which would mean a total of $p_{SI} * 8.8 * 10^6$ one-step samples in expectation. Thus for the lowest sample budget of $\overline{N} = 10^6$, we set $p_{SI} = 0.075$ to ensure that LR-MCPI + `SI` has sufficient samples to run successfully for all ten trials.

For the larger values of $\overline{N} \in [10^7, 10^8, 10^9]$, we test eight values of $p_{SI} \in [0.2, 0.3, \ldots, 0.9]$. Figure 4 shows the average ℓ_∞ error of $\bar{Q}_1$ for LR-EVI + `SI` and LR-MCPI + `SI` for the different values

of p_{SI} and $\overline{N}$ for LR-EVI + `SI` using allocation scheme τ^2 and LR-MCPI + `SI` using allocation scheme τ^1, where each experiment is repeated ten times.



(a) LR-EVI + `SI`

(b) LR-MCPI + `SI`

Figure 4: Max entrywise error of $\bar{Q}_1$ vs. p_{SI} for four different values of $\overline{N}$ for LR-EVI + `SI` and LR-MCPI + `SI`.

Figure 4 shows that when the sample budget is smaller, i.e. $\overline{N} = 10^7$, smaller values of p perform better; this is expected as there is insufficient sample budget so that increasing p means the number of samples or trajectories allocated to each $(s, a) \in \Omega_h$ will be small, resulting in large noise. For large sample budget, i.e. $\overline{N} = 10^9$, the performance is not very sensitive to the choice of p_{SI}, though the larger values of p_{SI} do perform better. This is also expected as there is sufficient samples to both sample more entries while still having $N_{s,a,h}$ large enough that the noise is well controlled. For $\overline{N} = 10^8$ the performance with respect to p_{SI} is quite different in these two plots, and it may be due in part to the different allocation schemes. Allocation scheme τ^2 significantly skews the proportion of samples to the earlier time steps compared to τ^1. Hence for $\overline{N} = 10^8$, with scheme τ^2, increasing p_{SI} results in the error LR-EVI + `SI` growing perhaps due to high noise in the later time steps. In contrast, with scheme τ^1, for $\overline{N} = 10^8$, the error of LR-MCPI + `SI` does not increase in p_{SI}.

Table 5 displays the value of p_{SI} we use for our experiments in Section 8. As discussed before, the value of p_{SI} is chosen for $\overline{N} = 10^6$ simply to ensure that the observation set is small enough such that the algorithms can produce some estimate for the given sample budget. For $\overline{N} \in [10^7, 10^8, 10^9]$, p_{SI} is chosen according to the value that minimized the error in the results displayed in Figure 4.

	$\overline{N} = 10^6$	$\overline{N} = 10^7$	$\overline{N} = 10^8$	$\overline{N} = 10^9$
LR-EVI + `SI`	0.2	0.3	0.3	0.9
LR-MCPI + `SI`	0.075	0.2	0.5	0.9

Table 5: Values of p_{SI} for each $\overline{N}$ in the experiments in Section 8.

D Additional Experiments for Double Integrator Problem

We empirically evaluate the benefit of including a low-rank subroutine in tabular RL algorithms on the discretized finite-horizon version of the Double Integrator problem, a stochastic control problem seen in [56, 37].

Experimental Setup: We formulate the Double Integrator problem as finite-horizon tabular MDP with state space $S = \{(x, \dot{x})\}$ for $x \in \{-2, -1.9, \ldots, 1.9\}, \dot{x} \in \{-1, -0.9, \ldots, 0.9\}$, action space $A = \{-0.5, -0.499, \ldots, 0.5\}$, and $H = 5$. With this setup, the size of the state space is $|S| = 40 \times 20 = 800$, and the size of the action space is $|A| = 1000$. The learner's goal is to control a unit brick on a frictionless surface and guide it to the origin, state $(0, 0)$. x refers to the brick's position, and $\dot{x}$ denotes the brick's velocity. At each step, the learner is given a noisy reward that penalizes them for the brick's current position,

$$r_h((x, \dot{x}), a) = -\frac{x^2 + \dot{x}^2}{2} + \mathcal{N}(0, 1),$$

for all $h \in [H]$, and $\mathcal{N}(0, 1)$ is a standard normal random variable. The learner chooses an action a to change the velocity of the brick. The dynamics of the system for a given state-action pair $((x, \dot{x}), a)$ for all $h \in [H]$ are

$$x' \coloneqq \min(\max(x + \dot{x}, -2), 1.9), \quad \dot{x}' \coloneqq \min(\max(\lfloor \dot{x} + a \rfloor, -1), 0.9),$$

where $\lfloor \dot{x} \rfloor$ rounds $\dot{x}$ down to the nearest tenth. Since the reward function does not depend on the action, the rank of the reward function is one. Due to the deterministic dynamics, for a given next state $(x', \dot{x}')$, the current state $(x, \dot{x})$ must minimally satisfy $x' = x + \dot{x}$. Thus, there are at most twenty $(x, \dot{x})$ pairs that satisfy $x' = x + \dot{x}$ (the velocity can only take on twenty different values), so there are at most twenty non-zero entries in $P((x', \dot{x}')|\cdot, \cdot)$. Therefore, the Tucker rank of $P((x', \dot{x}')|(x, \dot{x}), a)$ is upperbounded by $(|S|, 20, |A|)$. Hence, this MDP satisfies Assumption 4. Figure 5 displays a heat map of Q_1^* and a plot of Q_1^* singular values from largest to smallest.

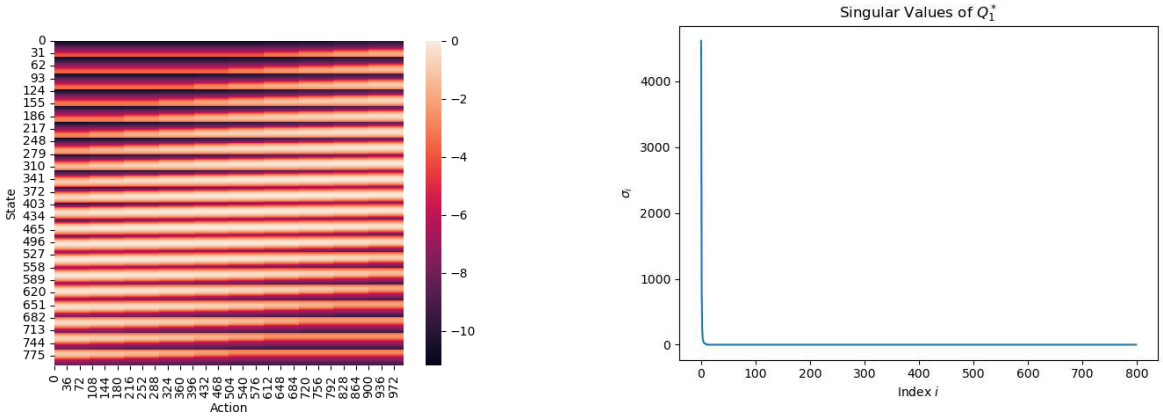


Figure 5: Heat map (left) and singular values (right) of Q_1^*.

While the transition kernel has Tucker rank upper bounded by $(|S|, 20, |A|)$, the rank of Q_h^* is ten for $h \in [4]$ while the rank of $Q_H^* = r_H$ is one. Table 6 displays the rank and stable rank of Q_h^* for $h \in [H]$.

	$h = 1$	$h = 2$	$h = 3$	$h = 4$	$h = 5$
Rank	10	10	10	10	1
Stable Rank	1.04	1.02	1.01	1.00	1.00

Table 6: Rank and stable rank of Q_h^* for $h \in [H]$.

From the results in Table 6, it's clear that the rank of Q_h^* is much smaller than $|S|$ or $|A|$.

Algorithms: We compare the same algorithms used in the oil discovery experiments. Since $|S| \neq |A|$, we allow for p_S and p_A to be different. Instead of using `Soft-Impute` from the `fancyimpute` package, we use the implementation from [17] because it yielded better results and shorter runtimes.

D.1 Hyperparameter Tuning

In this section, we discuss the results of tuning the allocation schemes, p_S, p_A, and p_{SI} for our different algorithms.

Allocation Schemes: To determine how to divide samples across the five time steps, we test our algorithms on the five different allocation schemes introduced in Appendix C.3. Recall that the allocation scheme τ^i is $\tau_h^i = a_h^i / \sum_{h' \in [H]} a_{h'}^i$ for $[a_1, \ldots, a_H]$. The five different allocation schemes we consider are, $\tau^i = \{\tau_h^i\}_{h \in [H]}$, corresponding to

$$\begin{aligned} a_h^1 &= H - h + 1, & a_h^4 &= \lfloor (h+1)/2 \rfloor, \\ a_h^2 &= (H - h + 1)^2, & a_h^5 &= 1. \\ a_h^3 &= (H - h + 1)^3 \end{aligned}$$

With our implementation, roughly $\tau_h^i \bar{N}$ samples are allocated to estimating Q_h^* ($\bar{N}$ is the sample budget). Table 7 displays the mean entrywise error of $\bar{Q}_1$ of all the algorithms over five trials for each allocation schemes. We set the sample budget $\overline{N} = 10^8$, $p_S = 0.1, p_A = 0.08$, and $p_{SI} = 0.4$.

	τ^1	τ^2	τ^3	τ^4	τ^5
LR-EVI	0.761	1.20	4.30	2.15	0.507
LR-MCPI	0.550	0.416	0.633	2.07	1.16
LR-EVI+SI	0.459	0.469	1.343	0.512	0.456
LR-MCPI+SI	0.394	0.405	0.388	0.436	0.441
EVI	1.044	2.921	1.343	3.074	1.469
MCPI	0.642	0.615	1.09	2.225	2.526

Table 7: Mean ℓ_∞ error of $\bar{Q}_1$ of LR-EVI, LR-MCPI, LR-EVI+SI, LR-MCPI+SI, EVI, and MCPI.

While the errors are roughly similar for many of the allocation schemes for each algorithm, we choose the allocation scheme that corresponds to the lowest error. Hence, we use allocation scheme τ^5 for LR-EVI, τ^2 for LR-MCPI, τ^5 for LR-EVI+SI, τ^3 for LR-MCPI+SI, τ^1 for EVI, and τ^2 for MCPI.

Choosing p_S and p_A: Similar to `Soft-Impute`, varying p_s and p_a as a function of the total number of samples improves the performance of LR-EVI and LR-MPCI. When the sample budget is small ($\bar{N} = 10^7$), one should set p_s and p_a to be smaller, which increases the bias from the matrix estimation method but decreases the noise on the empirical estimates. However, when the sample budget is increased ($\bar{N} = 10^9$), one should increase p_s and p_a to reduce the bias of the matrix estimation method as the estimation error on $\hat{Q}$ is already very small. Thus, for $\bar{N} \in [10^7, 10^8, 10^9]$, we try the following $(p_s, p_a) \in [(0.025, 0.02), (0.05, 0.04), (0.1, 0.08), (0.2, 0.16)]$ over five trials. Table 8 displays the entrywise error of $\bar{Q}_1$ obtained from running LR-EVI with allocation scheme τ^5.

Table 9 displays the entrywise error of $\bar{Q}_1$ obtained from running LR-MCPI with allocation scheme τ^2.

	$\bar{N} = 10^7$	$\bar{N} = 10^8$	$\bar{N} = 10^9$
$(p_s, p_a) = (0.025, 0.02)$	1.72	1.09	0.650
$(p_s, p_a) = (0.05, 0.04)$	2.74	0.236	0.190
$(p_s, p_a) = (0.1, 0.08)$	5.43	0.488	0.0175
$(p_s, p_a) = (0.2, 0.16)$	10.9	1.05	0.0365

Table 8: Mean ℓ_∞ error of $\bar{Q}_1$ of LR-EVI for varying values of (p_S, p_A).

	$\bar{N} = 10^7$	$\bar{N} = 10^8$	$\bar{N} = 10^9$
$(p_s, p_a) = (0.025, 0.02)$	2.26	1.50	1.34
$(p_s, p_a) = (0.05, 0.04)$	4.70	0.293	0.143
$(p_s, p_a) = (0.1, 0.08)$	4.54	0.288	0.0452
$(p_s, p_a) = (0.2, 0.16)$	10.84	0.747	0.108

Table 9: Mean ℓ_∞ error of $\bar{Q}_1$ of LR-MCPI for varying values of (p_S, p_A).

Hence, for LR-EVI, we use $(p_s, p_a) = (0.025, 0.02)$ for $\bar{N} = 10^7$, $(p_s, p_a) = (0.05, 0.04)$ for $\bar{N} = 10^8$, and $(p_s, p_a) = (0.1, 0.08)$ for $\bar{N} = 10^9$. For LR-MCPI, we use $(p_s, p_a) = (0.025, 0.02)$ for $\bar{N} = 10^7$, $(p_s, p_a) = (0.1, 0.08)$ for $\bar{N} = 10^8$, and $(p_s, p_a) = (0.1, 0.08)$ for $\bar{N} = 10^9$.

Choosing p_{SI}: For LR-EVI + `SI` and LR-MCPI + `SI`, we test different values of p_{SI} for $\overline{N} \in [10^7, 10^8, 10^9]$ to determine what to set p_{SI} to in our final experiments in Section 8. We test five values of $p_{SI} \in [0.1, 0.5, \ldots, 0.9]$ for the different $\overline{N}$. Figure 6 shows the average ℓ_∞ error of $\bar{Q}_1$ for LR-EVI + `SI` and LR-MCPI + `SI` for the different values of p_{SI} and $\overline{N}$ for LR-EVI + `SI` using allocation scheme τ^5 and LR-MCPI + `SI` using allocation scheme τ^3, where each experiment is repeated five times.

