Model-Free Reinforcement Learning: from Clipped Pseudo-Regret to Sample Complexity

Zihan Zhang [1] Yuan Zhou [2] Xiangyang Ji [1]

Abstract

In this paper we consider the problem of learning an ϵ-optimal policy for a discounted Markov Decision Process (MDP). Given an MDP with S states, A actions, the discount factor $\gamma \in (0, 1)$, and an approximation threshold $\epsilon > 0$, we provide a model-free algorithm to learn an ϵ-optimal policy with sample complexity $\tilde{O}(\frac{SA\ln(1/p)}{\epsilon^2(1-\gamma)^{5.5}})$ [1] and success probability $(1-p)$. For small enough ϵ, we show an improved algorithm with sample complexity $\tilde{O}(\frac{SA\ln(1/p)}{\epsilon^2(1-\gamma)^3})$. While the first bound improves upon all known model-free algorithms and model-based ones with tight dependence on S, our second algorithm beats all known sample complexity bounds and matches the information theoretic lower bound up to logarithmic factors.

1. Introduction

Reinforcement learning (RL) (Burnetas & Katehakis, 1997) studies the problem of how to make sequential decisions to learn and act in unknown environments (which is usually modeled by a Markov Decision Process (MDP)) and maximize the collected rewards. There are mainly two types of algorithms to approach the RL problems: model-based algorithms and model-free algorithms. Model-based RL algorithms keep explicit description of the learned model and make decisions based on this model. In contrast, model-free algorithms only maintain a group of value functions instead of the complete model of the system dynamics. Due to their space- and time-efficiency, model-free RL algorithms have been getting popular in a wide range of practical tasks (e.g., DQN (Mnih et al., 2015), TRPO (Schulman et al., 2015), and A3C (Mnih et al., 2016)).

In RL theory, model-free algorithms are explicitly defined to be the ones whose space complexity is always sublinear relative to the space required to store the MDP parameters (Jin et al., 2018). For tabular MDPs (i.e., MDPs with finite number of states and actions, usually denoted by S and A respectively), this requires that the space complexity to be $o(S^2A)$. Motivated by the empirical effectiveness of model-free algorithms, the intriguing question of whether model-free algorithms can be rigorously proved to perform as well as the model-based ones has attracted much attention and been studied in the settings such as regret minimization for episodic MDPs (Azar et al., 2017; Jin et al., 2018; Zhang et al., 2020)).

In this work, we study the PROBABLY-APPROXIMATELY-CORRECT-RL (PAC-RL) problem, i.e., to designing an algorithm for learning an approximately optimal policy. We will focus on designing the model-free algorithms, and under the model of discounted tabular MDPs with a discount factor γ. The RL algorithm runs for infinitely many time steps. At each time step t, the RL agent learns a policy π_t based on the information collected before time t, observes the current state s_t, makes an action $a_t = \pi_t(s_t)$, receives the reward r_t and transits to the next state s_{t+1} according to the underlying environments. The goal of the agent is to learn the policy π_t at each time t so as to maximize the γ-*discounted accumulative reward* $V^{\pi_t}(s_t)$. More concretely, we wish to minimize the *sample complexity* for the agent to learn an ϵ-optimal policy, which is defined to be the number of time steps that $V^{\pi_t}(s_t) < V^*(s_t) - \epsilon$, where V^* is the optimal discounted accumulative reward that starts with s_t, and the formal definitions of both V^{π} and V^* can be found in Section 2.

The PAC-RL addresses the important problem about how many trials are required to learn a good policy. We also note that in the PAC-RL definition, the exploration at each time step has to align with the learned policy (i.e., $a_t = \pi_t(s_t)$). This is stronger than the usual PAC learning definition in other online learning settings such as multi-armed bandits (see, e.g., (Even-Dar et al., 2006)) and PAC-RL with a simulator (see Section 1.2), where the exploration actions can be arbitrary and may incur a large regret compared to the optimum.

[1]Tsinghua University [2]University of Illinois Urbana Chanmpion. Correspondence to: Zihan Zhang <zihan-zh17@mails.tsinghua.edu.cn>, Yuan Zhou <yuanz@illinois.edu>, Xiangyang Ji <xyji@tsinghua.edu.cn>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

[1] In this work, the notation $\tilde{O}(\cdot)$ hides poly-logarithmic factors of $S, A, 1/(1-\gamma)$, and $1/\epsilon$.

Quite a few algorithms have been proposed over the past nearly two decades for the PAC-RL problem. For model-based algorithms, MoRmax (Szita & Szepesvari, 2010) achieves the $\tilde{O}(\frac{SA\ln(1/p)}{\epsilon^2(1-\gamma)^6})$ sample complexity, and UCRL-γ (Lattimore & Hutter, 2012) achieves $\tilde{O}(\frac{S^2A\ln(1/p)}{\epsilon^2(1-\gamma)^3})$. It is also worthwhile to mention that R-max (Brafman & Tennenholtz, 2003) was designed for learning the more general stochastic games and achieves the $\tilde{O}(\frac{S^2A\ln(1/p)}{\epsilon^3(1-\gamma)^6})$ sample complexity in our setting (as analyzed in (Kakade, 2003)). Unfortunately, none of these algorithms matches the information theoretical lower bound $\Omega(\frac{SA}{\epsilon^2(1-\gamma)^3})$ proved by (Lattimore & Hutter, 2012). On the model-free side, known bounds are even less optimal – the delayed Q-learning algorithm proposed by (Strehl et al., 2006) achieves the sample complexity of $\tilde{O}(\frac{SA\ln(1/p)}{\epsilon^4(1-\gamma)^8})$, and recent work (Dong et al., 2019) made an improvement to $\tilde{O}(\frac{SA\ln(1/p)}{\epsilon^2(1-\gamma)^7})$ via a more carefully designed Q-learning variant. Besides the results above, (Pazis et al., 2016) provided $\tilde{O}\left(\frac{S^2A}{\epsilon^2(1-\gamma)^4}\right)$ sample complexity. However, their algorithm consumes $\tilde{O}(\frac{SA}{\epsilon^2(1-\gamma)^4})$ space cost and $\tilde{O}\left(\frac{SA^2}{\epsilon^2(1-\gamma)^4}\right)$ computational cost each step, which is far beyond the cost of both model-based and model-free algorithms when ϵ is small.

1.1. Our Results

We design a model-free algorithm that achieves asymptotically optimal sample complexity, as follows.

Theorem 1. *By the model-free algorithm* UCB-MULTISTAGE-ADVANTAGE, *for any discounted MDP with S states, A actions, and the discount factor γ, any approximation threshold $\epsilon \in (0, \frac{(1-\gamma)^{14}}{S^2A^2})$ and failure probability parameter p, with probability $(1-p)$, the sample complexity to learn an ϵ-optimal policy with* UCB-MULTISTAGE-ADVANTAGE *is bounded by* $\tilde{O}(\frac{SA\ln(1/p)}{\epsilon^2(1-\gamma)^3})$.

In the theorem statement, $\mathrm{poly}(S, A, 1/(1-\gamma))$ stands for a universal polynomial that is independent of the MDP. Our UCB-MULTISTAGE-ADVANTAGE algorithm is model-free, which uses only $O(SA)$ space , and its time complexity per time step is $O(1)$. In contrast, the model-based algorithms have to consume $\Omega(S^2A)$ space. For asymptotically small ϵ, the sample complexity of UCB-MULTISTAGE-ADVANTAGE matches the information theoretic lower bound of $\Omega(\frac{SA}{\epsilon^2(1-\gamma)^3})$ up to poly-logarithmic terms, and improves upon all known algorithms in literature, even including the model-based ones. In Appendix A, we present a tabular view of the comparison between our algorithms and the previous works.

