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

Tyler Sam  
Cornell University  
Ithaca, NY, USA

Yudong Chen  
University of Wisconsin-Madison  
Madison, WI, USA

Christina Lee Yu  
Cornell University  
Ithaca, NY, USA

CCS CONCEPTS
• Computing methodologies → Sequential decision making:

KEYWORDS
Reinforcement Learning, Low-Rank Matrix Estimation

ACM Reference Format:
Tyler Sam, Yudong Chen, and Christina Lee Yu. 2023. Overcoming the Long Horizon Barrier for Sample-Efficient Reinforcement Learning with Latent Low-Rank Structure. In Abstracts of the 2023 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '23 Abstracts), June 19–23, 2023, Orlando, FL, USA. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3578338.3593562

1 INTRODUCTION

Reinforcement learning (RL) methods have been increasingly popular in sequential decision making tasks due to its empirical success. However, large state and action spaces in real-world problems modeled as a Markov decision processes (MDPs) limit the use of RL algorithms. Given a standard finite-horizon MDP \( (S, A, P, R, H) \) with state space \( S \), action space \( A \), transition kernel \( P = \{P_h\}_{h \in [H]} \), reward function \( R = \{R_h\}_{h \in [H]} \), and time horizon \( H \), one needs \( \tilde{Q}(\{[S]\}A[|H|^2]/\epsilon^2) \) samples given a generative model to learn an optimal policy [3], which can be impractical when \( S \) and \( A \) are large. The above tabular RL framework does not capture the fact that many real-world systems in fact have additional structure that if exploited should improve computational and statistical efficiency. Moreover, [1] empirically verifies that optimal and near-optimal action-value functions (both viewed as \( |S| \)-by-\( |A| \) matrices) of classical stochastic control tasks have low rank. Thus, the critical question is what are the minimal low rank structural assumptions that allow for computationally and statistically efficient learning?

Under the assumption that \( Q^* \) is low rank, [2] develops an algorithm that combines a novel matrix estimation (ME) method with value iteration to find an \( \epsilon \)-optimal action-value function with \( \tilde{Q}(d(|S| + |A|)/\epsilon^2) \) samples for infinite-horizon \( \gamma \)-discounted MDPs assuming \( Q^* \) has rank \( d \) and the discount factor \( \gamma \) is bounded above by a small constant effectively limits their results to short constant horizons. Furthermore, their algorithm 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.

To illustrate the additional complexities that arise from MDPs with long horizons, evidenced by the restriction on \( \gamma \) in [2], we construct a class of two MDPs, indexed by \( \theta \), with low rank \( Q^* \), where one must incur an exponential number of samples in \( H \) to distinguish between two. In this setting, 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.

**Theorem 1.** There exists a class of MDPs with the above observation model, such that to learn a 1/8-optimal policy with probability at least 0.9, the learner must observe \( n = \Omega(4^H) \) samples from \( R^0_H(\theta) \).

This result shows that the constant horizon assumption in [2] is not merely an artifact of their analysis, motivating us to consider additional assumptions beyond \( Q^*_H \) being low rank in order to achieve stable and sample-efficient learning with long horizons.

2 ASSUMPTIONS

Our information theoretic lower bound shows that assuming only \( Q^* \) is low rank is too weak of an assumption. Below we present three additional assumptions that show that assuming \( rank(Q^*_{h}) = \) \( d \) for all \( h \in [H] \) combined with any single one of these three assumptions enables our main algorithm to achieve the desired sample complexity.

**Assumption 1 (Suboptimality Gap).** For each \( (s, a) \in S \times A \), the suboptimality gap is defined as \( \Delta_h(s, a) := V^\pi_h(s) - Q^*_h(s, a) \). 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 1 stipulates the existence of a suboptimality gap, which is always positive for any non-trivial MDP in which there is at least one suboptimal action with finite \( S, A, \) and \( H \).

**Assumption 2 (\( \epsilon \)-optimal \( \pi \) have low-rank \( Q^\pi \)).** For all \( \epsilon \)-optimal policies \( \pi \), the associated \( Q^\pi_h \) matrices are rank-\( d \) for all \( h \in [H] \), i.e., can be represented via \( Q^\pi_h = U^{(h)} \Sigma^{(h)} (V^{(h)})^\top \) for some \( |S| \times d \) matrix \( U^{(h)} \), \( |A| \times d \) matrix \( V^{(h)} \), and \( d \times d \) diagonal matrix \( \Sigma^{(h)} \).

Assumption 2 imposes that all \( \epsilon \)-optimal policies \( \pi \) have low-rank \( Q^\pi_H \). We have not seen this assumption in existing literature.
Sample Complexity

We present algorithms that admit sample-efficient RL under the transition kernel $P_h$ has Tucker rank $(|S|, |A|, d)$, with shared latent factors, i.e., 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 $P_h(s'|s, a) = \sum_{i=1}^d U(h)(s', s, i)V(h)(a, i)$ and $r_h(s, a) = \sum_{i=1}^d W(h)(s, i)V(h)(a, i)$.

Assumption 3 is our strongest low-rank structural assumption as it implies that for any value function estimate $\hat{V}_h$, the matrix $r_h + [P_h \hat{V}_h]'$ is low rank, which is key in the analysis of our algorithm.

To characterize the error guarantees of the matrix estimation algorithm we use, we present the following definition.

Definition 2 $(k, \alpha)$-Anchor States and Actions. A set of states $S_h^k \subset S$ and a set of actions $A_h^k \subset A$ are $(k, \alpha)$-anchor states and actions for a rank-$d$ matrix $Q_h$ if $|S_h^k| |A_h^k| \leq k$, the submatrix $Q_h(S_h^k, A_h^k)$ has rank $d$, and $\|Q_h\|_2/\sigma_2(Q_h(S_h^k, A_h^k)) \leq \alpha$.