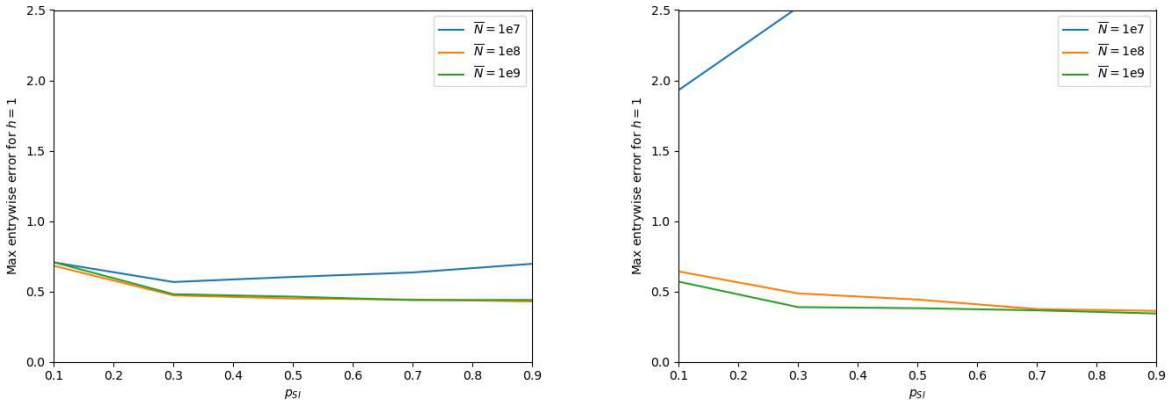


Figure 6: Mean ℓ_∞ error of $\bar{Q}_1$ of LR-EVI+SI(Left) and LR-MCPI+SI(Right) for various values of p_{SI} and $\bar{N}$

From these results, we choose the p_{SI} value that corresponds to the lowest error for our final experiments. Note that for $\bar{N} = 10^7$, the error is strictly increasing as p_{SI} increases for LR-MCPI+SI.

Hence, for LR-EVI+SI, we use $p_{SI} = 0.3$ for $\bar{N} = 10^7$, $p_{SI} = 0.9$ for $\bar{N} = 10^8$, and $p_{SI} = 0.9$ for $\bar{N} = 10^9$. For LR-MPCI+SI, we use $p_{SI} = 0.1$ for $\bar{N} = 10^7$, $p_{SI} = 0.9$ for $\bar{N} = 10^8$, and $p_{SI} = 0.9$ for $\bar{N} = 10^9$.

Results: For each value of the sample budget $\bar{N} \in [10^7, 10^8, 10^9]$, we run each of the six algorithms ten times, with the hyperparameters specified above, and compute the average ℓ_∞ error of $\bar{Q}_1$. Figure 7 displays the mean entrywise error of $\bar{Q}_1$ over ten simulations with the error bars corresponding to the standard deviation. For vanilla MCPI to produce an estimate of Q_1^*, it requires at least $\sum_{h=1}^{H} SAh = 1.2 \times 10^7$ one-step samples. Hence, there is no error bar for MCPI with $\bar{N} = 10^7$.

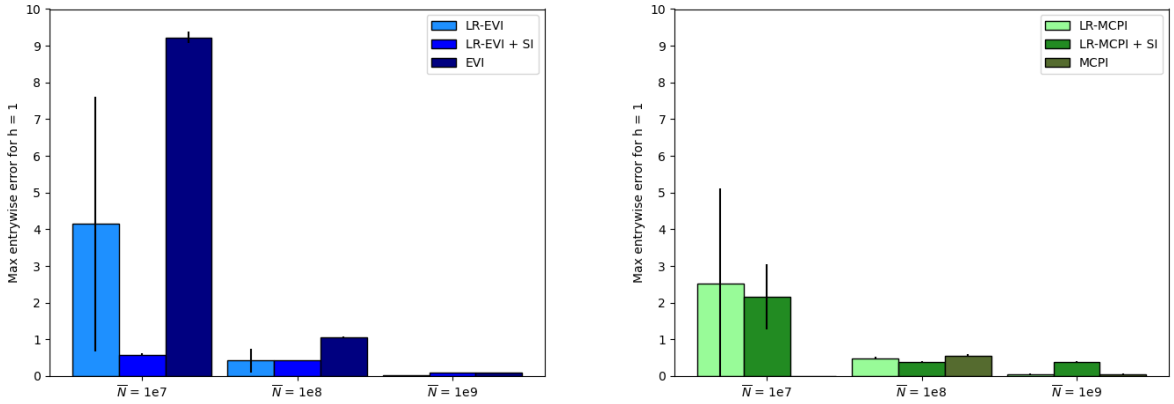


Figure 7: Max entrywise error of $\bar{Q}_1$ vs. sample budget for LR-EVI, LR-EVI + `Soft Impute`, empirical value iteration and LR-MCPI, LR-MCPI + `Soft-Impute`, Monte Carlo policy iteration at $h = 1$. Note that the optimal Q_1^* function ranges in value from roughly -11.2 to 0, such that 1.12 error would be roughly 10% error.

Similarly to the results from the oil discovery problem, Figure 7 shows that even when there are not enough samples to MCPI, LR-MCPI produces a reasonable estimate. Furthermore, the low-rank algorithms perform better than the tabular versions, EVI and MCPI, when the sample budget is small, i.e., $\bar{N} = 10^7$ or $\bar{N} = 10^8$. When the sample budget is large, i.e., $\bar{N} = 10^9$, the low-rank methods perform similarly to EVI and MCPI, except for LR-MCPI+SI. The relatively large error from LR-MCPI+SI for $\bar{N} = 10^9$ suggests that the matrix estimation methods are sensitive to the choice of hyperparameters, so in practice, one should carefully tune these given their computational limits, e.g., storage and runtime constraints. In contrast to the oil discovery simulations, the policy iteration algorithms achieve a similar error to the value iteration algorithms with the same sample budget.

E Proof of Lemma 1

Proof of Lemma 1. Consider the MDP defined in Section 4. Let $\pi_h(1) = \pi_h(2) = 2$ for all $h \in \{2, \dots, H-1\}$. We prove that

$$Q_h^{\pi,\theta} = \begin{pmatrix} \frac{1}{4} & \frac{1}{2} \\ \frac{1}{2} + 2^{H-h}\theta, & 1 + 2^{H-h+1}\theta \end{pmatrix}, \qquad V_h^{\pi,\theta} = \begin{pmatrix} \frac{1}{2} \\ 1 + 2^{H-h+1}\theta \end{pmatrix}$$

with backwards induction on h. Since $V_H^{\pi,\theta} = \begin{pmatrix} \frac{1}{2} \\ 1+2\theta \end{pmatrix}$, the base case occurs at step $H-1$. Applying the exact Bellman operator, it follows that

$$Q_{H-1}^{\pi,\theta} = \begin{pmatrix} \frac{1}{4} & \frac{1}{2} \\ \frac{1}{2}+2\theta, & 1+2^2\theta \end{pmatrix}$$

and

$$V_{H-1}^{\pi,\theta} = \begin{pmatrix} \frac{1}{2} \\ 1+2^2\theta \end{pmatrix}$$

because for both values of θ, $1+4\theta > 0$. Next, assume that the induction hypothesis holds, that is for some $t \in \{2, \ldots, H-1\}$,

$$Q_t^{\pi,\theta} = \begin{pmatrix} \frac{1}{4} & \frac{1}{2} \\ \frac{1}{2}+2^{H-t}\theta, & 1+2^{H-t+1}\theta \end{pmatrix}, \qquad V_t^{\pi,\theta} = \begin{pmatrix} \frac{1}{2} \\ 1+2^{H-t+1}\theta \end{pmatrix}.$$

Applying the exact Bellman operator, it follows that

$$\begin{aligned} Q_{t-1}^{\pi,\theta} &= r_{t-1} + P_{t-1}V_t^{\pi,\theta} \\ &= \begin{pmatrix} -\frac{1}{4} & 0 \\ -\frac{1}{2} & 2^{H-t+1}\theta \end{pmatrix} + \begin{pmatrix} V_t^{\pi,\theta}(1) & V_t^{\pi,\theta}(1) \\ V_t^{\pi,\theta}(2) & (2) \end{pmatrix} \\ &= \begin{pmatrix} \frac{1}{4} & \frac{1}{2} \\ \frac{1}{2}+2^{H-t+1}\theta, & 1+2^{H-t+2}\theta \end{pmatrix}. \end{aligned}$$

Because $2^H|\theta| = 3/4$, $1+2^H\theta > 0$, which implies that $Q_{t-1}^{\pi,\theta}(2,2) > Q_{t-1}^{\pi,\theta}(2,1)$. Therefore,

$$V_{t-1}^{\pi,\theta} = \begin{pmatrix} \frac{1}{2} \\ 1+2^{H-t+2}\theta \end{pmatrix}$$

and the induction hypothesis holds. Finally, since one stays in the same state at all steps after $h=1$ by construction, $\pi_h(1) = \pi_h(2) = 2$ for all $h \in \{2, \ldots, H-1\}$ is the unique optimal policy because $2^H|\theta| = 3/4$, which implies that $Q_h^{\pi,\theta}(2,2) = 2Q_h^{\pi,\theta}(2,1) > 0$. □

F Proof of Proposition 4

Proposition 4 states that if the reward function and transition kernel are low rank, then for any value function estimate $\hat{V}_{h+1}$, $r_h + [P_h\hat{V}_{h+1}]$ has rank upper bounded by d.

Proof of Proposition 4. Let MDP $M = (S, A, P, r, H)$ satisfy Assumption 4 (specifically, P_h has Tucker rank $(|S|, |S|, d)$. Hence, for any value function estimate $\hat{V}_{h+1}$,

$$\begin{aligned} r_h(s,a) + P_h\hat{V}_{h+1} &= \sum_{i=1}^{d} W^{(h)}(s,i)V^{(h)}(a,i) + \sum_{s'\in S} \hat{V}_{h+1}(s')P_h(s'|s,a) \\ &= \sum_{i=1}^{d} W^{(h)}(s,i)V^{(h)}(a,i) + \sum_{s'\in S} \hat{V}_{h+1}(s') \sum_{i=1}^{d} U^{(h)}(s',s,i)V^{(h)}(a,i) \\ &= \sum_{i=1}^{d} V^{(h)}(a,i) \left(W^{(h)}(s,i) + \sum_{s'\in S} \hat{V}_{h+1}(s')U^{(h)}(s',s,i) \right). \end{aligned}$$

Since $W^{(h)}(:,:) + \sum_{s'\in S} \hat{V}_{h+1}(s')U^{(h)}(s',:,:)$ is an $|S| \times d$ matrix, $r_h(s,a) + P_h\hat{V}_{h+1}$ has rank at most d. The same result holds when P_h has Tucker rank $(|S|, d, |A|)$ from a similar argument. □

G Proof of Lemma 10 (Random Sampling of Anchor States and Actions)

As stated in Lemma 10, our sampling method is as follows: we sample states and actions using the Bernoulli model. Let $\tilde{U} \in \mathbb{R}^{|S| \times d}, \tilde{V} \in \mathbb{R}^{|A| \times d}$ such that

$$\tilde{U}_i = \begin{cases} U_i \text{ with probability } p_1, \\ 0 \text{ otherwise} \end{cases}, \qquad \tilde{V}_i = \begin{cases} V_i \text{ with probability } p_2, \\ 0 \text{ otherwise} \end{cases}$$

Let $\tilde{Q}_h := \tilde{U}\Sigma\tilde{V}^\top \in \mathbb{R}^{|S| \times |A|}$. The sampled anchor states and actions are the states corresponding to the non-zero rows and columns, respectively. We remark that the Bernoulli model is chosen for convenience and similar results hold if we sample with replacement. To prove Lemma 10, we present two intermediate lemmas, the first shows $p_1^{-1/2}\tilde{U}$ and $p_2^{-1/2}\tilde{V}$ have near orthonormal columns, which implies that $\tilde{U}$ and $\tilde{V}$ have full column rank, with high probability.