To prove Theorem 1, we make two main technical contributions. The first one is a novel relation between sample complexity and the so-called *clipped pseudo-regret*, which can also be viewed as the clipped Bellman error of the learned value function and policy at each time step. This relation enables us to reduce the sample complexity analysis to bounding the clipped pseudo-regret. Our second technique is a *multi-stage update rule*, where the visits to each state-action pair are partitioned according to two types of stages. An update to the Q-function is triggered only when a stage of either type has concluded. The lengths of the two types of stages are set by different choices of parameters so that we can reduce the clipped pseudo-regret while still maintaining a decent rate to learn the value function. Finally, we also spend much technical effort to incorporate the variance reduction technique for RL via *reference-advantage decomposition* introduced in the recent work (Zhang et al., 2020).

A more detailed overview of our techniques is available in Section 4. Since the proof of Theorem 1 is rather involved, we will first provide a proof of the following weaker statement, and defer the full proof of Theorem 1 to Appendix D.

Theorem 2. *By the model-free algorithm* UCB-MULTISTAGE, *for any approximation threshold $\epsilon \in (0, \frac{1}{1-\gamma}]$ and any failure probability parameter p, with probability $(1-p)$, the sample complexity to learn an ϵ-policy with* UCB-MULTISTAGE *is bounded by* $\tilde{O}(\frac{SA\ln(1/p)}{\epsilon^2(1-\gamma)^{5.5}})$.

We highlight that the sample complexity bound in Theorem 2 holds for every possible $\epsilon \in (0, \frac{1}{1-\gamma}]$. Although the dependency on γ becomes $(1-\gamma)^{-5.5}$, UCB-MULTISTAGE still beats all known model-free and model-based algorithms with tight dependence on S. The proof of Theorem 2 does not rely on the variance reduction technique based on reference-advantage decomposition (Zhang et al., 2020), but is sufficient to illustrate both of our main technical contributions.

1.2. Additional Related Works

The PAC-RL problem has also been extensively studied under the setting of finite-horizon episodic MDPs (Dann & Brunskill, 2015; Dann et al., 2017; 2019), where the sample complexity is defined as the number of episodes in which the policy is not ϵ-optimal. Assuming H is the length of an episode, the optimal sample complexity bound is $\tilde{O}(\frac{SAH^2\ln(1/p)}{\epsilon^2})$, proved by (Dann et al., 2019). Note that the sample complexity bounds for finite-horizon episodic MDP do not imply sample complexity bounds for infinite-horizon discounted MDP because one ϵ-optimal episode may contain non-ϵ-optimal steps. Also we note that existing algorithms for the finite-horizon case are model-based. It is still an open problem whether model-free algorithm can achieve near-optimal sample complexity bound for the finite-horizon case.

Much effort has also been made to study the PAC learning problem for discounted infinite-horizon MDPs, with the access to a generative model (a.k.a., a simulator). In this problem, the agent can query the simulator to draw a sample $s' \sim P(\cdot|s, a)$ for any state-action pair (s, a), and the goal is to output an ϵ-optimal policy (with probability $(1-p)$) at the end of the algorithm. This problem has been studied in (Even-Dar & Mansour, 2003; Azar et al., 2011; Gheshlaghi et al., 2012; Sidford et al., 2018b;a), and (Sidford et al., 2018a) achieves the almost tight sample complexity $\tilde{O}(\frac{SA\ln(1/p)}{\epsilon^2(1-\gamma)^3})$.

2. Preliminaries

A discounted Markov Decision Process is given by the five-tuple $M = \langle \mathcal{S}, \mathcal{A}, P, r, \gamma \rangle$, where $\mathcal{S} \times \mathcal{A}$ is the state-action space, P is the transition probability matrix, r is the deterministic reward function[2] and $\gamma \in (0, 1)$ is the discount factor.

The RL agent interacts with the environment for infinite number of times. At the t-th time step, the agent learns a policy π_t based on the samples collected before time t, observes s_t, executes $a_t = \pi_t(s_t)$, receives the reward $r(s_t, a_t)$, and then transits to s_{t+1} according to $P(\cdot|s_t, a_t)$.

Given a deterministic[3] stationary policy $\pi : \mathcal{S} \to \mathcal{A}$, the value function and Q function are defined as

$$V^\pi(s) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r(s_t, \pi(s_t)) \Big| s_1 = s, a_t = \pi(s_t)\right]$$
$$Q^\pi(s, a) = r(s, a) + \gamma P(\cdot|s, a)^\top V^\pi = r(s, a) + \gamma P_{s,a} V^\pi,$$

where we use xy to denote $x^\top y$ for x and y of the same dimension and use $P_{s,a}$ to denote $P(\cdot|s, a)$ for simplicity.

The optimal value function is given by $V^*(s) = \sup_\pi V^\pi(s)$ and the optimal Q-function is defined to be $Q^*(s, a) = r(s, a) + \gamma P_{s,a} V^*$ for any $(s, a) \in \mathcal{S} \times \mathcal{A}$.

We present below the formal definitions for sample complexity and PAC-RL .

Definition 1 (ϵ-sample complexity)**.** *Given an algorithm $\mathcal{G}$ and $\epsilon \in (0, \frac{1}{1-\gamma}]$, the ϵ-sample complexity for $\mathcal{G}$ is* $\sum_{t \geqslant 1} \mathbb{I}\left[V^*(s_t) - V^{\pi_t}(s_t) > \epsilon\right]$.

Definition 2 ((ϵ, p)-PAC-RL)**.** *An algorithm $\mathcal{G}$ is said to be (ϵ, p)-**PAC-RL** (Probably Approximately Correct in RL) if for any $\epsilon \in (0, \frac{1}{1-\gamma}], p > 0$, with probability $1-p$, the sample complexity of $\mathcal{G}$ is bounded by some polynomial in $(S, A, \frac{1}{\epsilon}, \frac{1}{1-\gamma}, \ln(\frac{1}{p}))$.*

When ϵ and p are clear in the context, we simply write (ϵ, p)-PAC-RL and ϵ-sample complexity as PAC-RL and sample complexity respectively. The goal is to propose an PAC-RL algorithm to minimize the sample complexity.

3. The UCB-MultiStage Algorithm

In this section, we introduce the UCB-MultiStage algorithm. The algorithm takes $\mathcal{S}, \mathcal{A}, \gamma, \epsilon$, sets $H = \max\{\frac{\ln(8/((1-\gamma)\epsilon))}{\ln(1/\gamma)}, \frac{1}{1-\gamma}\}$ and $B = \sqrt{H}$. Throughout the paper, we set $\iota = \ln(2/p)$. The algorithm is described in Algorithm 1. For each state-action pair (s, a), the samples are partitioned into consecutive stages. When a stage is filled, we update $Q(s, a)$ and $V(s)$ according to the samples in the stage via the usual value iteration method. The most interesting aspect about our method is that two types of stages, namely the *type-I and type-II stages*, are introduced. More concretely, the length of the j-th type-I stage is roughly $\check{e}_j \approx H(1+1/H)^{j/B}$ and the length of the j-th type-II stage is roughly $\bar{e}_j \approx H(1+1/H)^j$.