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 $d$. The parameter $\alpha$ depends on the quality of the anchor sets; sub-matrices that are close to being singular result in large $\alpha$. While Shah et al. [2] posit that it suffices empirically to choose states and actions that are far from each other as anchor states and actions, 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.

Definition 3 (Incoherence). Let $Q_h \in \mathbb{R}^{|S| \times |A|}$ be a rank-$d$ matrix with singular value decomposition $Q_h = USV^T$ with $U \in \mathbb{R}^{|S| \times d}$ and $V \in \mathbb{R}^{|A| \times d}$. $Q_h$ is $\mu$-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 $U$.

Both $\mu$ and $\kappa$, the condition number of $Q_h$, are used in the analysis to show that the entrywise error amplification from the matrix estimation method scales with $\mu, d, \kappa$ instead of the size of the state or action space, $k$, or $\alpha$.

3 ALGORITHMS

We present algorithms that admit sample-efficient RL under the assumptions in the previous section. Our algorithms synthesize ME with either empirical value iteration or Monte Carlo policy iteration. We use the ME method proposed in [2] as it admits entrywise error bounds provided that the algorithm has knowledge of special states and actions (Step 2). LR-EVI refers to the the algorithm which uses option (a) for Step 1 below, and LR-MCPI refers to the algorithm which uses option (b) for Step 1.

Hyperparameters: $\Omega_h \in [H], \{N_{e,a,h}\}_{(s,a,h)} \in S \times A \times H$, and $\mathbb{E}()$

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

For each $h \in \{H, H - 1, H - 2, \ldots, 1\}$ 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 $N_{e,a,h}$ samples of a single transition starting from state $s$ and action $a$ at step $h$ to estimate $\hat{Q}_h(s, a)$:
  
  $\hat{Q}_h(s, a) = \hat{r}_h(s, a) + \mathbb{E}_{s' \sim P_h(\cdot|s, a)}[\hat{V}_{h+1}(s')]$,
  
  where $\hat{r}_h(s, a)$ denotes the empirical average reward, and $\hat{P}_h(\cdot|s, a)$ denotes the empirical distribution over the sampled states.

  (b) **Monte Carlo Policy Evaluation:** Collect $N_{e,a,h}$ independent full trajectories starting from state $s$ and action $a$ at step $h$, where actions are chosen according to the learned policy ($\hat{\pi}_{h+1}, \ldots, \hat{\pi}_{H}$), to estimate $\hat{Q}_h(s, a)$:
  
  $\hat{Q}_h(s, a) = \frac{\text{sum}}{N_{e,a,h}}(s, a) = \frac{1}{N_{e,a,h}} \sum_{i=1}^{N_{e,a,h}} \sum_{t=h}^{H} \hat{r}_i(s, a)$,
  
  where $\frac{\text{sum}}{N_{e,a,h}}(s, a)$ denotes the empirical average reward of the $N_{e,a,h}$ trajectories.

  - **Step 2:** Predict the action-value function for all $(s, a) \in S \times A$ according to $\mathbb{E}()$, $\hat{Q}_h = \mathbb{E}\{\hat{Q}_h(s, a)\}_{(s,a) \in \Omega_h}$.

  - **Step 3:** Compute the estimates of the value function and the optimal policy according to $\hat{V}_h(s) = \max_{a \in A} \hat{Q}_h(s, a)$ and $\hat{\pi}_h(s) = \delta_{\arg\max \hat{Q}_h(s, a)}$.

4 THEORETICAL RESULTS

Table displays the sample complexities of learning an $\epsilon$-optimal action-value function under different low-rank assumptions, beginning from the weakest to the strongest setting.

| MDP Setting | Sample Complexity |
|-------------|--------------------|
| Low-rank $Q^*_h$ & $\Delta_{\min} > 0$ (LR-MPCI) | $O\left(\frac{d^3\mu^3\kappa^2(|S|+|A|)^2H^4}{N_{e,a,h}}\right)$ |
| $\epsilon$-optimal policies have low-rank $Q^*_h$ (LR-MPCI) | $O\left(\frac{d^3\mu^3\kappa^2(|S|+|A|)^2H^4}{e^2}\right)$ |
| Low Tucker rank transition kernels (LR-EVI) | $O\left(\frac{d^3\mu^3\kappa^2(|S|+|A|)^2H^4}{\epsilon^2}\right)$ |

Table 1: Sample complexity bounds under low-rank settings. The $\mathcal{O}()$ notation hides terms independent of $|S|$ or $|A|$.

In all three low-rank settings, Table 1 show that LR-EVI or LR-MCPI learn near-optimal policies and action-value functions in a sample efficient manner. Specifically, in both algorithms, under the ME regularity assumption, incorporating a ME subroutine decreases the sample complexity’s dependence on $S$ and $A$ from $|S||A|$ to $|S| + |A|$, which is minimax optimal with respect to $|S|$ and $|A|$.

REFERENCES

[1] ROZADA, S., AND MARQUES, A. G. Tensor and matrix low-rank value-function approximation in reinforcement learning. arXiv preprint arXiv:2201.09736 (2022).

[2] SHAH, D., SONG, D., XU, Z., AND YANG, Y. Sample efficient reinforcement learning via low-rank matrix estimation. In Advances in Neural Information Processing Systems (2020), H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., pp. 12092–12103.

[3] SUNZARO, A., WANG, M., WU, X., YANG, L., AND YU, Y. Near-optimal time and sample complexities for solving markov decision processes with a generative model. In Advances in Neural Information Processing Systems (2018), S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31, Curran Associates, Inc.