Lemma 23. *Let $Q_h, U, \tilde{U}, \Sigma, V$, and $\tilde{V}$ be defined as above. Let Q_h be μ-incoherent. Then, with probability at least $1 - 4(|S| \wedge |A|)^{-10}$, we have*

$$\|p_1^{-1}\tilde{U}^\top\tilde{U} - I_{d\times d}\|_{op} \le \sqrt{\frac{40\mu d \log(|S|)}{p_1|S|}} + \frac{40\mu d \log(|S|)}{p_1|S|}$$

$$\|p_2^{-1}\tilde{V}^\top\tilde{V} - I_{d\times d}\|_{op}, \le \sqrt{\frac{40\mu d \log(|A|)}{p_2|A|}} + \frac{40\mu d \log(|A|)}{p_2|A|}.$$

Proof of Lemma 23. For each $i \in [|S|]$, let $Z^{(i)} \in \mathbb{R}^{|S|\times d}$ be the matrix obtained from U by zeroing out all but the i-th row. Let $\delta_1, \ldots, \delta_{|S|}$ be i.i.d. Bernoulli(p_1) random variables. We can express

$$U = \sum_{i\in[|S|]} Z^{(i)} \text{ and } \tilde{U} = \sum_{i\in[|S|]} \delta_i Z^{(i)}$$

Note that

$$\tilde{U}^\top\tilde{U} = \sum_{i\in[|S|]}\sum_{j\in[|S|]} \delta_i\delta_j Z^{(i)\top}Z^{(j)} \tag{10}$$

$$= \sum_{i\in[|S|]} \delta_i^2 Z^{(i)\top}Z^{(i)} \tag{11}$$

by construction of $Z^{(i)}$ and $Z^{(j)}$. Hence,

$$\begin{aligned}\mathbb{E}[\tilde{U}^\top\tilde{U}] &= p_1 \sum_{i\in[|S|]} {Z^{(i)}}^\top Z^{(i)} \\ &= p_1 \sum_{i\in[|S|]}\sum_{j\in[|S|]} {Z^{(i)}}^\top Z^{(j)} \\ &= p_1 U^\top U \\ &= p_1 I_{d\times d} \end{aligned} \tag{12}$$

where the last equality is due to U having orthonormal columns. For each $i \in [|S|]$, we define the following the mean-zero matrices

$$X^{(i)} := (\delta_i^2 - \mathbb{E}[\delta_i^2])Z^{(i)\top}Z^{(i)} = (\delta_i - p_1)Z^{(i)\top}Z^{(i)}.$$

Since Q_h^* is $\mu-$incoherent,

$$\|X^{(i)}\|_{op} \leq |\delta_i - p_1| \|Z^{(i)^\top} Z^{(i)}\|_{op} \leq \|Z^{(i)^\top} Z^{(i)}\|_{op} = \|U_{i-}\|_2^2 \leq \frac{\mu r d}{|S|} \quad \text{surely.}$$

Furthermore,

$$\begin{aligned}
\sum_{i\in[|S|]} \mathbb{E}[X^{(i)^\top} X^{(i)}] = \sum_{i\in[|S|]} \mathbb{E}[X^{(i)} X^{(i)^\top}] &= \sum_{i\in[|S|]} \mathbb{E}[(\delta_i - p)^2] Z^{(i)^\top} Z^{(i)} Z^{(i)^\top} Z^{(i)} \\
&= p_1(1-p_1) \sum_{i\in[|S|]} \|U_{i-}\|_2^2 Z^{(i)^\top} Z^{(i)} \\
&\preceq p_1 \cdot \frac{d\mu}{|S|} \sum_{i\in[|S|]} Z^{(i)^\top} Z^{(i)} \\
&= \frac{d\mu p_1}{|S|} U^T U \\
&= \frac{d\mu p_1}{|S|} I_{d\times d}.
\end{aligned}$$

Thus,

$$\| \sum_{i\in[|S|]} \mathbb{E}[X^{(i)^\top} X^{(i)}] \|_{op} = \| \sum_{i\in[|S|]} \mathbb{E}[X^{(i)} X^{(i)^\top}] \|_{op} \leq \frac{d\mu p_1}{|S|}$$

From the matrix Bernstein inequality (Theorem 32), we have

$$\begin{aligned}
\mathbb{P}\left(\|\tilde{U}^\top \tilde{U} - p_1 I_{d\times d}\|_{op} \geq t \right) &= \mathbb{P}\left(\left\| \sum_{i\in[|S|]} \left((\delta_i^2 - p_1) Z^{(i)^\top} Z^{(i)} \right) \right\|_{op} \geq t \right) \\
&= \mathbb{P}\left(\left\| \sum_{i\in[|S|]} X^{(i)} \right\|_{op} \geq t \right) \\
&\leq 2|S| \exp\left(-\frac{t^2/2}{\frac{p_1\mu d}{|S|} + \frac{\mu d}{3|S|} t} \right) \\
&\leq 2|S| \exp\left(-\frac{t^2}{\frac{2p_1\mu d}{|S|} + \frac{2\mu d}{|S|} t} \right)
\end{aligned}$$

where the first equality follows from equations 11 and 12. For $t = \sqrt{\frac{40 p_1 \mu d \log(|S|)}{|S|}} + \frac{40\mu d \log(|S|)}{|S|}$, we have

$$\left\| \tilde{U}^\top \tilde{U} - p_1 I_{d\times d} \right\|_{op} \leq \sqrt{\frac{40 p_1 \mu d \log(|S|)}{|S|}} + \frac{40\mu d \log(|S|)}{|S|}$$

with probability at least $1 - 2|S|^{-10}$. Dividing both sides by p_1 yields the first inequality in the lemma. The corresponding bound for $\tilde{V}$ holds from a similar argument. Taking a union bound over the two events proves the lemma. □

Now, we present our second lemma that shows that the uniformly sample submatrix ($\tilde{O}(d)$ by $\tilde{O}(d)$ in expectation) has rank-d with its smallest non-zero singular value bounded away from zero.

Lemma 24. *Let $p_1 = \frac{\mu d \log(|S|)}{320|S|}$ and $p_2 = \frac{\mu d \log(|A|)}{320|A|}$. Under the event in Lemma 23, we have*

$$\sigma_d((p_1 \vee p_2)^{-1}\tilde{Q}) \geq \frac{1}{2}\sigma_d(Q_h).$$

Proof Of Lemma 24. Under the assumption that $p_1 = \frac{\mu d \log(|S|)}{320|S|}$ and $p_2 = \frac{\mu d \log(|A|)}{320|A|}$ and the event in Lemma 23, we have $\|p_1^{-1}\tilde{U}^\top\tilde{U} - I_{d\times d}\|_{op} \leq \frac{1}{2}$. From Weyl's inequality, we have $\sigma_d(p_1^{-1}\tilde{U}^\top\tilde{U}) \geq \frac{1}{2}$, which implies $\sigma_d(p_1^{-1/2}\tilde{U}) \geq \frac{1}{\sqrt{2}}$. From a similar argument, $\sigma_d(p_1^{-1/2}\tilde{V}) \geq \frac{1}{\sqrt{2}}$. Let $p = p_1 \vee p_2$, from the singular value version of the Courant-Fischer minimax theorem (Theorem 7.3.8 [21]), we have

$$\begin{aligned}
\sigma_d(p^{-1}\tilde{Q}) &= \max_{S:dim(S)=d}\min_{x\in S, x\neq 0} \frac{\|p^{-1}\tilde{U}\Sigma\tilde{V}^\top x\|_2}{\|x\|_2} \\
&= \max_{S:dim(S)=d}\min_{x\in S, x\neq 0} \frac{\|(p^{-1/2}\tilde{U})\Sigma(p^{-1/2}\tilde{V}^\top)x\|_2}{\|\Sigma(p^{-1/2}\tilde{V}^\top)x\|_2}\frac{\|\Sigma(p^{-1/2}\tilde{V}^\top)x\|_2}{\|p^{-1/2}\tilde{V}^\top x\|_2}\frac{\|p^{-1/2}\tilde{V}^\top x\|_2}{\|x\|_2} \\
&\geq \max_{S:dim(S)=d}\min_{x\in S, x\neq 0} \frac{\|(p^{-1/2}\tilde{U})\Sigma(p^{-1/2}\tilde{V}^\top)x\|_2}{\|(p^{-1/2}\tilde{U})^\dagger\|_{op}\|(p^{-1/2}\tilde{U})\Sigma(p^{-1/2}\tilde{V}^\top)x\|_2} \\
&\qquad \cdot \frac{\|\Sigma(p^{-1/2}\tilde{V}^\top)x\|_2}{\|\Sigma^{-1}\|_{op}\|\Sigma(p^{-1/2}\tilde{V}^\top)x\|_2}\frac{\|p^{-1/2}\tilde{V}^\top x\|_2}{\|x\|_2} \\
&= \sigma_d(p^{-1/2}\tilde{U})\cdot\sigma_d(\Sigma)\max_{S:dim(S)=d}\min_{x\in S, x\neq 0}\frac{\|p^{-1/2}\tilde{V}^\top x\|_2}{\|x\|_2} \\
&= \sigma_d(p^{-1/2}\tilde{U})\cdot\sigma_d(\Sigma)\sigma_d(p^{-1/2}\tilde{V}^\top) \\
&\geq \sigma_d(p_1^{-1/2}\tilde{U})\cdot\sigma_d(\Sigma)\sigma_d(p_2^{-1/2}\tilde{V}^\top) \\
&\geq \frac{1}{\sqrt{2}}\sigma_d(Q_h)\frac{1}{\sqrt{2}} \\
&= \frac{1}{2}\sigma_d(Q_h)
\end{aligned}$$

where the first inequality comes from properties of the operator norm and inverses/pseudo-inverses and the second inequality comes from replacing $p = p_1 \vee p_2$ with either p_1 or p_2. □

Using the two above lemmas, we next prove Lemma 10.

Proof of Lemma 10. Let p_1, p_2 be defined as in Lemma 24. From the previous two lemmas, it follows that with probability at least $1 - 4(|S| \wedge |A|)^{-10}$, we have $\sigma_d((p_1 \vee p_2)^{-1}\tilde{Q}) \geq \frac{1}{2}\sigma_d(Q_h)$. Next, we upper bound $\alpha = \frac{\|Q_h\|_\infty}{\sigma_d(Q_h(S^\#, A^\#))}$ assuming that Q_h is μ-incoherent with condition number κ. Let the singular value decomposition of the rank d matrix Q_h be $Q_h = U\Sigma V^\top$. For $(s,a) \in S \times A$,

$$\begin{aligned}
|Q_h(s,a)| &= |U_s\Sigma V_a| \\
&\leq \|\Sigma\|_{op}|U_s V_a| \\
&\leq \sigma_1(Q_h)\|U_s\|_2|V_a\|_2 \\
&\leq \sigma_1(Q_h)\sqrt{\frac{\mu d}{|S|}}\sqrt{\frac{\mu d}{|A|}} \\
&= \frac{d\sigma_1(Q_h)\mu}{\sqrt{|S||A|}}
\end{aligned}$$

where the third inequality comes from Q_h being μ incoherent. Hence,

$$\begin{aligned}
\frac{\|Q_h\|_\infty}{\sigma_d(Q_h(S^\#, A^\#))} &\leq \frac{d\sigma_1(Q_h)\mu}{\sigma_d(Q_h(S^\#, A^\#))\sqrt{|S||A|}} \\
&\leq \frac{d\sigma_1(Q_h)\mu}{\sigma_d(Q_h(S^\#, A^\#))(|S| \wedge |A|)} \\
&= \frac{320\sigma_1(Q_h)}{\sigma_d((p_1 \vee p_2)^{-1}Q_h(S^\#, A^\#))\log(|S| \wedge |A|)} \\
&= \frac{640\sigma_1(Q_h)}{\sigma_d(Q_h)\log(|S| \wedge |A|)} \\
&= \frac{640\kappa}{\log(|S| \wedge |A|)}
\end{aligned}$$

where the third line comes from the definition of p_1 and p_2 and the fourth line comes from Lemma 23. Hence, $\alpha \in O(\kappa)$. Next, we upperbound the size of the anchor sets with high probability.

From the one-sided Bernstein's inequality, Proposition 31, for $C'' = \frac{25600}{3\mu d}$,

$$\begin{aligned}
\mathbb{P}\left(|S^\#| - \mathbb{E}[|S^\#|] \geq C'' p_1 |S|\right) &\leq \exp\left(-\frac{p_1^2 (C'')^2 |S|}{2(p_1 + \frac{p_1 C''}{3})}\right) \\
&\leq \exp\left(-\frac{\mu d C''}{640(1+\frac{1}{3})}\log(|S|)\right) \\
&= |S|^{-10}.
\end{aligned}$$

With a similar argument,

$$\mathbb{P}\left(|A^\#| - \mathbb{E}[|A^\#|] \geq C'' p_2 |A|\right) \leq |A|^{-10}.$$

From our definition of p_1, p_2, it follows that $\mathbb{E}\left[|S^\#|\right] = O\left(d\mu \log(|S|)\right)$ and $\mathbb{E}\left[|A^\#|\right] = O\left(d\mu \log(|A|)\right)$. A union bound on the above two events and the one in Lemma 23 asserts that

$$|S^\#| \in O\left(d\mu \log(|S|)\right), \quad |A^\#| \leq O\left(d\mu \log(|A|)\right), \quad \text{and } \alpha \in O(\kappa)$$

with probability at least $1 - 6(|S| \wedge |A|)^{-10}$.

□

H Proof of Lemma 12 (Entrywise Bounds for Matrix Estimation)

Lemma 12 provides bounds for the entrywise error amplification of the matrix estimation method as a function of on k and α, assuming that $S^\#$ and $A^\#$ are (k, α)-anchor states and actions for matrix Q_h.