We note that the recent work (Zhang et al., 2020) designed a (single-)stage-based model-free RL algorithm for regret minimization. Our type-II stage is similar to their work, and its goal is to make sure that the value function is learned at a decent rate. In contrast, our type-I stage is new: it is shorter than the type-II stage, so that triggers more frequent updates and helps to reduce the difference between the value functions learned in neighboring type-I stages. The hyper-parameter B is used to adjust the frequency of type-I updates (i.e., updates triggered by type-I stage). The two types of stages work together to reduce the clipped pseudo-regret, and therefore achieve low sample complexity.

The precise definition of the stages. Let $d_1 = H$, $d_{j+1} = \lfloor (1 + \frac{1}{H}) d_j \rfloor$ for all $j \geqslant 1$. The sizes of the j-th type-I and type-II stage are given by $\check{e}_j = d_{\lceil j/B \rceil}$ and $\bar{e}_j = d_j$ respectively.

Let $N_0 = c_1 \cdot \frac{S^3AH^5\ln(4H^2S/\epsilon)\iota}{\epsilon^2}$ for some large enough constant c_1. We stop updating $Q(s, a)$ if the number of visits to (s, a) is greater than N_0, since the value functions will be sufficiently learned by that time.

Therefore, the time steps when an update is triggered by the type-I and type-II stages are respectively given by $\check{\mathcal{L}} = \{\sum_{i=1}^{j} \check{e}_i | 1 \leqslant j \leqslant \check{J}\}$ and $\bar{\mathcal{L}} = \{\sum_{i=1}^{j} \bar{e}_i | 1 \leqslant j \leqslant \bar{J}\}$, where $\check{J} = \max\{j | \sum_{i=1}^{j-1} \check{e}_i \leqslant N_0\}$ and $\bar{J} = \max\{j | \sum_{i=1}^{j-1} \bar{e}_i \leqslant N_0\}$. Without loss of generality, we assume that $\sum_{i=1}^{\check{J}} \check{e}_i = N_0$.

[2] It is easy to generalize our results to stochastic reward functions.

[3] In this work, we mainly consider deterministic policies since the optimal value function can be achieved by a deterministic policy.

The statistics. We maintain the following statistics during the algorithm: for each (s, a), we use $N(s, a)$, $\check{N}(s, a)$, and $\bar{N}(s, a)$ to respectively denote the total visit number, the visit number in the current type-I stage and the visit number in the current type-II stage of (s, a). We also maintain $\check{\mu}(s, a)$ and $\bar{\mu}(s, a)$, which are respectively the accumulators for state values $V(s')$ (where s' is the next state observed after (s, a)) during the current type-I and type-II stages.

We also remark that throughout the paper we will use ‘ ˇ’ to denote the quantities related to the type-I stage, and use ‘¯’ to denote the quantities related to the type-II stage.

Algorithm 1 UCB-MULTISTAGE

```
Initialize: $\forall(s, a) \in \mathcal{S} \times \mathcal{A}$: $Q(s, a) \leftarrow \frac{1}{1-\gamma}$,
$N(s, a), \check{N}(s, a), \bar{N}(s, a), \check{\mu}(s, a), \bar{\mu}(s, a) \leftarrow 0$;
for $t = 1, 2, 3, \ldots$ do
    Observe $s_t$;
    Take action $a_t = \arg\max_a Q(s_t, a)$ and observe $s_{t+1}$;
    \\ Maintain the statistics
    $(s, a, s') \leftarrow (s_t, a_t, s_{t+1})$;
    $n := N(s, a) \leftarrow N(s, a) + 1$;
    $\check{n} := \check{N}(s, a) \leftarrow \check{N}(s, a) + 1$;
    $\check{\mu} := \check{\mu}(s, a) \leftarrow \check{\mu}(s, a) + V(s')$;
    $\bar{n} := \bar{N}(s, a) \leftarrow \bar{N}(s, a) + 1$;
    $\bar{\mu} := \bar{\mu}(s, a) \leftarrow \bar{\mu}(s, a) + V(s')$;
    \\ Update triggered by a type-I stage
    if $n \in \check{\mathcal{L}}$ then
        $\check{b} \leftarrow \min\{2\sqrt{H^2\iota/\check{n}}, 1/(1-\gamma)\}$;    (1)
        $Q(s, a) \leftarrow \min\{r(s, a) + \gamma(\check{\mu}/\check{n}) + \check{b}, Q(s, a)\}$;    (2)
        $\check{N}(s, a) \leftarrow 0$;
        $\check{\mu}(s, a) \leftarrow 0$;
        $V(s) \leftarrow \max_a Q(s, a)$;
    end if
    \\ Update triggered by a type-II stage
    if $n \in \bar{\mathcal{L}}$ then
        $\bar{b} \leftarrow \min\{2\sqrt{H^2\iota/\bar{n}}, 1/(1-\gamma)\}$;
        $Q(s, a) \leftarrow \min\{r(s, a) + \gamma(\bar{\mu}/\bar{n}) + \bar{b}, Q(s, a)\}$;    (3)
        $\bar{N}(s, a) \leftarrow 0$;
        $\bar{\mu}(s, a) \leftarrow 0$;
        $V(s) \leftarrow \max_a Q(s, a)$;
    end if
end for
```

4. Technical Overview

Both of the algorithms introduced in this paper are variants of Q-learning, where the optimistic value function V and the Q-function are maintained. For each time t, we use V_t and Q_t to denote the corresponding functions at the beginning of the time step. The learned policy π_t will always be the greedy policy based on Q_t, i.e., $\pi_t(s) = \arg\max_a Q_t(s, a)$ for all $s \in \mathcal{S}$. Below we explain the main techniques used in UCB-MULTISTAGE as well as UCB-MULTISTAGE-ADVANTAGE.

Reducing Sample Complexity to Bounding the Clipped Pseudo-Regret. For any time t, define the *pseudo-regret* vector ϕ_t to be the vector such that for any $s \in \mathcal{S}$,

$$\phi_t(s) = V_t(s) - (r(s, \pi_t(s)) + \gamma P_{s,\pi_t(s)} V_t).$$

We now outline our first technical idea that the sample complexity can be bounded by the total clipped pseudo-regret, approximately in the form of (5) (up to a ϵ^{-1} factor and an additive error term).

Note that ϕ_t can also be viewed as the Bellman error vector of the value function V_t and the policy π_t. Let P_{π_t} be the transition matrix such that $\mathsf{P}_{\pi_t}(s) = P_{s,\pi_t(s)}$ for any $s \in \mathcal{S}$. By Bellman equation we have that

$$\begin{aligned}
&V_t - V^{\pi_t} \\
&= \gamma \mathsf{P}_{\pi_t}(V_t - V^{\pi_t}) + \phi_t \\
&= (\gamma \mathsf{P}_{\pi})^2(V_t - V^{\pi_t}) + \gamma \mathsf{P}_{\pi_t}\phi_t + \phi_t \\
&= \ldots \\
&= \sum_{i=0}^{\infty} (\gamma \mathsf{P}_{\pi_t})^i \phi_t.
\end{aligned}$$

Define $\mathrm{clip}(x, y) = x\mathbb{I}[x \geqslant y]$ for $x, y \in \mathbb{R}$ and

$$\mathrm{clip}(x, y) = [\mathrm{clip}(x_1, y), \ldots, \mathrm{clip}(x_n, y)]^\top$$

for $x = [x_1, \ldots, x_n]^\top \in \mathbb{R}^n$.

