Restless-UCB, an Efficient and Low-complexity Algorithm for Online Restless Bandits

Siwei Wang[1], **Longbo Huang**[2], **John C.S. Lui**[3]
[1]Department of Computer Science and Technology, Tsinghua University
wangsw2020@mail.tsinghua.edu.cn
[2]Institute for Interdisciplinary Information Sciences, Tsinghua University
longbohuang@mail.tsinghua.edu.cn
[3]Department of Computer Science and Engineering, The Chinese University of Hong Kong
cslui@cse.cuhk.edu.hk

Abstract

We study the online restless bandit problem, where the state of each arm evolves according to a Markov chain, and the reward of pulling an arm depends on both the pulled arm and the current state of the corresponding Markov chain. In this paper, we propose Restless-UCB, a learning policy that follows the explore-then-commit framework. In Restless-UCB, we present a novel method to construct offline instances, which only requires $O(N)$ time-complexity (N is the number of arms) and is exponentially better than the complexity of existing learning policy. We also prove that Restless-UCB achieves a regret upper bound of $\tilde{O}((N+M^3)T^{\frac{2}{3}})$, where M is the Markov chain state space size and T is the time horizon. Compared to existing algorithms, our result eliminates the exponential factor (in M, N) in the regret upper bound, due to a novel exploitation of the sparsity in transitions in general restless bandit problems. As a result, our analysis technique can also be adopted to tighten the regret bounds of existing algorithms. Finally, we conduct experiments based on real-world dataset, to compare the Restless-UCB policy with state-of-the-art benchmarks. Our results show that Restless-UCB outperforms existing algorithms in regret, and significantly reduces the running time.

1 Introduction

The restless bandit problem is a time slotted game between a player and the environment [50]. In this problem, there are N arms (or actions), and the state of each arm i evolves according to a Markov chain M_i, which makes one transition per time slot during the game (regardless of being pulled or not). At each time slot t, the player chooses one arm to pull. If he pulls arm i, he observes the current state $s_i(t)$ of M_i, and receives a random reward $x_i(t)$ that depends on i and $s_i(t)$, i.e., $\mathbb{E}[x_i(t)] = r(i, s_i(t))$ for some function r. The goal of the player is to maximize his expected cumulative reward during the time horizon T, i.e., $\mathbb{E}[\sum_{t=1}^{T} x_{a(t)}(t)]$, where $a(t) \in [N]$ denotes the pulled arm at time step t.

Restless bandit can model many important applications. For instance, in a job allocation problem, an operator allocates jobs to N different servers. The state of each server, i.e., the number of background jobs currently running at the server, can be modeled by a Markov chain, and it changes every time slot according to an underlying transition matrix [32, 20]. At each time slot, the operator allocates a job to one server, and receives the reward from that server, i.e., whether the job is completed, which depends on the current state of the server. At the same time, the operator can determine the current state of the chosen server based on its feedback. For servers that are not assigned jobs at the current

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

time slot, however, the operator does not observe their current state or transitions. The operator's objective is to maximize his cumulative reward in the T time slots.

Another application of the restless bandit model is in wireless communication. In this scenario, a base-station (player) transmits packets over N distinct channels to users. Each channel may be in "good" or "bad" state due to the channel fading condition, which evolves according to a two-state Markov chain [37, 40]. Every time, the player chooses one channel for packet transmission. If the transmission is successful, the player gets a reward of 1. The player also learns about the state of the channel based on receiver feedback. The goal of the player is to maximize his cumulative reward, i.e., deliver as many packets as possible, within the given period of time.

Most existing works on restless bandit focus on the offline setting, i.e., all parameters of the game are known to the player, e.g., [50, 48, 31, 8, 44]. In this setting, the objective is to search for the best policy of the player. In practice, one often cannot have the full system information beforehand. Thus, traditional solutions instead choose to solve the offline problem using empirical parameter values. However, due to the increasing sizes of the problem instances in practice, a small error on parameter estimation can lead to a large error, i.e., regret.

The online restless bandit setting, where parameters have to be learned online, has been gaining attention, e.g., [42, 13, 26, 23, 22, 33]. However, many challenges remain unsolved. First of all, existing policies may not perform close to the optimal offline one, e.g., [26] only considers the best policy that constantly pulls one arm. Second, for the class of Thompson Sampling based algorithms, e.g., [23, 22], theoretical guarantees are often established in the Bayesian setting, where the update methods can be computationally expensive when the likelihood functions are complex, especially for prior distributions with continuous support. Third, the existing policy with theoretical guarantee of a sublinear regret upper bound, i.e., colored-UCRL2 [33], suffers from an *exponential computation complexity* and a regret bound that is *exponential in the numbers of arms and states*, as it requires solving a set Bellman equations with an exponentially large space set.