Proof of Lemma 12. Let $S^\#$ and $A^\#$ be (k, α)-anchor states and actions for matrix Q_h. For all $(s, a) \in \Omega^\# = S^\# \times A^\#$, assume that $\hat{Q}_h(s, a)$ satisfies $|\hat{Q}_h(s, a) - Q_h(s, a)| \leq \eta^\#$, and for all $(s, a) \in \Omega \setminus \Omega^\#$, assume that $\hat{Q}_h(s, a)$ satisfies $|\hat{Q}_h(s, a) - Q_h(s, a)| \leq \eta$. We follow the same argument as the proof of Proposition 13 in [37] except we upperbound equations (22) and (23) with $\|Q_h\|_\infty$ instead of $V_{\max}$. Following the steps in [37], i.e., using the triangle inequality and from the

definition of the operator norm, for all $(s,a) \in S \times A$, since $S^\#$ and $A^\#$ are (k,α)-anchor states and actions,

$$|\bar{Q}_h(s,a) - Q_h(s,a)| \leq \sqrt{2} \left\| [\hat{Q}_h(S^\#, A^\#)]^\dagger \right\|_{op} \left\| \hat{Q}_h(S^\#, a)\hat{Q}_h(s, A^\#) - Q_h(S^\#, a)Q_h(s, A^\#) \right\|_F + \left\| [\hat{Q}_h(S^\#, A^\#)]^\dagger - [Q_h(S^\#, A^\#)]^\dagger \right\|_{op} \left\| Q_h(S^\#, a)Q_h(s, A^\#) \right\|_F.$$

Following the steps in the proof of Proposition 13, we upperbound the first operator norm term with Weyl's inequality and our assumption on ϵ and the second operator norm term with a classic result from perturbing pseudoinverses,

$$\left\| [\hat{Q}_h(S^\#, A^\#)]^\dagger \right\|_{op} \leq \frac{2}{\sigma_d(Q_h(S^\#, A^\#))}$$

$$\left\| [\hat{Q}_h(S^\#, A^\#)]^\dagger - [Q_h(S^\#, A^\#)]^\dagger \right\|_{op} \leq 2(1+\sqrt{5}) \frac{\eta^\# k}{\sigma_d(Q_h(S^\#, A^\#))^2}.$$

Since for all $s, s' \in S$ and $a, a' A$,

$$\begin{aligned} \left| \hat{Q}_h(s',a)\hat{Q}_h(s,a') - Q_h(s',a)Q_h(s,a') \right| &\leq \left| (Q_h(s',a)+\eta)(Q_h(s,a')+\eta) - Q_h(s',a)Q_h(s,a') \right| \\ &\leq \eta |Q_h(s',a)| + \eta |Q_h(s,a')| + \eta^2 \\ &\leq 2\eta \|Q_h\|_\infty + \eta^2, \end{aligned}$$

then, $\left\| \hat{Q}_h(S^\#, a)\hat{Q}_h(s, A^\#) - Q_h(S^\#, a)Q_h(s, A^\#) \right\|_F \leq (2\eta\|Q_h\|_\infty + \eta^2)k$. Because $|Q_h(s',a)Q_h(s,a')| \leq \|Q_h\|_\infty^2$ for all $s, s' \in S$ and $a, a' A$, clearly $\left\| Q_h(S^\#, a)Q_h(s, A^\#) \right\|_F \leq \|Q_h\|_\infty^2 k$. Using these inequalities gives that for all $(s,a) \in S \times A$,

$$|\bar{Q}_h(s,a) - Q_h(s,a)| \leq \left(6\sqrt{2}\alpha k \eta + 2(1+\sqrt{5})\alpha^2 k^2 \eta^\# \right) \in O(\alpha k \eta + \alpha^2 k^2 \eta^\#) \tag{13}$$

since $\eta \leq \|Q_h\|_\infty$. □

I Inductive Arguments for Theorems 7, 8, and 9

We next present the missing proofs of our sample complexity bounds in Section 7. Recall that for ease of notation,

$$N_{s,a,h} \coloneqq \begin{cases} N_h^\# \text{ if } (s,a) \in \Omega_h^\# = S_h^\# \times A_h^\# \\ N_h \text{ otherwise.} \end{cases}$$

Proof of Theorem 7. Assume that Q_h^* is rank d and has suboptimality gap $\Delta_{\min}$ (Assumptions 1 and 2), and $S_h^\#, A_h^\#$ are (k,α)-anchor states and actions for Q_h^* for all $h \in [H]$. Let $N_{H-t} = \frac{2(t+1)^2(c')^2k^2\alpha^2\log(2H|S||A|/\delta)}{\Delta_{\min}^2}$, $N_{H-t}^\# = \alpha^2 k^2 N_{H-t}$, where c' satisfies the inequality in Lemma 12, for all $h \in [H]$. We prove the correctness of LR-MCPI with high probability with induction on t that the learned policy $\hat{\pi}_{H-t}$ is an optimal policy with probability at least $1 - \delta(t+1)/H$.

The base case occurs at step $t = 0$ in which case our estimates, $\hat{Q}_H(s,a) = \frac{1}{N_{(s,a,H)}} \sum_{i=1}^{N_{(s,a,H)}} r_H^i(s,a)$ over Ω_H, are only averages of realizations $r_H^i \sim R_H(s,a)$. Since $R_H(s,a)$ has bounded support for

all $(s,a) \in S \times A$, from Hoeffding's inequality (Theorem 30) with our choice of $N_{(s,a,H)}$,

$$|\hat{Q}_H(s,a) - Q^*_H(s,a)| \leq \frac{\Delta_{\min}}{2c'k\alpha} \qquad \forall (s,a) \in \Omega_H$$
$$|\hat{Q}_H(s,a) - Q^*_H(s,a)| \leq \frac{\Delta_{\min}}{2c'k^2\alpha^2} \qquad \forall (s,a) \in \Omega^{\#}_H$$

with probability at least $1 - \delta/H$ because $|\Omega_h| \leq |S||A|$. Step 2 of LR-MCPI gives

$$|\bar{Q}_H(s,a) - Q^*_H(s,a)| \leq \frac{\Delta_{\min}}{2}$$

for all $(s,a) \in S \times A$ from Lemma 12. From Step 3 of LR-MCPI, the identified policy is $\hat{\pi}_H(s) = \text{argmax}_{a \in A} \bar{Q}_H(s,a)$. Assume for sake of contradiction that there exists an $s \in S$ such that $Q^*_H(s, \hat{\pi}_H(s)) < Q^*_H(s, \pi^*_H(s))$. Let $\hat{\pi}_H(s) = a, \pi^*_H(s) = a^*$. Hence,

$$\begin{aligned} Q^*_H(s,a^*) - Q^*_H(s,a) &= Q^*_H(s,a^*) - \bar{Q}_H(s,a) + \bar{Q}_H(s,a) - Q^*_H(s,a) \\ &\leq Q^*_H(s,a^*) - \bar{Q}_H(s,a^*) + \frac{\Delta_{\min}}{2} \\ &\leq \Delta_{\min} \end{aligned}$$

where the first inequality comes from how $\hat{\pi}_H(s)$ is defined and the matrix estimation step. Hence, we reach a contradiction since $Q^*_H(s,a^*) - Q^*_H(s,a)$ is less than the suboptimality gap. Thus, $\hat{\pi}_H(s)$ is an optimal policy. Hence, the base case holds.

Next, let $x \in \{0, \ldots, H-1\}$. Assume that the inductive hypothesis, the policy $\hat{\pi}_{H-x}$ found in Step 4 of LR-MCPI is an optimal policy with probability at least $1 - \delta(x+1)/H$, holds.

Following Step 1 of LR-MCPI, we have $\hat{Q}_{H-x-1}(s,a) = \hat{r}^{\text{cum}}_{H-x-1}(s,a)$, which is an unbiased estimate of $Q^*_{H-x-1}(s,a)$ and also bounded. Hence, from Hoeffding's inequality (Theorem 30), with the choice of $N_{H-x-1} = \frac{2(x+2)^2(c')^2k^2\alpha^2\log(2H|S||A|/\delta)}{\Delta^2_{\min}}, N^{\#}_{H-x-1} = \alpha^2k^2N_{H-x-1}$, it follows that

$$|\hat{Q}_{H-x-1}(s,a) - Q^*_{H-x-1}(s,a)| \leq \frac{\Delta_{\min}}{2c'k\alpha} \qquad \forall (s,a) \in \Omega_{H-x-1}$$
$$|\hat{Q}_{H-x-1}(s,a) - Q^*_{H-x-1}(s,a)| \leq \frac{\Delta_{\min}}{2c'k^2\alpha^2} \qquad \forall (s,a) \in \Omega^{\#}_{H-x-1}$$

with probability $1 - \frac{\delta}{H|S||A|}$. Step 2 of LR-MCPI gives

$$|\bar{Q}_{H-x-1} - Q^*_{H-x-1}|_\infty \leq \frac{\Delta_{\min}}{2}$$

from Lemma 12. From a union bound, it follows that $\hat{\pi}_{H-x}$ is an optimal policy and the above event occur with probability at least $1 - \delta(x+2)/H$. From Step 3 of LR-MCPI, the identified policy is $\hat{\pi}_{H-x-1}(s) = \text{argmax}_{a \in A} \bar{Q}_{H-x-1}(s,a)$. Assume for sake of contradiction that there exists an $s \in S$ such that $Q^*_{H-x-1}(s, \hat{\pi}_{H-x-1}(s)) < Q^*_{H-x-1}(s, \pi^*_{H-x-1}(s))$. Let $\hat{\pi}_{H-x-1}(s) = a, \pi^*_{H-x-1}(s) = a^*$. Hence,

$$\begin{aligned} &Q^*_{H-x-1}(s,a^*) - Q^*_{H-x-1}(s,a) \\ &\qquad = Q^*_{H-x-1}(s,a^*) - \bar{Q}_{H-x-1}(s,a) + \bar{Q}_{H-x-1}(s,a) - Q^*_{H-x-1}(s,a) \\ &\qquad \leq Q^*_{H-x-1}(s,a^*) - \bar{Q}_{H-x-1}(s,a^*) + \frac{\Delta_{\min}}{2} \\ &\qquad \leq \Delta_{\min} \end{aligned}$$

where the first inequality comes from how $\hat{\pi}_{H-x-1}(s)$ is defined and the matrix estimation step. Hence, we reach a contradiction since $Q^*_{H-x-1}(s,a^*) - Q^*_{H-x-1}(s,a)$ is less than the suboptimality gap. Thus, $\hat{\pi}_{H-x-1}(s)$ is an optimal policy, and the inductive step holds for $x+1$. It follows from mathematical induction that the learned policy $\hat{\pi}$ is an optimal policy with probability at least $1-\delta$.

Next, we bound the number of required samples. The number of samples used is

$$\sum_{t=0}^{H-1} (k(|A|+|S|))N_{H-t}(t+1) + k^2 N^{\#}_{H-t}(t+1)$$

where the $t+1$ comes from the length of the rollout. With our choice of N_{H-t}, it follows that

$$\begin{aligned}
&\sum_{t=0}^{H-1} (k(|A|+|S|))N_{H-t}(t+1) \\
&\quad= \sum_{t=0}^{H-1} (k(|A|+|S|)) \frac{2(t+1)^3(c')^2k^2\alpha^2_{H-t}\log(2H|S||A|/\delta)}{\Delta^2_{\min}} + \frac{2(t+1)^3(c')^2k^4\alpha^4_{H-t}\log(2H|S||A|/\delta)}{\Delta^2_{\min}} \\
&\quad\le \left(\frac{2c'^2k^3\alpha^2(|S|+|A|)\log(2H|S||A|/\delta)}{\Delta^2_{\min}} + \frac{k^6\alpha^4c'^2\log(2H|S||A|/\delta}{\Delta^2_{\min}}\right)\sum_{t=0}^{H-1}(t+1)^3 \\
&\quad\in \tilde{O}\left(\frac{k^3\alpha^2(|S|+|A|)H^4}{\Delta^2_{\min}} + \frac{k^6\alpha^4H^4}{\Delta^2_{\min}}\right).
\end{aligned}$$

□

Proof of Theorem 8. This proof follows the same steps as the previous one. Assume that for all ϵ-optimal policies π, Q^π_h is rank d (Assumption 3), and $S^{\#}_h, A^{\#}_h$ are (k,α)-anchor states and actions for $Q^{\hat{\pi}}_h$, where $\hat{\pi}$ is the learned policy from Low Rank Monte Carlo Policy Iteration for all $h \in [H]$. Let $N_{H-t} = \frac{2(t+1)^2(c')^2k^2\alpha^2H^2\log(2H|S||A|/\delta)}{\epsilon^2}, N^{\#}_{H-t} = \alpha^2k^2N_{H-t}$, where c' satisfies the inequality in Lemma 12, for all $h \in [H]$. We prove the correctness of LR-MCPI with high probability with induction on t that the learned policy $\hat{\pi}_{H-t}$ is $\epsilon(t+1)/H$-optimal policy with probability at least $1-\delta(t+1)/H$.

The base case occurs at step $t=0$ in which case our estimates, $\hat{Q}_H(s,a) = \frac{1}{N_{s,a,H}}\sum_{i=1}^{N_{s,a,H}} r^i_H(s,a)$ over Ω_H, are only averages of realizations $r^i_H \sim R_H(s,a)$. Since $R_H(s,a)$ has bounded support for all $(s,a) \in S \times A$, from Hoeffding's inequality (Theorem 30) with our choice of $N_{s,a,H}$,

$$\begin{aligned}
|\hat{Q}_H(s,a) - Q^*_H(s,a)| &\le \frac{\epsilon}{2c'k\alpha H} \qquad \forall (s,a) \in \Omega_h \\
|\hat{Q}_H(s,a) - Q^*_H(s,a)| &\le \frac{\epsilon}{2c'k^2\alpha^2 H} \qquad \forall (s,a) \in \Omega^{\#}_h
\end{aligned}$$

with probability at least $1-\delta/H$ because $|\Omega_h| \le |S||A|$. Step 2 of LR-MCPI gives

$$|\bar{Q}_H(s,a) - Q^*_H(s,a)| \le \frac{\epsilon}{2H}$$

for all $(s,a) \in S \times A$ from Lemma 12. Assume for sake of contradiction that there exists an $s \in S$ such that $Q^*_H(s,\hat{\pi}_H(s)) < Q^*_H(s,\pi^*_H(s)) - \epsilon/H$. Let $\hat{\pi}_H(s) = a, \pi^*_H(s) = a^*$. Hence,

$$\begin{aligned}
Q^*_H(s,a^*) - Q^*_H(s,a) &= Q^*_H(s,a^*) - \bar{Q}_H(s,a) + \bar{Q}_H(s,a) - Q^*_H(s,a) \\
&\le Q^*_H(s,a^*) - \bar{Q}_H(s,a^*) + \frac{\epsilon}{2H} \\
&\le \frac{\epsilon}{H}
\end{aligned}$$

where the first inequality comes from how $\hat{\pi}_H(s)$ is defined and the matrix estimation step. Hence, we reach a contradiction since $Q^*_H(s, a^*) - Q^*_H(s, a)$ is less ϵ/H. Thus, $\bar{Q}_H$ and $\hat{\pi}_H$ are both ϵ/H-optimal, and the base case holds.

Next, let $x \in \{0, \ldots, H-1\}$. Assume that the inductive hypothesis, the policy $\hat{\pi}_{H-x}$ and action-value function estimate $\bar{Q}_{H-x}$ found in Step 3 of LR-MCPI are $\epsilon(x+1)/H$-optimal with probability at least $1 - \delta(x+1)/H$, holds.