Therefore, if $V_t(s_t) - V^{\pi_t}(s_t) > \epsilon$, then for some constant $M > 1$,

$$\begin{aligned}
&\mathbf{1}_{s_t}^\top \sum_{i=0}^{\infty} (\gamma \mathsf{P}_{\pi_t})^i \mathrm{clip}(\phi_t, \frac{\epsilon(1-\gamma)}{M}) \\
&\geqslant \mathbf{1}_{s_t}^\top \sum_{i=0}^{\infty} (\gamma \mathsf{P}_{\pi_t})^i \left(\phi_t - \frac{\epsilon(1-\gamma)}{M}\right) \\
&= \mathbf{1}_{s_t}^\top \sum_{i=0}^{\infty} (\gamma \mathsf{P}_{\pi_t})^i \phi_t - \frac{1}{1-\gamma} \cdot \frac{\epsilon(1-\gamma)}{M} \\
&= V_t(s_t) - V^{\pi_t}(s_t) - \frac{\epsilon}{M} \\
&> \frac{(M-1)\epsilon}{M},
\end{aligned}$$

where $\mathbf{1}_{s_t}$ is the unit vector with the only non-zero entry at s_t and the first inequality is by the fact $\mathrm{clip}(x, y) \geqslant x - y$ for $x, y \geqslant 0$. For any $H = \Theta(\ln(((1-\gamma)\epsilon)^{-1})/(1-\gamma))$, it then follows that

$$\begin{aligned}&\mathbb{I}\left[V_t(s_t) - V^{\pi_t}(s_t) > \epsilon\right]\epsilon \\ &\leqslant O\left(\mathbf{1}_{s_t}^{\top} \sum_{i=0}^{H-1} (\gamma \mathsf{P}_{\pi_t})^i \mathrm{clip}(\phi_t, \epsilon(1-\gamma)/M)\right). \qquad (4)\end{aligned}$$

We now sum up (4) over all time steps t. If we can carefully design the algorithm so that π_t, V_t (and therefore ϕ_t) do not change frequently, we have $\pi_t = \pi_{t+i}$ and $\phi_t = \phi_{t+i}$ for small enough i and most t, and therefore we can upper bound $\sum_{t\geqslant 1} \mathbb{I}\left[V_t(s_t) - V^{\pi_t}(s_t) > \epsilon\right]\epsilon$ by the order of

$$\begin{aligned}&\sum_{t\geqslant 1} \mathbf{1}_{s_t}^{\top} \sum_{i=0}^{H-1} (\gamma \mathsf{P}_{\pi_{t+i}})^i \mathrm{clip}(\phi_{t+i}, \epsilon(1-\gamma)/M) \\ &\leqslant \sum_{t\geqslant 1} \mathbf{1}_{s_t}^{\top} \sum_{i=0}^{H-1} (\mathsf{P}_{\pi_{t+i}})^i \mathrm{clip}(\phi_{t+i}, \epsilon(1-\gamma)/M) \\ &\approx O(H) \cdot \sum_{t\geqslant 1} \mathrm{clip}(\phi_t(s_t), \epsilon(1-\gamma)/M), \qquad (5)\end{aligned}$$

where the approximation (5) also uses the assumption that $\pi_t = \pi_{t+i}$ and $\phi_t = \phi_{t+i}$ hold for most t and i. In Lemma 5, we formalize this intuition and show that if we set $M = 8H(1-\gamma)$, the sample complexity $\sum_{t\geqslant 1} \mathbb{I}\left[V_t(s_t) - V^{\pi_t}(s_t) > \epsilon\right]$ can be upper bounded by $O(H/\epsilon) \cdot \sum_{t\geqslant 1} \mathrm{clip}(\phi_t(s_t), \epsilon(1-\gamma)/M)$ (plus an additive error), and therefore we only need to upper bound the total clipped pseudo-regret.

The Multi-Stage Update Rule. As stated before, the design of type-I stage is our main technical contribution. To better explain the intuition and motivate the type-I stage, let us consider a fixed state-action pair (s, a). Suppose at time step $(t-1)$, (s, a) is visited and the visit number reaches the end of a type-I stage, then the following update is triggered:

$$Q_t(s, a) \leftarrow \min\{r(s, a) + \check{b} + \frac{\gamma}{\check{n}} \sum_{i=1}^{\check{n}} V_{\check{l}_i}(s_{\check{l}_i+1}), Q_{t-1}(s, a)\},$$

where $\check{n}$ is the number of samples in this stage, $\check{l}_i$ is time of the i-th sample in the stage, and $\check{b}$ denotes the exploration bonus. Thanks to the update rule, V_t and Q_t are non-increasing in t. By concentration inequalities and the proper design of $\check{b}$, we get

$$\begin{aligned}&Q_t(s, a) \\ &\leqslant r(s, a) + 2\check{b} + P_{s,a}\left(\frac{\gamma}{\check{n}} \sum_{i=1}^{\check{n}} V_{\check{l}_i}\right) \\ &\leqslant r(s, a) + 2\check{b} + \gamma P_{s,a} V_t + \gamma P_{s,a}\left(\frac{1}{\check{n}} \sum_{i=1}^{\check{n}} V_{\check{l}_i} - V_t\right) \qquad (6) \\ &\leqslant r(s, a) + 2\check{b} + \gamma P_{s,a} V_t + \gamma P_{s,a}(V_{\underline{t}} - V_{\bar{t}}), \qquad (7)\end{aligned}$$

where $\underline{t} = \min_i \check{l}_i$ is the start time of the stage and $\bar{t}$ is the start time of the next stage. Let $a = \pi_t(s)$. By the definition of $\phi_t(s)$ and optimism of V_t, when $Q_t(s, a) - Q^*(s, a) < \epsilon(1-\gamma)/M$, we have that

$$\begin{aligned}&\mathrm{clip}(\phi_t(s), \epsilon(1-\gamma)/M) \\ &\leqslant \mathrm{clip}(Q_t(s, a) - Q^*(s, a), \epsilon(1-\gamma)/M) = 0 \qquad (8)\end{aligned}$$

In the case $Q_t(s, a) - Q^*(s, a) \geqslant \epsilon(1-\gamma)/M$, with an averaging argument we have that

$$\begin{aligned}&\mathrm{clip}(\phi_t(s), \epsilon(1-\gamma)/M) \\ &\leqslant \mathrm{clip}(2\check{b} + \gamma P_{s,a}(V_{\underline{t}} - V_{\bar{t}}), \epsilon(1-\gamma)/M) \\ &\leqslant 2\mathrm{clip}(2\check{b}, \epsilon(1-\gamma)/(2M)) \\ &\qquad + O(\gamma) \cdot P_{s,a}\mathrm{clip}(V_{\underline{t}} - V_{\bar{t}}, \epsilon(1-\gamma)/(2M)). \qquad (9)\end{aligned}$$

On the benefit of type-II stages, $N_t(s, a) \geqslant N_0$ implies $Q_t(s, a) - Q^*(s, a) < \epsilon(1-\gamma)/M$. So it suffices to bound

$$\begin{aligned}&\mathbb{I}[N_t(s, a) < N_0] P_{s,a}\mathrm{clip}(V_{\underline{t}} - V_{\bar{t}}, \epsilon(1-\gamma)/(2M)) \\ &+ \mathbb{I}[N_t(s, a) < N_0]\mathrm{clip}(2\check{b}, \epsilon(1-\gamma)/M) \qquad (10)\end{aligned}$$

.

We now discuss how to deal with the two terms and how the parameter B affects the bounds.