In this paper, we aim to tackle the high computation complexity and exponential factor in the regret bounds for online restless bandit. Specifically, we consider a class of restless bandit problems with birth-death state Markov chains, and develop online algorithms to achieve a regret bound that is only polynomial in the numbers of arms and states. We emphasize that, birth-death Markov chains have been widely used to model real-world applications, e.g., queueing systems [24] and wireless communication [46], and are generalization of the two-state Markov chain assumption often made in prior works on restless bandits, e.g., [17, 28]. Our model can also be applied to many important applications, e.g., communications [3, 2], recommendation systems [30] and queueing systems [4].

The main contributions of this paper are summarized as follows:

- We consider a general class of online restless bandit problems with birth-death Markov structures and propose the Restless-UCB policy. Restless-UCB contains a novel method for constructing offline instances in guiding action selection, and only has an $O(N)$ complexity (N is the number of arms), which is exponentially better than that of colored-UCRL2, the state-of-the-art policy with theoretical guarantee [33] for online restless bandits.
- We devise a novel analysis and prove that Restless-UCB achieves an $\tilde{O}((N + M^3)T^{\frac{2}{3}})$ regret, where M is the Markov chain state space size and T is the time horizon. Our bound improves upon existing regret bounds in [33, 22], which are exponential in N, M. The novelty of our analysis lies in the exploitation of the sparsity in general restless bandit problems, i.e., each belief state can only transit to M other ones. This approach can also be combined with the analysis in [33, 22] to reduce the exponential factors in the regret bound to polynomial values (complexity remains exponential) in online restless bandit problems. Thus, our analysis can be of independent interest in online restless bandit analysis.
- We show that Restless-UCB can be combined with an efficient offline approximation oracle to guarantee $O(N)$ time-complexity and an $\tilde{O}(T^{\frac{2}{3}})$ approximation regret upper bound. Note that existing algorithms suffer from either an exponential complexity or no theoretical guarantee even with an efficient approximation oracle.
- We conduct experiments based on real-world datasets, and compare our policy with existing benchmarks. Our results show that Restless-UCB outperforms existing algorithms in both regret and running time.

1.1 Related Works

The offline restless bandit problem was first proposed in [50]. Since then, researchers concentrated on finding the exact best policy via index methods [50, 48, 28], i.e., first giving each arm an index, then choosing actions with the largest index and update their indices. However, index policies may not always be optimal. In fact, it has been shown that there exist examples of restless bandit problems where no index policy achieves the best cumulative reward [50]. [35] further shows that finding the best policy of any offline restless bandit model is a PSPACE-hard problem. As a result, researchers also worked on finding an approximate policy of restless bandit [27, 17].

There have also been works on online restless bandit. One special case is the stochastic multi-armed bandit [7, 25] in which the arms all have singe-state Markov chains. Under this setting, the best offline policy is to choose the action with the largest expected reward forever. Researchers also propose numerous policies to solve the online problem, classical algorithms include UCB policy [5] and the Thompson Sampling policy [43].

[42, 13, 26] considered the online restless bandit model with weak regret, i.e., comparing with single action policies (which is similar as the best policy in stochastic multi-armed bandit model), and they proposed UCB-based policies. Specifically, the algorithms choose to pull a single arm for a long period, so that the average reward during this period is close to the actual average reward of always pulling this single arm. Based on this fact, they established the upper confidence bounds for every arm, and showed that always choosing the action with the largest upper confidence bound achieves an $O(\log T)$ weak regret.