Following Step 1 from LR-MCPI, we have $\hat{Q}_{H-x-1}(s, a) = \hat{r}^{\text{cum}}_{H-x-1}(s, a)$, which is bounded and an unbiased estimate of $Q^{\hat{\pi}}(s, a)$ for $\hat{\pi} = \{\hat{\pi}_h\}_{H-x \le h \le H}$, which is an ϵ-optimal policy. Hence, from Hoeffding's inequality (Theorem 30), with the choice of $N_{H-x-1} = \frac{2(x+2)^2 (c')^2 H^2 \alpha^2 k^2 \log(2H|S||A|/\delta)}{\epsilon^2}$, $N^{\#}_{H-x-1} = \alpha^2 k^2 N_{H-x-1}$, it follows that

$$
\begin{aligned}
|\hat{Q}_{H-x-1}(s, a) - Q^{\hat{\pi}}_{H-x-1}(s, a)| &\le \frac{\epsilon}{2c'\alpha k H} && \forall (s, a) \in \Omega_{H-x-1} \\
|\hat{Q}_{H-x-1}(s, a) - Q^{\hat{\pi}}_{H-x-1}(s, a)| &\le \frac{\epsilon}{2c'\alpha^2 k^2 H} && \forall (s, a) \in \Omega^{\#}_{H-x-1}
\end{aligned}
$$

with probability $1 - \frac{\delta}{H|S||A|}$. Step 2 of LR-MCPI gives

$$
\|\bar{Q}_{H-x-1} - Q^{\hat{\pi}}_{H-x-1}\|_\infty \le \frac{\epsilon}{2H}
$$

from Lemma 12. The union bound asserts that the above error guarantee and $\hat{\pi}_{H-x}$ and $\bar{Q}^{\hat{\pi}}_{H-x}$ are $(x+1)\epsilon/H$ holds with probability at least $1 - \delta(x+2)/H$. From step 3 of LR-MCPI, the identified policy is $\hat{\pi}_{H-x-1}(s) = \text{argmax}_{a \in A} \bar{Q}_{H-x-1}(s, a)$. For all $(s, a) \in S \times A$,

$$
\begin{aligned}
|\bar{Q}_{H-x-1}(s, a) - Q^*_{H-x-1}(s, a)| &\le |\bar{Q}_{H-x-1}(s, a) - Q^{\hat{\pi}}_{H-x-1}(s, a)| \\
&\quad + |Q^{\hat{\pi}}_{H-x-1}(s, a) - Q^*_{H-x-1}(s, a)| \\
&\le \frac{\epsilon}{2H} + \left| \mathbb{E}_{s' \sim P_{H-x-1}(\cdot|s,a)} \left[V^{\hat{\pi}}_{H-x}(s') - V^*_{H-x}(s') \right] \right| \\
&\le \frac{\epsilon}{2H} + \left| \mathbb{E}_{s' \sim P_{H-x-1}(\cdot|s,a)} \left[(x+1)\epsilon/H \right] \right| \\
&= \frac{(2x+3)\epsilon}{2H}.
\end{aligned}
$$

Thus, $\bar{Q}_{H-x-1}$ is $\frac{\epsilon(x+2)}{H}$-optimal. It follows from the construction of $\hat{\pi}_{H-x-1}(s)$ that

$$
\bar{Q}_{H-x-1}(s, \hat{\pi}_{H-x-1}(s)) \ge \bar{Q}_{H-x-1}(s, a'),
$$

where $a' = \arg\max_a Q^*_{H-x-1}(s, a)$. Hence, for all $s \in S$,

$$
\begin{aligned}
|V^*_{H-x-1}(s) - V^{\hat{\pi}}_{H-x-1}(s)| &\le |Q^*_{H-x-1}(s, a') - \bar{Q}_{H-x-1}(s, \hat{\pi}_{H-x-1}(s))| \\
&\quad + |\bar{Q}_{H-x-1}(s, \hat{\pi}_{H-x-1}(s)) - Q^{\hat{\pi}}_{H-x-1}(s, \hat{\pi}_{H-x-1}(s))| \\
&\le \frac{(2x+3)\epsilon}{2H} + \frac{\epsilon}{2H} \\
&= \frac{(x+2)\epsilon}{H}.
\end{aligned}
$$

Thus, $\hat{\pi}_{H-x-1}(s)$ and $\bar{Q}_{H-x-1}$ are $(x+2)\epsilon/H$-optimal, and the inductive step holds for $x+1$. It follows from mathematical induction that the learned policy $\hat{\pi}$ and action-value function are ϵ-optimal with probability at least $1 - \delta$.

Next, we bound the number of required samples. The number of samples used is

$$\sum_{t=0}^{H-1}(k(|A|+|S|))N_{H-t}(t+1)+k^2N^{\#}_{H-t}(t+1)$$

where the $t+1$ comes from the length of the rollout. With our choice of N_{H-t}, it follows that

$$\begin{aligned}
&\sum_{t=0}^{H-1}(k(|A|+|S|))N_{H-t}(t+1)\\
&\quad=\sum_{t=0}^{H-1}(k(|A|+|S|))\frac{2(t+1)^3(c')^2k^2\alpha^2H^2\log(2H|S||A|/\delta)}{\epsilon^2}+\frac{2(t+1)^3(c')^2k^6\alpha^4H^2\log(2H|S||A|/\delta)}{\epsilon^2}\\
&\quad\in\tilde{O}\left(\frac{k^3\alpha^2(|S|+|A|)H^6}{\epsilon^2}+\frac{k^6\alpha^4H^6}{\epsilon^2}\right).
\end{aligned}$$

□

Proof. Proof of Theorem 9 This proof follows the same steps as the previous two proofs. Assume that for any ϵ-optimal value function V_{h+1}, the matrix corresponding to $Q'_h=[r_h+[P_hV_{h+1}]]$ is rank d, and $S^{\#}_h, A^{\#}_h$ are (k,α)-anchor states and actions for $\hat{Q}'_h=[r_h+[P_h\hat{V}_{h+1}]]$, where $\hat{V}_{h+1}$ is the learned value function from Low Rank Empirical Value Iteration for all $h\in[H]$. Let $N_{H-t}=\frac{2(t+1)^2(c')^2k^2\alpha^2H^2\log(2H|S||A|/\delta)}{\epsilon^2}, N^{\#}_{H-t}=\alpha^2k^2N_{H-t}$, where c' satisfies the inequality in Lemma 12, and $Q'_h=[r_h+P_h\hat{V}_{h+1}]$ for all ϵ-optimal value fucntions $\hat{V}_{h+1}$ for all $h\in[H]$. We prove the correctness of LR-EVI with high probability with induction on t that

$$\|\bar{Q}_{H-t}-Q^*_{H-t}\|_\infty\le\frac{\epsilon(t+1)}{H},\qquad\|\bar{Q}_{H-t}-Q^{\hat{\pi}}_{H-t}\|_\infty\le\frac{\epsilon(t+1)}{H}$$

where $\bar{Q}_{H-t}$ and $\hat{\pi}_{H-t}$ are the learned Q function and policy with probability at least $1-\delta(t+1)/H$.

The base case occurs at step $t=0$ in which case our estimates, $\hat{Q}_H(s,a)=\frac{1}{N_{s,a,H}}\sum_{i=1}^{N_{s,a,H}}r^i_H(s,a)$ over Ω_H, are only averages of realizations $r^i_H\sim R_H(s,a)$ since $\hat{V}_{H+1}=\vec{0}$. Since $R_H(s,a)$ has bounded support for all $(s,a)\in S\times A$, from Hoeffding's inequality (Theorem 30) with our choice of $N_{s,a,H}$,

$$\begin{aligned}
|\hat{Q}_H(s,a)-Q^*_H(s,a)|&\le\frac{\epsilon}{c'k\alpha H}&&\forall(s,a)\in\Omega_h\\
|\hat{Q}_H(s,a)-Q^*_H(s,a)|&\le\frac{\epsilon}{c'k^2\alpha^2H}&&\forall(s,a)\in\Omega^{\#}_h
\end{aligned}$$

with probability at least $1-\delta/H$ because $|\Omega_h|\le|S||A|$. Step 2 of LR-EVI gives

$$|\bar{Q}_H(s,a)-Q^*_H(s,a)|\le\frac{\epsilon}{H}$$

for all $(s,a)\in S\times A$ from Lemma 12. Since $Q^*_H=Q^{\hat{\pi}}_H$, the base case holds.

Next, let $x\in\{0,\dots,H-1\}$. Assume that the inductive hypothesis, the action-value function estimates $\bar{Q}_{H-x}$ and learned policy $\hat{\pi}_{H-x}$ satisfy

$$\|\bar{Q}_{H-x}-Q^*_{H-x}\|_\infty\le\frac{(x+1)\epsilon}{H},\quad\|\bar{Q}_{H-x}-Q^{\hat{\pi}}_{H-x}\|_\infty\le\frac{(x+1)\epsilon}{H}$$

holds with probability at least $1 - \delta(x+1)/H$. Following Step 1 from LR-EVI, we have

$$\hat{Q}_{H-x-1}(s,a) = \hat{r}_{H-x-1}(s,a) + \mathbb{E}_{s' \sim \hat{P}_{H-x-1}(\cdot|s,a)}[\hat{V}_{H-x}(s')],$$

an unbiased estimate of $Q'_{H-x-1}(s,a) = r_{H-x-1}(s,a) + \mathbb{E}_{s' \sim P_{H-x-1}(\cdot|s,a)}[\hat{V}_{H-x}(s')]$, which is bounded. Hence, from Hoeffding's inequality (Theorem 30), with the choice of $N_{H-x-1} = \frac{(x+2)^2 (c')^2 k^2 H^2 \alpha^2 \log(2H|S||A|/\delta)}{2\epsilon^2}$, $N^{\#}_{H-x-1} = \alpha^2 k^2 N_{H-x-1}$, it follows that

$$\begin{aligned}
|\hat{Q}_{H-x-1}(s,a) - Q'_{H-x-1}(s,a)| &\leq \frac{\epsilon}{2c' k \alpha H} && \forall (s,a) \in \Omega_{H-x-1} \\
|\hat{Q}_{H-x-1}(s,a) - Q'_{H-x-1}(s,a)| &\leq \frac{\epsilon}{2c' k^2 \alpha^2 H} && \forall (s,a) \in \Omega^{\#}_{H-x-1}
\end{aligned}$$

with probability $1 - \frac{\delta}{H|S||A|}$. Step 2 of LR-EVI gives

$$|\bar{Q}_{H-x-1} - Q'_{H-x-1}|_\infty \leq \frac{\epsilon}{H}$$

from Lemma 12. The union bound asserts that the above error guarantee and $\bar{Q}_{H-x'}$ is close to $Q^*_{H-x'}$ and $Q^{\hat{\pi}}_{H-x'}$ for $x \in [x]$ holds with probability at least $1 - \delta(x+2)/H$. Hence, for all $(s,a) \in S \times A$,

$$\begin{aligned}
|\bar{Q}_{H-x-1}(s,a) - Q^*_{H-x-1}(s,a)| &\leq |\bar{Q}_{H-x-1}(s,a) - Q'_{H-x-1}(s,a)| + |Q'_{H-x-1}(s,a) - Q^*_{H-x-1}(s,a)| \\
&\leq \frac{\epsilon}{H} + |\mathbb{E}_{s' \sim P_{H-x-1}(\cdot|s,a)}[\max_{a \in A} \bar{Q}_{H-x}(s',a') - V^*_{H-x}(s')]| \\
&\leq \frac{\epsilon}{H} + |\mathbb{E}_{s' \sim P_{H-x-1}(\cdot|s,a)}[(x+1)\epsilon/H]| \\
&= \frac{(x+2)\epsilon}{H}
\end{aligned}$$

Thus, $\bar{Q}_{H-x-1}$ is $(x+2)\epsilon/H$-optimal. Next, we note that

$$\begin{aligned}
|\bar{Q}_{H-x-1}(s,a) - Q^{\hat{\pi}}_{H-x-1}(s,a)| &\leq |\bar{Q}_{H-x-1}(s,a) - Q'_{H-x-1}(s,a)| + |Q'_{H-x-1}(s,a) - Q^{\hat{\pi}}_{H-x-1}(s,a)| \\
&\leq \frac{\epsilon}{H} + |\mathbb{E}_{s' \sim P_{H-x-1}(\cdot|s,a)}[\max_{a \in A} \bar{Q}_{H-x}(s',a') - V^{\hat{\pi}}_{H-x}(s')]| \\
&\leq \frac{\epsilon}{H} + |\mathbb{E}_{s' \sim P_{H-x-1}(\cdot|s,a)}[(x+1)\epsilon/H]| \\
&= \frac{(x+2)\epsilon}{H}
\end{aligned}$$

where the third inequality holds because

$$\begin{aligned}
|\max_{a \in A} \bar{Q}_{H-x}(s',a') - V^{\hat{\pi}}_{H-x}(s')| &\leq |\mathbb{E}_{a' \sim \hat{\pi}_{H-x}(s')}[\bar{Q}_{H-x}(s',a')] - \mathbb{E}_{a' \sim \hat{\pi}_{H-x}(s')}[Q^{\hat{\pi}}_{H-x}(s',a')] \\
&\leq \|\hat{Q}_{H-x} - Q^{\hat{\pi}}_{H-x}\|_\infty \\
&\leq \frac{(x+1)\epsilon}{H}
\end{aligned}$$

from the induction hypothesis, and the inductive step holds for $x+1$. It follows from mathematical induction (and the triangle inequality) that the learned policy $\hat{\pi}$ and action-value function are 2ϵ and ϵ-optimal with probability at least $1-\delta$. Scaling N_h by a factor of four results in learning an ϵ-optimal policy with probability at least $1-\delta$ without changing the sample complexity's dependence on $|S|, |A|, H$, or ϵ.