Bounding the first term of (10). We first focus on the second term $(\mathbb{I}[N_t(s_t, a_t) < N_0] P_{s,a}\mathrm{clip}(V_{\underline{t}} - V_{\bar{t}}, \epsilon(1-\gamma)/(2M)))$ in (10). For each j, let $t_j = t_j(s, a)$ be the start time of the j-th stage of (s, a). The total contribution of the second term in (10) is bounded by the order of

$$\sum_{s,a} \sum_j \check{e}_j P_{s,a}\mathrm{clip}\left((V_{t_{j-1}(s,a)} - V_{t_{j+1}(s,a)}), \epsilon(1-\gamma)/(2M)\right). \qquad (11)$$

Thanks to the updates triggered by the type-II stages, V_t converges to V^* at a rate that is independent of B. Increasing B will shorten the length of the type-I stages, making $V_{t_{j-1}(s,a)}$ closer to $V_{t_{j+1}(s,a)}$, and reduce the magnitude of (11). In Lemma 8, we formalize this intuition and show that when $M = 8H(1-\gamma)$, (11) can be upper bounded by $\tilde{O}(SAH^5 \ln(1/p)/(\epsilon B))$. Therefore, choosing a large enough B will eliminate the H factors in the numerator.

Bounding the second term of (10). On the other hand, however, a larger B means smaller number of samples in the type-I stages, leads to a bigger estimation variance, and therefore forces us to choose a greater exploration bonus $\check{b}$. We have to choose $B = \Theta(\sqrt{H})$ to achieve the optimal balance between the two terms in (10).

To utilize the full power of our multi-stage update rule, we would like to set $B = \Theta(H^3)$. However, the second term in (10) becomes much bigger. In the next subsection, we discuss how to deal with this problem via the variance reduction method, which leads to the asymptotically near-optimal bound in Theorem 1.

Variance Reduction via Reference-Advantage Decomposition. This technique is only used in UCB-MultiStage-Advantage and the proof of Theorem 1, which is deferred to Appendix D due to space constraints. We explain the technique as follows.

As discussed above, when B is set large, we suffer bigger estimation variance, as fewer samples are allowed in the type-I stages. In model-free regret minimization tasks, similar problem arises where the algorithm (e.g., (Jin et al., 2018)) can only use the recent tiny fraction of the samples and incurs sub-optimal dependency on the episode length. Recent work (Zhang et al., 2020) resolves this problem via the *reference-advantage decomposition* technique.

The high-level idea is that, assuming we have a δ-accurate estimation of V^*, namely the *reference value function* V^{ref}, such that $\|V^{\text{ref}} - V^*\|_\infty \leqslant \delta$, we only need to use the samples to estimate the difference $V^{\text{ref}} - V^*$, which is called the *advantage*. Therefore, the estimation error (incurred in places such as (6)) will be much smaller when δ is small. Choosing $\delta = 1/\sqrt{B}$, and together with the Bernstein-type exploration bonus (see, e.g., (Azar et al., 2017; Jin et al., 2018)), we are able to bound the total contribution of the first term in (9) [4] by $\tilde{O}(SA/(\epsilon(1-\gamma)^2)$, which (together with the H factor in (5)) aligns with the $(1-\gamma)^{-3}$ factor in the bound of Theorem 1. The discussion till now is based on the access of the reference value function V^{ref}. In reality, however, we need to learn the reference value function on the fly. This will incur an additive warm-up cost that polynomially depends on $1/\delta$. However, since δ is independent of ϵ, the extra cost is only a lower-order term.

5. Analysis of Sample Complexity

In this section, we prove Theorem 2 for UCB-MultiStage. We start with a few notations: we use $N_t(s,a)$, $\check{N}_t(s,a)$,$\bar{N}_t(s,a)$, $Q_t(s,a)$, $V_t(s)$ to denote respectively the values of $N(s,a)$, $\check{N}(s,a)$, $\bar{N}(s,a)$, $Q(s,a)$, $V(s)$ before the t-th time step. Let $\check{n}_t(s,a)$, $\check{\mu}_t(s,a)$ and $\check{b}^t(s,a)$ be the values of $\check{n}(s,a)$, $\check{\mu}(s,a)$ and $\check{b}(s,a)$ (respectively) in the latest type-I update of $Q(s,a)$ before the t-th time step. In other words, $\check{n}_t(s,a)$ is the length of the type-I stage immediately before the current type-I stage with respect to (s,a); $\check{b}_t(s,a) = \min\{2\sqrt{H^2\iota/\check{n}_t(s,a)}, 1/(1-\gamma)\}$; and

$$\check{\mu}_t(s,a) = \sum_{i=1}^{\check{n}_t(s,a)} V_{\check{l}_{t,i}(s,a)}(s_{\check{l}_{t,i}(s.a)+1}), \tag{12}$$

where $\check{l}_{t,i}(s,a)$ is the time step of the i-th visit among the $\check{n}_t(s,a)$ visits mentioned above. When t belongs to the first type-I stage of (s,a), we define $\check{n}_t(s,a) = 0$, $\check{\mu}_t(s,a) = 0$, and $\check{b}_t(s,a) = 1/(1-\gamma)$.

Given (s,a) and a time step t such that $(s_t, a_t) = (s,a)$, we use $j_t(s,a)$ to denote the index of the type-I which (the beginning of) the t-th time step belongs to with respect to (s,a). For $1 \leqslant j \leqslant \check{J}$, we use $\rho(j,s,a)$ to denote the start time of the j-th type-I with respect to (s,a). Besides, we define $\rho(\check{J}+1,s,a)$ to be the time t such that $N_t(s,a) = N_0$. We also define $\underline{\rho}_t(s,a) := \rho(j_t(s,a)-1,s,a)$ if $j_t(s,a) \geqslant 2$ and 0 otherwise, and $\overline{\rho}_t(s,a) := \rho(j_t(s,a)+1,s,a)$.

5.1. The Good Event

Let (s,a) and j be fixed. With a slight abuse of notation, we define $\check{l}_i$ to be the time when the i-th visit in the j-th type-I stage of (s,a) occurs. Define $\check{b}^{(j)} = \min\{2\sqrt{\frac{H^2\iota}{\check{e}_j}}, \frac{1}{1-\gamma}\}$ for $j \geqslant 2$. Define $\check{E}^{(j)}(s,a)$ be the event where the inequalities below hold

$$\frac{1}{\check{e}_j}\sum_{i=1}^{\check{e}_j} V^*(s_{\check{l}_i+1}) + \check{b}^{(j)} \geqslant P_{s,a}V^*;$$
$$\left|\frac{1}{\check{e}_j}\sum_{i=1}^{\check{e}_j}\left(V_{\check{l}_i}(s_{\check{l}_i+1}) - P_{s,a}V_{\check{l}_i}\right)\right| \leqslant \check{b}^{(j)}.$$

Similarly, let $\bar{l}_i$ be the time when the i-th visit in the j-th type-II stage of (s,a) occurs and $\bar{b}^{(j)} = \min\{2\sqrt{\frac{H^2\iota}{\bar{e}_j}}, \frac{1}{1-\gamma}\}$ for $j \geqslant 1$. Define $\bar{E}^j(s,a)$ be the event where

$$\frac{1}{\bar{e}_j}\sum_{i=1}^{\bar{e}_j} V^*(s_{\bar{l}_i+1}) + \bar{b}^{(j)} \geqslant P_{s,a}V^*;$$
$$\left|\frac{1}{\bar{e}_j}\sum_{i=1}^{\bar{e}_j}\left(V_{\bar{l}_i}(s_{\bar{l}_i+1}) - P_{s,a}V_{\bar{l}_i}\right)\right| \leqslant \bar{b}^{(j)}.$$

[4] More precisely, we refer to the total contribution related to the exploration bonus, which is actually in a different form from the first term in (9). This is because $\check{b}$ has to be re-designed using the Bernstein-type exploration bonus technique and evolves to a more complex expression. Please refer to Appendix D for more explanation.

hold.