Next, we bound the number of required samples. The number of samples used is

$$\sum_{t=0}^{H-1}(k(|A|+|S|))N_{H-t}+k^2N^{\#}_{H-t}.$$

Note that there is no $(t+1)$ term as samples are single transitions instead of rollouts. With our choice of N_{H-t}, it follows that

$$\begin{aligned}
&\sum_{t=0}^{H-1}(k(|A|+|S|))N_{H-t}\\
&\qquad=\sum_{t=0}^{H-1}k(|A|+|S|)\frac{8(t+1)^2(c')^2k^2\alpha^2H^2\log(2H|S||A|/\delta)}{\epsilon^2}+\frac{8(t+1)^2(c')^2k^6\alpha^4H^2\log(2H|S||A|/\delta)}{\epsilon^2}\\
&\qquad\in\tilde{O}\left(\frac{k^3\alpha^2(|S|+|A|)H^5}{\epsilon^2}+\frac{k^6\alpha^4H^5}{\epsilon^2}\right).
\end{aligned}$$

□

J Proofs for Approximately Low Rank Models

We first present the proof of Proposition 13, which shows that if the reward function and transition kernel are low rank, then for any value function estimate $\hat{V}_{h+1}$, $r_h+[P_h\hat{V}_{h+1}]$ has rank upper bounded by d.

Proof of Proposition 13. Let $\xi_R,\xi_P,r_{h,d},[P_{h,d}\hat{V}_{h+1}]$ be defined as in Section 7.3. Then, for all $(s,a,h)\in S\times A\times[H]$,

$$\begin{aligned}
&[r_{h,d}+P_{h,d}\hat{V}_{h+1}](s,a)-[r_h+P_h\hat{V}_{h+1}](s,a)|\\
&\qquad\le|[r_h-r_{h,d}](s,a)|+|[(P_{h,d}-P_h)\hat{V}_{h+1}](s,a)|\\
&\qquad=\xi_R+|\sum_{s'\in S}\hat{V}_{h+1}(s')(P_{h,d}(s'|s,a)-P_h(s'|s,a))|\\
&\qquad\le\xi_R+(H-h)|\sum_{s'\in S}(P_{h,d}(s'|s,a)-P_h(s'|s,a))|\\
&\qquad=\xi_R+(H-h)2d_{\mathrm{TV}}(P_h(\cdot|s,a),P_{h,d}(\cdot|s,a))_{\mathrm{TV}}\\
&\qquad=\xi_R+(H-h)\xi_P
\end{aligned}$$

since $\hat{V}_{h+1}(s)\in[0,H-h-1]$. □

We next prove that the learned policy's error is additive with respect to the approximation error.

Proof. Proof of Theorem 14 This proof follows the same steps as the proof of Theorem 9 while accounting for the approximation error. Assume that we have a (d,ξ_R,ξ_P)-approximately low-rank MDP. Let $S^{\#}_h,A^{\#}_h$ be (k,α)-anchor states and actions, c' be a constant that satisfies the inequality in Lemma 12, $N_{H-t}=\frac{2(t+1)^2(c')^2k^2\alpha^2H^2\log(2H|S||A|/\delta)}{\epsilon^2}$, $N^{\#}_{H-t}=\alpha^2k^2N_{H-t}$, and $Q'_h=[r_h+P_h\hat{V}_{h+1}]$

for all ϵ-optimal value functions $\hat{V}_{h+1}$ for all $h \in [H]$. We prove the correctness of LR-EVI with high probability with induction on t that

$$\|\bar{Q}_{H-t} - Q^*_{H-t}\|_\infty, \|\bar{Q}_{H-t} - Q^{\hat{\pi}}_{H-t}\|_\infty \leq (t+1)\epsilon/H + \sum_{i=0}^{t} (c'k^2\alpha^2 + 1)\,(\xi_R + i\xi_P)$$

where $\bar{Q}_{H-t}$ and $\hat{\pi}_{H-t}$ are the learned Q function and policy with probability at least $1-\delta(t+1)/H$ for each $t \in \{0, \ldots, H-1\}$.

The base case occurs at step $t = 0$ in which case our estimates, $\hat{Q}_H(s,a) = \frac{1}{N_{s,a,H}} \sum_{i=1}^{N_{s,a,H}} r_{H,i}(s,a)$ over Ω_H, are only averages of realizations $r^i_H \sim R_H(s,a)$ since $\hat{V}_{H+1} = \vec{0}$. Since $R_H(s,a)$ has bounded support for all $(s,a) \in S \times A$, from Hoeffding's inequality (Theorem 30) with our choice of $N_{s,a,H}$

$$\begin{aligned} |\hat{Q}_H(s,a) - Q^*_H(s,a)| &\leq \frac{\epsilon}{c'k\alpha H} && \forall (s,a) \in \Omega_h \\ |\hat{Q}_H(s,a) - Q^*_H(s,a)| &\leq \frac{\epsilon}{c'k^2\alpha^2 H} && \forall (s,a) \in \Omega^\#_h \end{aligned}$$

with probability at least $1 - \delta/H$ because $|\Omega_h| \leq |S||A|$. Under the event above, it follows that $|\hat{Q}_H(s,a) - Q^*_{H,d}(s,a)| \leq \frac{\epsilon}{c'_H\alpha^2k^2H} + \xi_R$ or $|\hat{Q}_H(s,a) - Q^*_{H,d}(s,a)| \leq \frac{\epsilon}{c'_H\alpha kH} + \xi_R$ for all $(s,a) \in \Omega_H$. Step 2 of LR-EVI gives

$$|\bar{Q}_H(s,a) - Q^*_{H,d}(s,a)| \leq \frac{\epsilon}{H} + Ck^2\alpha^2\xi_R$$

for all $(s,a) \in S \times A$ from Lemma 12 for some positive constant C. By definition of the approximation error,

$$|\bar{Q}_H(s,a) - Q^*_H(s,a)| \leq \frac{\epsilon}{H} + (Ck^2\alpha^2 + 1)\xi_R \qquad \forall (s,a) \in S \times A.$$

Since $Q^*_H = Q^{\hat{\pi}}_H$, the base case holds.

Next, let $x \in \{0, \ldots, H-1\}$. Assume that the inductive hypothesis, the action-value function estimates $\bar{Q}_{H-x}$ and learned policy $\hat{\pi}_{H-x}$ satisfy

$$\|\bar{Q}_{H-x} - Q^*_{H-x}\|_\infty \leq \frac{(x+1)\epsilon}{H}, \quad \|\bar{Q}_{H-x} - Q^{\hat{\pi}}_{H-x}\|_\infty \leq \frac{(x+1)\epsilon}{H} + \sum_{i=0}^{x} (Ck^2\alpha^2 + 1)\,(\xi_R + i\xi_P)$$

holds with probability at least $1 - \delta(x+1)/H$. At step $x+1$, following Step 1 from LR-EVI, we have

$$\hat{Q}_{H-x-1}(s,a) = \hat{r}_{H-x-1}(s,a) + \mathbb{E}_{s' \sim \hat{P}_{H-x-1}(\cdot|s,a)}[\hat{V}_{H-x}(s')],$$

an unbiased estimate of $Q'_{H-x-1}(s,a) = r_{H-x-1}(s,a) + \mathbb{E}_{s' \sim P_{H-x-1}(\cdot|s,a)}[\hat{V}_{H-x}(s')]$. Furthermore, $\hat{Q}_{H-x-1}(s,a) \in [0, x+2]$ is a bounded random variable because of bounded rewards. Hence, from Hoeffding's inequality (Theorem 30), with the choice of $N_{H-x-1} = \frac{(x+2)^2(c')^2k^2\alpha^2H^2\log(2H|S||A|/\delta)}{2\epsilon^2}$, $N^\#_{H-x-1} = \alpha^2k^2N_{H-x-1}$, it follows that

$$\begin{aligned} |\hat{Q}_{H-x-1}(s,a) - Q'_{H-x-1}(s,a)| &\leq \frac{\epsilon}{2c'k\alpha H} && \forall (s,a) \in \Omega_{H-x-1} \\ |\hat{Q}_{H-x-1}(s,a) - Q'_{H-x-1}(s,a)| &\leq \frac{\epsilon}{2c'k^2\alpha^2 H} && \forall (s,a) \in \Omega^\#_{H-x-1} \end{aligned}$$

with probability $1-\frac{\delta}{H|S||A|}$. Under the event above, it follows that $|\hat{Q}_{H-x-1}(s,a)-Q'_{H-x-1}(s,a)| \leq \frac{\epsilon}{c'k^2\alpha^2 H} + \xi_R + (H-x-1)\xi_P$ for all $(s,a) \in \Omega_{H-x-1}$ where $Q'_{h,d} = r_{h,d} + [P_{h,d}\hat{V}_{h+1}]$. Step 2 of LR-EVI gives

$$|\bar{Q}_{H-x-1} - Q'_{H-x-1,d}|_\infty \leq \frac{\epsilon}{H} + Ck^2\alpha^2\left(\xi_R + (H-x-1)\xi_P\right)$$

from Lemma 12 for some positive constant C. The union bound asserts that the above error guarantee holds with probability at least $1-\delta(x+2)/H$. Hence, for all $(s,a) \in S \times A$,

$$\begin{aligned}
&|\bar{Q}_{H-x-1}(s,a) - Q^*_{H-x-1}(s,a)| \\
&\leq |\bar{Q}_{H-x-1}(s,a) - Q'_{H-x-1,d}(s,a)| + |Q'_{H-x-1,d}(s,a) - Q'_{H-x-1}(s,a)| \\
&+ |Q'_{H-x-1}(s,a) - Q^*_{H-x-1}(s,a)| \\
&\leq \frac{\epsilon}{H} + c'k^2\alpha^2\left(\xi_R + (H-x-1)\xi_P\right) \\
&+ \xi_R + (H-x-1)\xi_P \\
&+ |\mathbb{E}_{s'\sim P_{H-x-1}(\cdot|s,a)}[\max_{a\in A}\bar{Q}_{H-x}(s',a') - V^*_{H-x}(s')]| \\
&\leq \frac{\epsilon}{H} + (1+c'k^2\alpha^2)\left(\xi_R + (H-x-1)\xi_P\right) \\
&+ |\mathbb{E}_{s'\sim P_{H-x-1}(\cdot|s,a)}[(x+1)\epsilon/H + \sum_{i=0}^{x}(Ck^2\alpha^2+1)\left(\xi_R + i\xi_P\right)]| \\
&= \frac{(x+2)\epsilon}{H} + \sum_{i=0}^{x+1}(Ck^2\alpha^2+1)\left(\xi_R + i\xi_P\right)].
\end{aligned}$$

With a similar argument, it follows that for all $(s,a) \in S \times A$,

$$\bar{Q}_{H-x-1}(s,a) - Q^{\hat{\pi}}_{H-x-1}(s,a)| \leq \frac{(x+2)\epsilon}{H} + \sum_{i=0}^{x+1}(Ck^2\alpha^2+1)\left(\xi_R + i\xi_P\right)].$$

Thus, the inductive step holds for $x+1$, and from mathematical induction, the lemma holds.

Choosing $t = H-1$ proves the correctness of the algorithm. Next, we bound the number of required samples. The number of samples used is the same as in the proof of Theorem 9, which implies a sample complexity of

$$\tilde{O}\left(\frac{k^3\alpha^2(|S|+|A|)H^5}{\epsilon^2} + \frac{k^6\alpha^4H^5}{\epsilon^2}\right).$$

□

K Proofs for Continuous MDPs

We next present the proofs of the results in Section A.1, starting with our procedure on how we obtain samples/rollouts from the discretized MDP.

Using the generative model, we simulate trajectories from M^β with the following procedure: to sample a trajectory from M^β following policy π starting at (s,a,h), first sample a state s' from $P_h(\cdot|s,a)$. Then, we take the closest discretized state s'_β to s' , i.e., $s'_\beta = \arg\min_{s\in S}\|s-s'\|_2$, to be the observed state in the trajectory. The generative model is then used to sample from $P_{h+1}(\cdot|s'_\beta, \pi(s'_\beta))$, and we repeat until the end of the horizon to obtain a trajectory from P^β_h using the generative model on the original MDP. Lemma 25 asserts the correctness of this procedure.

Lemma 25. *Let $\tau_h^\pi(s_h, a_h)$ be a rollout obtained with the procedure detailed above starting at state-action pair (s_h, a_h) at step h with policy π, $\tau_{\beta,h}^\pi(s_h, a_h)$ be a rollout following policy π from the discretized MDP starting at state-action pair (s_h, a_h) at step h, and τ be a realization of a rollout following policy π from the discretized MDP starting at state-action pair (s_h, a_h) at step h. Then,*

$$\mathbb{P}(\tau_h(s_h, a_h)) = \tau) = \mathbb{P}(\tau_h^\beta(s_h, a_h) = \tau).$$

Proof of Lemma 25. Let $\tau = (s_{h+1}, \pi(s_{h+1}), \ldots, s_H, \pi(s_H))$. From the Markov Property and the procedure defined above, it follows that

$$\begin{aligned}
\mathbb{P}(\tau_h(s_h, a_h)) = \tau) &= \int_{\{s':|s_{h+1}-s'|_2 \leq \beta\}} P(s'|s_h, a_h)\mathrm{d}s' \Pi_{i=h+1}^{H-1} \int_{\{s':|s_{i+1}-s'|_2 \leq \beta\}} P(s'|s_i, \pi(s_i))\mathrm{d}s' \\
&= P_h^\beta(s_{h+1}|s_h, a_h)\Pi_{i=h+1}^{H-1} P_h^\beta(s_{i+1}|s_i, \pi(s_i)) \\
&= \mathbb{P}(\tau_h^\beta(s_h, a_h) = \tau).
\end{aligned}$$

□

We next present the proof of Lemma 15, which allows us to use Q_h^β to estimate Q_h^*.