The total good event E_1 is then given by

$$E_1 = \left(\bigcap_{s,a,1 \leqslant j \leqslant \check{J}} \check{E}^{(j)}(s,a) \right) \bigcap \left(\bigcap_{s,a,1 \leqslant j' \leqslant \bar{J}} \bar{E}^{(j')}(s,a) \right). \tag{13}$$

We claim that E_1 happens with large probability.

Lemma 3. $\mathbb{P}[E_1] \geqslant (1 - SAH(\check{J} + \bar{J})p)$.

The following statement shows that $\{Q_t\}$ is a sequence of non-increasing optimistic estimates of Q^*.

Proposition 4. *Conditioned on the event E_1, it holds that $Q_t(s,a) \geqslant Q^*(s,a)$ and $Q_{t+1}(s,a) \leqslant Q_t(s,a)$ for all $t \geqslant 1$ and (s,a).*

The proofs of Lemma 3, Proposition 4 and all the lemmas in the remaining part of this section can be found in Appendix C. Throughout the rest of this section, the analysis will be done assuming the successful event E_1.

5.2. Using Clipped Pseudo-Regret to Bound Sample Complexity

By the update rule (2), for any $t \geqslant 1$ and s, letting $a = \pi_t(s)$, we have that

$$\begin{aligned}
& V_t(s) - V^{\pi_t}(s) \\
& \leqslant \check{b}_t(s,a) + \frac{\gamma}{\check{n}_t(s,a)} \sum_{u=1}^{\check{n}_t(s,a)} V_{\check{l}_{t,u}(s,a)}(s_{\check{l}_{t,u}(s,a)+1}) \\
& \qquad - \gamma P_{s,a} V^{\pi_t} \\
& \leqslant 2\check{b}_t(s,a) + \gamma P_{s,a} \left(\frac{1}{\check{n}_t(s,a)} \sum_{u=1}^{\check{n}_t(s,a)} V_{\check{l}_{t,u}(s,a)} - V^{\pi_t} \right) \qquad (14) \\
& \leqslant 2\check{b}_t(s,a) + \gamma P_{s,a}(V_{\underline{\rho}_t(s,a)} - V^{\pi_t}) \qquad (15) \\
& = 2\check{b}_t(s,a) + \gamma P_{s,a}(V_{\underline{\rho}_t(s,a)} - V_t) + \gamma P_{s,a}(V_t - V^{\pi_t}). \qquad (16)
\end{aligned}$$

where Inequality (14) is due to the concentration inequality, which is part of the successful event E_1 defined in (41), and Inequality (15) holds because $\underline{\rho}_t(s,a) \leqslant \check{l}_{t,u}(s,a)$ for any $1 \leqslant u \leqslant \check{n}_t(s,a)$ and the fact V_t is non-increasing in t (Proposition 4).

On the other hand, we also have

$$\begin{aligned}
& V_t(s) - V^{\pi_t}(s) \\
& = Q_t(s,a) - Q^*(s,a) + Q^*(s,a) - Q^{\pi_t}(s,a) \\
& = Q_t(s,a) - Q^*(s,a) + \gamma P_{s,a}(V^* - V^{\pi_t}) \\
& \leqslant Q_t(s,a) - Q^*(s,a) + \gamma P_{s,a}(V_t - V^{\pi_t}). \qquad (17)
\end{aligned}$$

Combining (16) and (17), we have that

$$\begin{aligned}
& V_t(s) - V^{\pi_t}(s) \\
& \leqslant \min \{ 2\check{b}_t(s,a) + \gamma P_{s,a}(V_{\underline{\rho}_t(s,a)} - V_t), \\
& \qquad Q_t(s,a) - Q^*(s,a) \} + \gamma P_{s,a}(V_t - V^{\pi_t}). \qquad (18)
\end{aligned}$$

Therefore, we have that

$$\begin{aligned}
& \phi_t(s) = V_t(s) - (r(s,a) + \gamma P_{s,a} V_t) \\
& = V_t(s) - V^{\pi_t}(s) - \gamma P_{s,a}(V_t - V^{\pi_t}) \\
& \leqslant \min \{ 2\check{b}_t(s,a) + \gamma P_{s,a}(V_{\underline{\rho}_t(s,a)} - V_t), \\
& \qquad Q_t(s,a) - Q^*(s,a) \}. \qquad (19)
\end{aligned}$$

Define κ_t by setting $\kappa_t(s)$ as the **RHS** of (19). Recall that P_{π_t} is the matrix such that $\mathsf{P}_{\pi_t}(s) = P_{s,\pi_t(s)}$ for any $s \in \mathcal{S}$. By Bellman equation we have that

$$\begin{aligned}
V^*(s_t) - V^{\pi_t}(s_t) & \leqslant V_t - V^{\pi_t} \\
& = \sum_{i=0}^{\infty} (\gamma \mathsf{P}_{\pi_t})^i \phi_t \\
& \leqslant \sum_{i=0}^{H-1} (\gamma \mathsf{P}_{\pi_t})^i \phi_t + \frac{\epsilon}{8} \qquad (20) \\
& \leqslant \sum_{s,a} (\gamma \mathsf{P}_{\pi_t})^i \kappa_t + \frac{\epsilon}{8}.
\end{aligned}$$

By definition of $\kappa_t(s)$, and noting that $x \leqslant \mathrm{clip}(x,y) + y$ for any $x, y > 0$, we further have that

$$\begin{aligned}
& V^*(s_t) - V^{\pi_t}(s_t) \\
& \leqslant \sum_{s,a} w_t(s,a) \Big(\min \{ 2\check{b}_t(s,a) + \gamma P_{s,a}(V_{\underline{\rho}_t(s,a)} - V_t), \\
& \qquad Q_t(s,a) - Q^*(s,a) \} \Big) + \frac{\epsilon}{8} \qquad (21) \\
& \leqslant \sum_{s,a} w_t(s,a) \Big(\min \{ \mathrm{clip}(Q_t(s,a) - Q^*(s,a), \frac{3\epsilon}{4H}), \\
& \qquad 2\mathrm{clip}(\check{b}_t(s,a), \frac{\epsilon}{8H}) + \gamma P_{s,a} \mathrm{clip}(V_{\underline{\rho}_t(s,a)} - V_t, \frac{\epsilon}{8H}), \} \Big) \\
& \qquad + \sum_{s,a} w_t(s,a) \max \{ \frac{3\epsilon}{4H}, \frac{\epsilon}{4H} + \gamma P_{s,a} \mathbf{1} \cdot \frac{\epsilon}{8H} \} + \frac{\epsilon}{8} \\
& \leqslant \sum_{s,a} w_t(s,a) \Big(\min \{ \mathrm{clip}(Q_t(s,a) - Q^*(s,a), \frac{3\epsilon}{4H}), \\
& \qquad 2\mathrm{clip}(\check{b}_t(s,a), \frac{\epsilon}{8H}) + \gamma P_{s,a} \mathrm{clip}(V_{\underline{\rho}_t(s,a)} - V_t, \frac{\epsilon}{8H}), \} \Big) \\
& \qquad + \frac{7\epsilon}{8} \qquad (22)
\end{aligned}$$

where $w_t(s,a) = \mathbb{I}[\pi_t(s) = a] \cdot \sum_{i=0}^{H-1} \mathbf{1}_{s_t}^\top (\gamma \mathsf{P}_{\pi_t})^i \mathbf{1}_s$ is the expected discounted visit number of (s,a) in the next H

steps following π_t; and Inequality (22) is due to an averaging argument and the fact that $\sum_{s,a} w_t(s,a) \leqslant H$.