Proof of Lemma 15. We prove this lemma via induction. For $h = H$, by construction of the β-nets, for any $s \in S, a \in A$ and $s' \in S^\beta, a' \in A^\beta$ such that $\|s - s'\|_2 \leq \beta, \|a - a'\|_2 \leq \beta$,

$$|Q_H^*(s, a) - Q_H^\beta(s', a')| \leq 2L\beta$$

because Q_H^* is L-Lipschitz. For all $s \in S$ and $s' \in S^\beta$ such that $\|s - s'\|_2 \leq \beta$, let $a_{max} = \mathrm{argmax}_{a \in A} Q_H^*(s, a)$ and $d_A(a)$ be the function that maps the action a to the closest action in A^β, which is at most β away. It follows that

$$\begin{aligned}
|V_H^*(s) - V_H^\beta(s')| &= |Q_H^*(s, a_{max}) - \max_{a^* \in A^\beta} Q_H^\beta(s', a^*)| \\
&\leq |Q_H^*(s, a_{max}) - Q_H^\beta(s', d(a_{max}))| \\
&\leq 2L\beta
\end{aligned}$$

where the first inequality comes from $Q_H^*(s, a_{max})$ being an upperbound of $Q_H^\beta(s', a')$ and the max operator, and the second inequality comes from Q_H^* being Lipschitz. Next, assume that for any $s \in S$ and $s' \in S^\beta$ such that $\|s - s'\|_2 \leq \beta$, $|V_{H-t+1}^*(s) - V_{H-t+1}^\beta(s')| \leq 2(H - t + 1)L\beta$. Let $d_S(s)$ be the function that maps the state s to the closest state in S^β, which is at most β away. For any

$s \in S, a \in A$ and $s' \in S^\beta, a' \in A^\beta$ such that $\|s - s'\|_2 \leq \beta, \|a - a'\|_2 \leq \beta$,

$$
\begin{aligned}
&|Q^*_{H-t}(s,a) - Q^\beta_{H-t}(s',a')| \\
&\leq |Q^*_{H-t}(s,a) - Q^*_{H-t}(s',a')| + |Q^*_{H-t}(s',a') - Q^\beta_{H-t}(s',a')| \\
&\leq 2L\beta + |\mathbb{E}_{s^* \sim P(\cdot|s',a')}[V^*_{H-t+1}(s^*)] - \mathbb{E}_{s^* \sim P^\beta(\cdot|s',a')}[V^\beta_{H-t+1}(s^*)]| \\
&= 2L\beta + |\int_{s^* \in S} P_{H+1}(s^*|s',a') V^*_{H-t+1}(s^*) \mathrm{d}s^* - \sum_{s^* \in S^\beta} P^\beta_{H+1}(s^*|s',a') V^\beta_{H-t+1}(s^*) \\
&= 2L\beta + |\int_{s^* \in S} P_{H+1}(s^*|s',a') V^*_{H-t+1}(s^*) \mathrm{d}s^* \\
&\quad - \sum_{s^* \in S^\beta} \int_{\{s'' \in S : |s^* - s''|_2 \leq \beta\}} V^\beta_{H-t+1}(s^*) P_h(s''|s',a') \mathrm{d}s^*| \\
&= 2L\beta + |\int_{s^* \in S} P_{H+1}(s^*|s',a') V^*_{H-t+1}(s^*) \mathrm{d}s^* - \int_{s^* \in S} P_{H+1}(s^*|s',a') V^\beta_{H-t+1}(d_S(s^*)) \mathrm{d}s^*| \\
&\leq 2L\beta + |\mathbb{E}_{s^* \sim P(\cdot|s',a')} 2(H-t+1)L\beta| \\
&= 2(H-t)L\beta
\end{aligned}
$$

where the fourth line comes from the definition of P^β_h. For all $s \in S$ and $s' \in S^\beta$ such that $\|s - s'\|_2 \leq \beta$, let $a_{max} = \mathrm{argmax}_{a \in A} Q^*_{H-t}(s,a)$. It follows that

$$
\begin{aligned}
|V^*_{H-t}(s) - V^\beta_{H-t}(s')| &= |Q^*_{H-t}(s, a_{max}) - Q^\beta_{H-t}(s', a'_{max})| \\
&\leq |Q^*_{H-t}(s, a_{max}) - Q^\beta_{H-t}(s', d_A(a_{max}))| \\
&\leq 2(H-t)L\beta
\end{aligned}
$$

Thus, from induction, for any $s \in S, a \in A$ and $s' \in S^\beta, a' \in A^\beta$ such that $\|s - s'\|_2 \leq \beta, \|a - a'\|_2 \leq \beta$, for all $h \in [H]$,

$$
|Q^*_h(s,a) - Q^\beta_h(s',a')| \leq 2L(H-h+1)\beta, \quad |V^*_h(s,a) - V^\beta_h(s',a')| \leq 2L(H-h+1)\beta.
$$

□

To prove the desired sample complexity bounds, we first state a lemma on covering numbers that upperbounds the number of points required for our β-nets.

Lemma 26 (Theorem 14.2 from [52])**.** *Let $\Theta \subset \mathbb{R}^n$. Then,*

$$
N(\Theta, \epsilon) \leq \left(\frac{3}{\epsilon}\right)^n \frac{Vol(\Theta)}{Vol(B)}
$$

where $N(\Theta, \epsilon)$ is the covering number of Θ, and B is the unit norm ball in $\mathbb{R}^n$.

We next present the sample complexity bound of LR-MCPI when M^β satisfies Assumption 3 and its proof.

Theorem 27. *Let $Q^\pi_{h,\beta} = [Q^\pi_h(s,a)]_{(s,a) \in S^\beta \times A^\beta}$, the action-value function of policy π at step h on only the discretized state-action pairs. After discretizing the continuous MDP, let Assumption 3 hold on M^β. Furthermore, assume that $S^\#_h, A^\#_h$ are (k,α)-anchor states and actions for $Q^{\pi,\beta}_h$ for all $h \in [H]$. Let $\bar{Q}_h$ be the action-value function estimates that Low Rank Monte Carlo Policy Iteration*

for Step 1 return for all $h \in [H]$ when run on M^β. For any $s \in S, a \in A$ and $s' \in S^\beta, a' \in A^\beta$ such that $\|s - s'\|_2 \leq \beta, \|a - a'\|_2 \leq \beta$ and $h \in [H]$,

$$|\bar{Q}_h(s', a') - Q^*_h(s, a)| \leq \epsilon$$

with probability at least $1 - \delta$ when $\beta = \frac{\epsilon}{4LH}$, $N_{H-t} = \frac{8(t+1)^2(c')^2H^2k^2\alpha^2\log(2H|S||A|/\delta)}{\epsilon^2}$, $N^{\#}_{H-t} = \alpha^2k^2N_{H-t}$, and c' satisfies the inequality in Lemma 12 for all $t \in \{0, \dots H-1\}$. Furthermore, at most $\tilde{O}\left(\frac{k^3\alpha^2H^{n+6}}{\epsilon^{n+2}Vol(B)}\right)$ number of samples are required with the same probability where B is the unit norm ball in $\mathbb{R}^n$.

Proof of Theorem 27. After discretizing the continuous MDP to get M^β for $\beta = \frac{\epsilon}{4LH}$, we note that $|S^\beta|, |A^\beta| \in O(\frac{H^n}{\epsilon^n Vol(B)})$ from Lemma 26. Since the required assumptions for Theorem 8 hold on M^β, it follows that each $\bar{Q}_h$ is $\epsilon(H-h+1)/H$-optimal for all $h \in [H]$ on M^β when running LR-MCPI with $N_{H-t} = \frac{8(t+1)^2(c')^2H^2k^2\alpha^2\log(2H|S||A|/\delta)}{\epsilon^2}, N^{\#}_{H-t} = \alpha^2k^2N_{H-t}$ using at most $\tilde{O}\left(\frac{k^3\alpha^2H^{n+6}}{\epsilon^{n+2}Vol(B)}\right)$ samples with probability at least $1-\delta$. Since $\beta = \frac{\epsilon}{4LH}$, from Lemma 15, for any $s \in S, a \in A$ and $s' \in S^\beta, a' \in A^\beta$ such that $\|s-s'\|_2 \leq \beta, \|a-a'\|_2 \leq \beta$ and for all $t \in \{0, \dots H-1\}$,

$$\begin{aligned}
|\bar{Q}_{H-t}(s', a') - Q^*_{H-t}(s, a)| &\leq |\bar{Q}_{H-t}(s', a') - Q^\beta_{H-t}(s', a')| + |Q^\beta_{H-t}(s', a') - Q^*_{H-t}(s, a)| \\
&\leq \frac{\epsilon}{2} + 2L(t+1)\beta \\
&\leq \epsilon.
\end{aligned}$$

Hence, an ϵ-optimal Q function on the continuous space is $\bar{Q}^c_h(s, a) = \bar{Q}_h(s', a')$, where (s', a') is the discretized state-action pair closest to (s, a). □

Proof of Theorem 16. After discretizing the continuous MDP to get M^β for $\beta = \frac{\epsilon}{4LH}$, we note that $|S^\beta|, |A^\beta| \in O(\frac{H^n}{\epsilon^n Vol(B)})$ from Lemma 26. Since Assumption 4 holds on M^β, from Theorem 9, it follows that each $\bar{Q}_h$ is $\epsilon/2$-optimal for all $h \in [H]$ on M^β when running LR-EVI with $N_{H-t} = \frac{4(t+1)^2(c')^2k^2\alpha^2H^2\log(2H|S||A|/\delta)}{\epsilon^2}, N^{\#}_{H-t} = \frac{4(t+1)^2(c')^2k^4\alpha^4H^2\log(2H|S||A|/\delta)}{\epsilon^2}$ using at most $\tilde{O}\left(\frac{k^3\alpha^2H^{n+5}}{\epsilon^{n+2}Vol(B)}\right)$ samples with probability at least $1-\delta$. Since $\beta = \frac{\epsilon}{4LH}$, from Lemma 15, for any $s \in S, a \in A$ and $s' \in S^\beta, a' \in A^\beta$ such that $\|s - s'\|_2 \leq \beta, \|a - a'\|_2 \leq \beta$ and for all $t \in \{0, \dots H-1\}$,

$$\begin{aligned}
|\bar{Q}_{H-t}(s', a') - Q^*_{H-t}(s, a)| &\leq |\bar{Q}_{H-t}(s', a') - Q^\beta_{H-t}(s', a')| + |Q^\beta_{H-t}(s', a') - Q^*_{H-t}(s, a)| \\
&\leq \frac{\epsilon}{2} + 2L(t+1)\beta \\
&\leq \epsilon.
\end{aligned}$$

Hence, an ϵ-optimal Q function on the continuous space is $\bar{Q}^c_h(s, a) = \bar{Q}_h(s', a')$, where (s', a') is the discretized state-action pair closest to (s, a). □

L Proofs for Infinite-Horizon Discounted MDPs

In this section, we present the omitted proofs from Appendix A.2. We first prove that for any estimate of the value function, $r + P\hat{V}_t$ has rank that is at most d.

Proof of Proposition 17. Let MDP $M = (S, A, P, R, \gamma)$ satisfy Assumption 7. For the Tucker rank $(|S|, |S|, d)$ case,it follows that

$$\begin{aligned}
r(s, a) + \gamma\mathbb{E}_{s' \sim P(\cdot|s,a)}[\hat{V}(s')] &= \textstyle\sum_{i=1}^d W(s, i)V(a, i) + \gamma \sum_{s' \in S} \sum_{i=1}^d U(s', s, i)V(a, i)\hat{V}(s') \\
&= \textstyle\sum_{i=1}^d V(a, i) \left(W(s, i) + \gamma \sum_{s' \in S} U(s', s, i)\hat{V}(s')\right)
\end{aligned}$$

Thus, $r + \gamma[P\hat{V}]$ has rank upper bounded by d, and the Tucker rank $(|S|, d, |A|)$ case follows the same steps. □

To prove the correctness of LR-EVI for the infinite-horizon setting, we first show that the error of the Q-function decreases in each iteration, Lemma 19.