Let

$$\beta_t := \sum_{s,a} w_t(s,a) \min\Big\{\mathrm{clip}(Q_t(s,a) - Q^*(s,a), \frac{3\epsilon}{4H}),$$
$$\big(2\mathrm{clip}(\check{b}_t(s,a), \frac{\epsilon}{8H}) + \gamma P_{s,a}\mathrm{clip}(V_{\underline{\rho}_t(s,a)} - V_t, \frac{\epsilon}{8H})\big)\Big\}. \tag{23}$$

Define $\mathcal{T} = \{t \geqslant 1 | \beta_t > \frac{1}{8}\epsilon\}$. By (22) we have that the sample complexity of UCB-MULTISTAGE is bounded by

$$\sum_{t\geqslant 1} \mathbb{I}\left[V^*(s_t) - V^{\pi_t}(s_t) > \epsilon\right] \leqslant \sum_{t \geqslant 1} \mathbb{I}\left[\beta_t > \frac{1}{8}\epsilon\right] = |\mathcal{T}|.$$

To bound $|\mathcal{T}|$, we consider bounding $\sum_{t\in\mathcal{T}} \beta_t$ instead, since $\sum_{t\in\mathcal{T}} \beta_t \geqslant \frac{|\mathcal{T}|\epsilon}{8}$ and therefore $|\mathcal{T}| \leqslant (8/\epsilon) \cdot \sum_{t\in\mathcal{T}} \beta_t$. Let

$$\tilde{\beta}_t := \min\Big\{\mathrm{clip}(Q_t(s_t,a_t) - Q^*(s_t,a_t), \frac{3\epsilon}{4H})$$
$$2\mathrm{clip}(\check{b}_t(s_t,a_t), \frac{\epsilon}{8H}) + \gamma P_{s_t,a_t}\mathrm{clip}(V_{\underline{\rho}_t(s_t,a_t)} - V_t, \frac{\epsilon}{8H})\Big\}, \tag{24}$$

If π_t does not change very frequently, we have the approximation that $\beta_t \approx \sum_{i=0}^{H-1} \tilde{\beta}_{t+i}$. More formally, we prove the following statement (see Appendix C.3 for the proof).

Lemma 5. *For any $K \geqslant 1$, it holds that*

$$\mathbb{P}\Big[\sum_{t\in\mathcal{T}} \beta_t \geqslant 12KH^3\iota + 24SAH^4B\ln(N_0),$$
$$\sum_{t\geqslant 1} \tilde{\beta}_t < 3KH^2\iota\Big] \leqslant Hp.$$

By Lemma 5 and the discussion above, if we are able to bound $\sum_{t\geqslant 1} \tilde{\beta}_t \leqslant X$ (for $X \geqslant 3H^2\iota$), then with high probability, the sample complexity of UCB-MULTISTAGE is bounded by roughly $O(H/\epsilon) \cdot X$.

5.3. Bounding the Clipped Pseudo-Regret

We now turn to bound $\sum_{t\geqslant 1} \tilde{\beta}_t$. By (24),for t such that $N_t(s_t,a_t) < N_0$, we have that

$$\tilde{\beta}_t \leqslant \Big(2\mathrm{clip}(\check{b}_t(s_t,a_t), \frac{\epsilon}{8H})$$
$$+ \gamma P_{s_t,a_t}\mathrm{clip}(V_{\underline{\rho}_t(s_t,a_t)} - V_t, \frac{\epsilon}{8H})\Big), \tag{25}$$

and for $N_t(s,a) \geqslant N_0$, we have

$$\tilde{\beta}_t \leqslant \mathrm{clip}(Q_t(s_t,a_t) - Q^*(s_t,a_t), \frac{3\epsilon}{4H}). \tag{26}$$

The first term in (25) is exploration bonus for the type-I stage. For this term, we have the following lemma (see Appendix C.4 for proof).

Lemma 6.

$$\sum_{t\geqslant 1} \mathrm{clip}(\check{b}_t(s_t,a_t), \frac{\epsilon}{8H}) \leqslant O\left(\frac{SAB\iota}{\epsilon(1-\gamma)^4}\right).$$

The exploration bonus is increasing in B because more frequent updates implies fewer available samples in a single update due to the limitation in model-free RL.

For the second term in (25), let $\alpha_t = \mathbb{I}[N_t(s_t,a_t) < N_0]P_{s_t,a_t}\mathrm{clip}(V_{\underline{\rho}_t(s_t,a_t)} - V_t, \frac{\epsilon}{8H})$ for short. On benefit of type-II updates, we can ensure a decent convergence rate for Q_t (see Appendix C.7 for proof).

Lemma 7. *Conditioned on the successful event of E_1 defined in* (41)*, for any $\epsilon_1 \in [\epsilon, \frac{1}{1-\gamma}]$ it holds that*

$$\sum_{t=1}^{\infty} \mathbb{I}\left[V_t(s_t) - V^*(s_t)) \geqslant \epsilon_1\right]$$
$$\leqslant \sum_{t=1}^{\infty} \mathbb{I}\left[Q_t(s_t,a_t) - Q^*(s_t,a_t)) \geqslant \epsilon_1\right]$$
$$\leqslant O\left(\frac{SAH^5\ln(\frac{4H}{\epsilon})\iota}{\epsilon_1^2}\right). \tag{27}$$

By the basic convergence rate provided by Lemma 7, we have that (see Appendix C.5 for proof)

Lemma 8. *With probability $1 - (1 + 2SAH(\check{J} + \bar{J}))p$, it holds that*

$$\sum_{t\geqslant 1} \alpha_t \leqslant O\left(\frac{SAH^5\ln(\frac{4H}{\epsilon})\iota}{\epsilon B} + SABH^3 + SAH\ln(N_0)\right).$$

The term α_t reflects the difference of the value functions between the neighboring updates. As mentioned in Section 4, we can reduces this term by increasing B as long as $\frac{SAH^2\ln(\frac{4H}{\epsilon})\iota}{\epsilon B}$ is larger $SABH^3$. We highlight that Lemma 7 is necessary to derive Lemma 8 even when B is large. This is due to the nature of model-free RL algorithms: more frequent updates would incur large variances (and thus greater exploration bonuses) due to fewer available samples between updates. As a result, without type-II updates, simply increasing B would not guarantee a decent convergence rate. In contrast, the type-II updates use more available samples, incurring a smaller exploration bonus, and thus guarantees a decent convergence rate.

Moreover, by Lemma 7, we have the lemma below to bound the term in (26) (see Appendix C.8 for proof).