Proof of Lemma 19. Let $Q'_{t+1} = r + \gamma P\bar{V}_t$ and $t \in [T-1]$. From proposition 17, Q'_{t+1} has rank at most d for all $t \in [T-1]$ Following step 1 from LR-EVI, $\hat{Q}_{t+1}(s,a) = \frac{1}{N_{t+1}} \sum_{i=1}^{N_{t+1}} R(s,a) + \gamma \bar{V}_t(s'_i)$ for all $(s,a) \in \Omega_{t+1}$. Hence, $\hat{Q}_{t+1}(s,a)$ is an unbiased estimate of $Q'_{t+1}(s,a)$ for all $(s,a) \in \Omega_t$. Furthermore, because of bounded rewards, $\hat{Q}_{t+1}(s,a) \in [0, \frac{1}{1-\gamma}]$ is a bounded random variable. With our choice of $N_{t+1} = \frac{2(c')^2 k^2 \alpha^2 \log(2T|S||A|/\delta)}{(1-\gamma)^4 B_t^2}, N_{t+1}^{\#} = N_{t+1}\alpha^2, k^2$, it follows from Hoeffding's inequality that for all $(s,a) \in \Omega_{t+1}$,

$$|\hat{Q}_{t+1}(s,a) - Q'_{t+1}(s,a)| \leq \frac{(1-\gamma)B_t}{2c'\alpha k} \quad \forall (s,a) \in \Omega_{t+1}$$
$$|\hat{Q}_{t+1}(s,a) - Q'_{t+1}(s,a)| \leq \frac{(1-\gamma)B_t}{2c'\alpha^2 k^2} \quad \forall (s,a) \in \Omega_{t+1}^{\#}$$

with probability at least $1 - \frac{\delta}{T|S||A|}$. Step 2 of LR-EVI gives that for all $(s,a) \in S \times A$

$$|\bar{Q}_{t+1}(s,a) - Q'_{t+1}(s,a)| \leq \frac{(1-\gamma)B_t}{2}$$

from Lemma 12. Hence, for all $(s,a) \in S \times A$,

$$\begin{aligned}
|\bar{Q}_{t+1}(s,a) - Q^*(s,a)| &\leq |\bar{Q}_{t+1}(s,a) - Q'_{t+1}(s,a)| + |Q'_{t+1}(s,a) - Q^*(s,a)| \\
&\leq \frac{(1-\gamma)B_t}{2} + |\gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}[\bar{V}_t(s') - V^*(s')]| \\
&\leq \frac{(1-\gamma)B_t}{2} + \gamma B_t \\
&= \frac{(1+\gamma)B_t}{2}.
\end{aligned}$$

From step 4 of LR-EVI, the estimate of the value function is defined as $\bar{V}_{t+1}(s) = \max_{a \in A} \bar{Q}_{t+1}(s,a)$ for all $s \in S$. It follows that $|\bar{V}_{t+1}(s) - V^*(s)| \leq \frac{(1+\gamma)B_t}{2}$. □

Proof of Theorem 18.] Since the value function estimate is initialized as the zero vector, $|\bar{V}_0 - V^*|_\infty \leq \frac{1}{1-\gamma} = B_0$. We prove the correctness of this algorithm by repeatedly applying Lemma 19 T times. From the union bound, the $\bar{Q}_T$ that the algorithm returns satisfies

$$|\bar{Q}_T(s,a) - Q^*(s,a)| \leq \left(\frac{1+\gamma}{2}\right)^T \left(\frac{1}{1-\gamma}\right)$$

with probability at least $1 - \delta$. With $T = \frac{\ln(\epsilon(1-\gamma))}{\ln(\frac{1+\gamma}{2})}$, it follows that $(\frac{1+\gamma}{2})^T(\frac{1}{1-\gamma}) = \epsilon$, so

$$|\bar{Q}_T(s,a) - Q^*(s,a)| \leq \epsilon$$

with probability at least $1 - \delta$. Note that since B_t is strictly decreasing with respect to t, it follows that $N_t = \frac{2(c')^2 \alpha^2 k^2 \log(2T/\delta)}{(1-\gamma)^4 B_{t-1}^2}, N_t^{\#} = \alpha^2 k^2$ are strictly increasing with respect to t. Furthermore,

since $B_{T-1} > \epsilon$, $N_t \in \tilde{O}\left(\frac{\alpha^2 k^2}{(1-\gamma)^4 \epsilon^2}\right)$ for all $t \in [T]$. It follows that the sample complexity of the algorithm is

$$\tilde{O}\left(\frac{\alpha^2 k^3 (|S| + |A|)}{\epsilon^2 (1-\gamma)^{-4}} + \frac{\alpha^4 k^6}{\epsilon^2 (1-\gamma)^{-4}}\right).$$

□

M Proofs for LR-EVI with Matrix Estimation using Nuclear Norm Regularization

In this subsection, we present the omitted proofs from Section A.3. We first prove a lemma that gives us the matrix estimation guarantee in our desired form.

Lemma 28. *Assume that for any ϵ-optimal value function $\hat{V}_{h+1}$, the matrix corresponding to $[r_h + [P_h \hat{V}_{h+1}]]$ is rank d, μ-incoherent, and has condition number bounded by κ. Then, for*

$$p_h = \frac{\mu^3 d^2 \kappa^2 H^4 C_{cvx}^2 \log(n)}{\epsilon^2 n},$$

where C_{cvx} is defined as in Theorem 20 with probability $1 - O(n^{-3})$, we have

$$\|\hat{Q}_h - r_h + [P_h \hat{V}_{h+1}]\|_\infty \leq \frac{\epsilon}{H}.$$

Proof of Lemma 28. Since $Q_h(s,a)$ is bounded by $H - h$, the estimates in Step 2 of LR-EVI-cvx are bounded random variables. Hence, they are unbiased with sub-Gaussian parameter $H - h$ [47]. Let $Q' = r_h + [P_h \hat{V}_{h+1}]$. From Theorem 20, with probability $1 - O(n^{-3})$,

$$\|\bar{Q}_h - Q'_h\|_\infty \leq \frac{C_{cvx}(H-h)}{\sigma_r(Q'_h)} \sqrt{\frac{\mu n \log n}{p_h}} \|Q'\|_\infty.$$

Let Q' have singular value decomposition $U \Sigma V^T$. Then, for $(s,a) \in S \times A$,

$$\begin{aligned}
|Q'_h(s,a)| &= |e_s^T U \Sigma V^T e_a| \\
&\leq \|U(s)\|_2 \|\Sigma\|_{op} \|V(a)\|_2 \\
&\leq \frac{\mu d}{n} \sigma_1(Q'_h) \\
&\leq \frac{\mu d \kappa}{n} \sigma_d(Q'_h)
\end{aligned}$$

where the second inequality comes from incoherence and the last inequality comes from bounded condition number. Plugging this inequality into the application of Theorem 20 gives

$$\|\bar{Q}_h - Q'_h\|_\infty \leq \mu d \kappa C_{cvx}(H-h) \sqrt{\frac{\mu \log n}{p_h n}}.$$

From our choice of p_h, we get the desired result. □

Next, we prove a helper lemma that follows the same steps as the helper lemmas needed to prove Theorems 7, 8, and 9. Similar lemmas can be proved in the suboptimality gap or all ϵ-optimal π have low-rank Q^π setting.

Lemma 29. *Let ϵ, p_{H-t}, and λ be defined as in Theorem 21. Then, the learned policy and action-value function estimate satisfy*

$$\|\bar{Q}_{H-t} - Q^*_{H-t}\|_\infty \leq \frac{\epsilon(t+1)}{H}, \qquad \|\bar{Q}_{H-t} - Q^{\hat{\pi}}_{H-t}\|_\infty \leq \frac{\epsilon(t+1)}{H}$$

with probability at least $1 - O((t+1)n^{-3})$ for all $t \in \{0, \ldots, H_1\}$.

Proof of Lemma 29. We prove this with induction on t. At step $t = 0$, it follows that from Lemma 28 with probability $1 - O(n^{-3})$,

$$\|\bar{Q}_H - Q^*_H\|_\infty \leq \frac{\epsilon}{H}.$$

Since $Q^*_H = Q^{\hat{\pi}}_H$, the base case holds.

Let $x \in [H-1]$. Assume that the inductive hypothesis,

$$\|\bar{Q}_{H-x} - Q^*_{H-x}\|_\infty \leq \frac{\epsilon(x+1)}{H}, \qquad \|\|\bar{Q}_{H-x} - Q^{\hat{\pi}}_{H-x}\|_\infty \leq \frac{\epsilon(x+1)}{H}$$

with probability at least $1 - O\left((x+1)n^{-3}\right)$, holds. Following the steps of LR-EVI with the convex program based matrix estimation method, it follows that with probability $1 - O(n^{-3})$,

$$\|\bar{Q}_{H-s-1} - Q'_{H-s-1}\|_\infty \leq \frac{\epsilon}{H}$$

where $Q'_{H-x-1} = r_{H-x-1} + P_{H-x-1}\hat{V}_{H-x}$. The union bound asserts that the above error guarantee holds with probability at least $1 - O((x+2)n^{-3})$. Hence, for all $(s,a) \in S \times A$,

$$\begin{aligned}
|\bar{Q}_{H-x-1}(s,a) - Q^*_{H-x-1}(s,a)| &\leq |\bar{Q}_{H-x-1}(s,a) - Q'_{H-x-1}(s,a)| + |Q'_{H-x-1}(s,a) - Q^*_{H-x-1}(s,a)| \\
&\leq \frac{\epsilon}{H} + |E_{s' \sim P_{H-x-1}(\cdot|s,a)}[\hat{V}_{H-x}(s') - V^*_{H-x}(s')]| \\
&\leq \frac{\epsilon}{H} + |E_{s' \sim P_{H-h}(\cdot|s,a)}[(x+1)\epsilon/H]| \\
&= \frac{(x+2)\epsilon}{H}.
\end{aligned}$$

Following the same steps,

$$\begin{aligned}
|\bar{Q}_{H-x-1}(s,a) - Q^{\hat{\pi}}_{H-x-1}(s,a)| &\leq |\bar{Q}_{H-x-1}(s,a) - Q'_{H-x-1}(s,a)| + |Q'_{H-x-1}(s,a) - Q^{\hat{\pi}}_{H-x-1}(s,a)| \\
&\leq \frac{\epsilon}{H} + |E_{s' \sim P_{H-x-1}(\cdot|s,a)}[\hat{V}_{H-x}(s') - V^{\hat{\pi}}_{H-x}(s')]| \\
&\leq \frac{\epsilon}{H} + |E_{s' \sim P_{H-h}(\cdot|s,a)}[(x+1)\epsilon/H]| \\
&= \frac{(x+2)\epsilon}{H}.
\end{aligned}$$

Hence, from mathematical induction, the lemma holds. □

We now present the proof of the main result of this section. Similarly, the same steps can be used to prove similar results in our other low-rank settings.

Proof of Theorem 21. We prove the correctness of the algorithm by applying Lemma 29 at time step 1, which occurs with probability at least $1 - O(Hn^{-3})$. Next, the number of samples used is

$\sum_{t=0}^{H-1} |\Omega_{H-t}|$. By definition of our sampling procedure, $k = |\Omega_h| \sim \text{Bin}(n^2, p_h)$. Hence, from the one-sided Bernstein's inequality, Proposition 31, for $h \in [H]$ and $C'' = \frac{\sqrt{8}}{C_{cvx}\sqrt{3}}$, it follows that

$$\begin{aligned}
\mathbb{P}(|\Omega_h| - \mathbb{E}[|\Omega_h|] \geq C'' p_h n^2) &\leq \exp\left(-\frac{p_h^2 (C'')^2 n^2}{2(p_h + \frac{p_h C''}{3}}\right) \\
&\leq \exp\left(-\frac{3 p_h C'' n^2}{8}\right) \\
&\leq \exp\left(-\mu^3 d^2 \kappa^2 H^4 n \log(n)/\epsilon^2\right).
\end{aligned}$$

Since $\mathbb{E}[|\Omega_h|] = n^2 p_h = C_{cvx} \mu^3 d^2 \kappa^2 H^4 n \log(n)/\epsilon^2$, from the union bound, it follows that $|\Omega_h| \in O(H^4 n \log(n)/\epsilon^2)$ for all $h \in [H]$ with probability at least $1 - \exp\left(-\mu^3 d^2 \kappa^2 H^4 n \log(n)/\epsilon^2\right)$. Hence, the sample complexity is upper bounded by

$$\sum_{t=0}^{H-1} |\Omega_{H-t}| \in \tilde{O}\left(\frac{\mu^3 H^5 n}{\epsilon^2}\right)$$

with probability at least $1 - O(Hn^{-3}) - \exp\left(-\mu^3 d^2 \kappa^2 H^4 n \log(n)/\epsilon^2\right)$.

□

N Additional Theorems for Reference

We present the following lemmas, propositions, and theorems for the readers' convenience.

Theorem 30 (Hoeffding's Inequality [47])**.** *Let $X_1, \ldots, X_n$ be independent, and X_i have mean μ_i and sub-Gaussian parameter σ_i. Then, for all $t \geq 0$, we have*

$$\mathbb{P}\left[\sum_{i=1}^{n} (X_i - \mu_i) \geq t\right] \leq \exp\left(-\frac{t^2}{2\sum_{i=1}^{n} \sigma_i^2}\right).$$

Proposition 31 (Proposition 2.14 (One-sided Bernstein's Inequality) [47])**.** *Given n independent random variables such that $X_i \leq b$ almost surely, we have*

$$\mathbb{P}\left(\sum_{i=1}^{n} (X_i - \mathbb{E}[X_i]) \geq cn\right) \leq \exp\left(-\frac{nc^2}{2(\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[X_i^2] + \frac{bc}{3})}\right).$$

Theorem 32 (Matrix Bernstein [43])**.** *Let $X^{(1)}, \ldots, X^{(n)} \in \mathbb{R}^{d_1 \times d_2}$ be independent zero-mean matrices satisfying*

$$\begin{aligned}
\|X^{(i)}\|_{op} \leq b, \quad a.s. \\
\max\{\|\sum_{i=1}^{n} \mathbb{E}[X^{(i)^\top} X^{(i)}]\|_{op}, \|\sum_{i=1}^{n} \mathbb{E}[X^{(i)} X^{(i)^\top}]\|_{op}\} \leq n\sigma^2.
\end{aligned}$$

Then

$$\mathbb{P}\left(\left\|\sum_{i=1}^{n} X^{(i)}\right\|_{op} \geq t\right) \leq (d_1 + d_2) \exp\left(-\frac{t^2}{2(n\sigma^2 + \frac{bt}{3})}\right).$$

Theorem 33 (Singular Value Courant-Fischer Minimax Theorem (Theorem 7.3.8 [21]))**.** *Let* $A \in \mathbb{R}^{m\times n}$*, and* $q = \min(m, n)$*, let* $\sigma_1(A), \sigma_2(A), \ldots, \sigma_q(A)$ *be the ordered singular values of* A*, and let* $k \in [q]$*. Then,*

$$\sigma_k(A) = \min_{S:dim(S)=m-k+1} \max_{x:0\neq X\in S} \frac{\|Ax\|_2}{\|x\|_2}$$

and

$$\sigma_k(A) = \max_{S:dim(S)=k} \min_{x:0\neq X\in S} \frac{\|Ax\|_2}{\|x\|_2}.$$