Lemma 9. *With probability $1 - (1 + 2SAH(\check{J} + \bar{J}))p$, for any $t \geqslant 1$ such that $N_t(s_t,a_t) \geqslant N_0$, it holds that*

$$\mathrm{clip}\left(Q_t(s_t,a_t) - Q^*(s_t,a_t), \frac{3\epsilon}{4H}\right) = 0.$$

Combining Lemma 6, Lemma 8 and Lemma 9, and by the definition of $\tilde{\beta}_t$, we have that

Lemma 10. *With probability* $1-(2+6SAH(\check{J}+\bar{J}))p$, $\sum_{t\geqslant 1}\tilde{\beta}_t$ *is bounded by*

$$O\left(\frac{SABH^4\iota}{\epsilon}+\frac{SAH^5\ln(\frac{4H}{\epsilon})\iota}{\epsilon B}+SABH^3\ln(N_0)\right).$$

5.4. Putting Everything Together

Invoking Lemma 5 with $K=\frac{c_2}{3H^2\iota}\Big(\frac{SABH^4\iota}{\epsilon}+\frac{SAH^5\ln(\frac{4H}{\epsilon})\iota}{\epsilon B}+SABH^3\ln(N_0)\Big)\geqslant 1$ for some large enough universal constant c_2, we have that conditioned on the successful event E_1,

$$\begin{aligned}
&\mathbb{P}\left[\sum_{t\in\mathcal{T}}\beta_t\geqslant 12KH^3\iota+24SAH^4B\ln(N_0)\right]\\
&\leqslant\mathbb{P}\Bigg[\sum_{t\in\mathcal{T}}\beta_t\geqslant 12KH^3\iota+24SAH^4B\ln(N_0),\\
&\qquad\qquad\qquad\qquad\sum_{t\geqslant 1}\tilde{\beta}_t<3KH^2\iota\Bigg]\\
&\quad+\mathbb{P}\left[\sum_{t\geqslant 1}\tilde{\beta}_t\geqslant 3KH^2\iota\right] &(28)\\
&\leqslant(4SAH(\check{J}+\bar{J})+H+2)p, &(29)
\end{aligned}$$

where the second term in (28) bounded due to Lemma 10. Combining Proposition 4 with (29), we obtain that with probability $1-(8SA(\check{J}+\bar{J})+(H+3))p$, it holds that

$$\begin{aligned}
\frac{|\mathcal{T}|\epsilon}{2}&\leqslant\sum_{t\in\mathcal{T}}\beta_t\\
&\leqslant O\left(\frac{SABH^5\iota}{\epsilon}+\frac{SAH^6\ln(\frac{4H}{\epsilon})\iota}{\epsilon B}+SAH^4B\ln(N_0)\right).
\end{aligned}\tag{30}$$

Noting that $B=\sqrt{H}$, we conclude that the number of ϵ-suboptimal steps is bounded by

$$\begin{aligned}
&O\left(\frac{SAH^{5.5}\ln(\frac{4H}{\epsilon})\iota}{\epsilon^2}+\frac{SAH^{4.5}\ln(N_0)}{\epsilon}\right)\\
&\leqslant O\left(\frac{SAH^{5.5}\ln(\frac{4H}{\epsilon})(\ln(N_0)+\iota)}{\epsilon^2}\right)
\end{aligned}$$

for any $\epsilon\in(0,\frac{1}{1-\gamma}]$. Noting that $H=\tilde{O}(\frac{1}{1-\gamma})$, $\check{J}=O(SAH\ln(N_0))$ and $\bar{J}=O(SAHB\ln(N_0))$, we finish the proof of Theorem 2 by replacing p with $\frac{p}{8SA(\check{J}+\bar{J})+H+3}$.

6. Conclusion

We design a stage-based model-free Q-learning Algorithm UCB-MultiStage-Advantage, which achieves a near-optimal sample complexity of $\tilde{O}\left(\frac{SA\ln(1/p)}{\epsilon^2(1-\gamma)^3}\right)$ for discounted reinforcement leaning problem asymptotically. By adjusting the number of stages, we also show a non-asymptotic sample complexity of $\tilde{O}\left(\frac{SA\ln(1/p)}{\epsilon^2(1-\gamma)^{5.5}}\right)$, which outperforms all previous model-free and model-based algorithms with tight dependence on S. We introduce a multi-stage update rule for Q-learning algorithm, which may be useful for other RL settings such as RL with linear function approximation.

References

Azar, M. G., Munos, R., Ghavamzadaeh, M., and Kappen, H. J. Speedy q-learning. 2011.

Azar, M. G., Osband, I., and Munos, R. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pp. 263–272. PMLR, 2017.

Brafman, R. I. and Tennenholtz, M. R-MAX - A general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3 (2):213–231, 2003.

Burnetas, A. N. and Katehakis, M. N. *Optimal Adaptive Policies for Markov Decision Processes*. 1997.

Dann, C. and Brunskill, E. Sample complexity of episodic fixed-horizon reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 2818–2826, 2015.

Dann, C., Lattimore, T., and Brunskill, E. Unifying PAC and regret: Uniform PAC bounds for episodic reinforcement learning. In *Advances in Neural Information Processing Systems 30: Annual Conference*, pp. 5713–5723, 2017.

Dann, C., Li, L., Wei, W., and Brunskill, E. Policy certificates: Towards accountable reinforcement learning. In *International Conference on Machine Learning*, pp. 1507–1516. PMLR, 2019.

Dong, K., Wang, Y., Chen, X., and Wang, L. Q-learning with ucb exploration is sample efficient for infinite-horizon mdp. *arXiv preprint arXiv:1901.09311*, 2019.

Even-Dar, E. and Mansour, Y. Learning rates for Q-learning. *Journal of Machine Learning Research*, 5(Dec):1–25, 2003.

Even-Dar, E., Mannor, S., and Mansour, Y. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7(Jun):1079–1105, 2006.

Freedman, D. A. et al. On tail probabilities for martingales. *the Annals of Probability*, 3(1):100–118, 1975.

Gheshlaghi, A., Munos, R., and Kappen, H. On the sample complexity of reinforcement learning with a generative mode. 2012.

Jin, C., Allen-Zhu, Z., Bubeck, S., and Jordan, M. I. Is Q-learning provably efficient? In *Advances in Neural Information Processing Systems*, pp. 4863–4873, 2018.

Kakade, S. *On the sample complexity of reinforcement learning.* PhD thesis, University of London, 2003.

Lattimore, T. and Hutter, M. Pac bounds for discounted mdps. In *International Conference on Algorithmic Learning Theory*, pp. 320–334. Springer, 2012.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533, 2015.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., and Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pp. 1928–1937, 2016.

Pazis, J., Parr, R. E., and How, J. P. Improving pac exploration using the median of means. In *Advances in Neural Information Processing Systems*, pp. 3898–3906, 2016.

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

Sidford, A., Wang, M., Wu, X., Yang, L., and Ye, Y. Near-optimal time and sample complexities for solving markov decision processes with a generative model. In *Advances in Neural Information Processing Systems*, pp. 5186–5196, 2018a.

Sidford, A., Wang, M., Wu, X., and Ye, Y. Variance reduced value iteration and faster algorithms for solving markov decision processes. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 770–787. SIAM, 2018b.

Strehl, A. L., Li, L., Wiewiora, E., Langford, J., and Littman, M. L. PAC model-free reinforcement learning. In *Proceedings of the 23rd International Conference on Machine learning*, pp. 881–888, 2006.

Szita, I. and Szepesvari, C. Model-based reinforcement learning with nearly tight exploration complexity bounds. In *Proceedings of the 27th International Conference on Machine Learning*, pp. 1031–1038, 2010.

Zhang, Z., Zhou, Y., and Ji, X. Almost optimal model-free reinforcement learning via reference-advantage decomposition. *arXiv preprint arXiv:2004.10019*, 2020.