To solve the general online restless bandit problem without policy constraints, [33] showed that the problem can be regarded as a special online reinforcement learning problem [41]. In this setting, prior works adapted the idea of UCB and Thompson Sampling, and proposed policies with regret upper bound $O(D\sqrt{T})$, where D is the diameter of the game [21, 34, 1]. Based on these approaches, people proposed UCB-based policies, e.g., colored-UCRL2 [33], and Thompson Sampling policies, e.g., [23, 22] for the online restless bandit problem. Colored-UCRL2 directly applies the UCRL policy in [21], and leads to an $O(D\sqrt{T})$ regret upper bound. Also, to search for a best problem instance within a confidence set, it needs to solve a set of Bellman equations with an exponentially large space set, resulting in an exponential time complexity. To avoid this exponential time cost, [22] adapted the idea of Thompson Sampling in reinforcement learning [1], and achieves a Bayesian regret upper bound $O(D\sqrt{T})$. However, it requires a complicated method for updating the prior distributions to posterior ones when the likelihood functions are complex, especially for prior distributions with continuous support. In addition to the large time complexity, another challenge is that, the diameter D of the game is usually exponential with the size of the game, leading to a large regret bound. [23] considered a different setting, i.e., the episodic one, in which the game always restarts after L time steps. This way, it avoided the D factor in the regret bound and achieved an $O(L\sqrt{T})$ regret.

A related topic of restless bandit is non-stationary bandit problem [15, 9], in which the expected reward of pulling each arm may vary across time. In non-stationary bandit, people work on the settings with limited number of breakpoints (the time steps that the expected rewards change) [15] or limited variation on the expected rewards [9]. The main difference is that non-stationary bandit problem does not assume any inner structure about how the expected rewards vary. In restless bandit, we assume that they follow a Markov chain structure, and focus on learning this structure out by observing more information about it (i.e., except for the received reward, we also observe the current state of the chosen action). Therefore, the algorithm and analysis for restless bandit can be very different with those for non-stationary bandit.

2 Model Setting

Consider an online restless bandit problem $\mathcal{R}$ which has one player (decision maker) and N arms (actions) $\{1, \cdots, N\}$. Each arm $i \in \{1, \cdots, N\}$ is associated with a Markov chain M_i. All the Markov chains $\{M_i, i = 1, 2, ..., N\}$ have the same state space $S = \{1, 2, \cdots, M\}$,[1] but may have different transition matrices $\{\boldsymbol{P}_i, i = 1, 2, ..., N\}$ and state-dependent rewards $\{r(i, s), \forall\, i, s\}$ that

[1] This is not restrictive and is only used to simplify notations. Our analysis still works in the case where the state space S_i of Markov chain M_i satisfies that $|S_i| \leq M$.

are unknown to the player. The initial states of the arms are denoted by $\boldsymbol{s}(0) = [s_1(0), \cdots, s_N(0)]$. The game duration is divided into T time steps. In each time t, the player chooses an arm $a(t) \in [N]$ to play. We assume without loss of generality that there is a default arm 0, whose existence does not influence the theoretical results in this paper, and it is only introduced to simplify the proofs.

If the chosen arm $a(t)$ is not the default one, i.e., $a(t) > 0$, this action gives a reward $x(t) \in [0, 1]$, which is an independent random variable with expectation $r(a(t), s_{a(t)}(t))$, where $s_{a(t)}(t)$ is the state of $M_{a(t)}$ at time t. On the other hand, pulling arm 0 always results in a reward of 0. In every time step, the Markov chain of each arm makes a transition according to its transition matrix, regardless of whether it is pulled or not. However, the player only observes the current state and reward of the chosen arm. The current states of the rest of the arms are unknown. The goal of the player is to design an online learning policy to maximize the total expected reward during the game.

We use *regret* to evaluate the efficiency of the learning policy, which is defined as the expected gap between the offline optimal, i.e., the best policy under the full knowledge of all transition matrices and reward information, and the cumulative reward of the arm selecting algorithm. Specifically, let $\mu(\pi, \mathcal{R})$ denote the expected average reward under policy π for problem $\mathcal{R}$, i.e., $\mu(\pi, \mathcal{R}) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[x^\pi(t)]$, where $x^\pi(t)$ is the random reward at time t when applying policy π during the game. Then, we define the optimal average reward $\mu^*(\mathcal{R})$ as $\mu^*(\mathcal{R}) = \sup_\pi \mu(\pi, \mathcal{R})$. The regret of policy π is then defined as $Reg(T) = T\mu^*(\mathcal{R}) - \sum_{t=1}^{T} \mathbb{E}[x^\pi(t)]$.

Next, we state the assumptions made in the paper.

Assumption 1. *For any action i, we have $r(i, j) \geq r(i, k)$ for states $j < k$.*

This assumption is common in real-world applications, and is widely adopted in the restless bandit literature, e.g., [28, 2]. For instance, in job allocation, a busy server has a larger probability of dropping an incoming job than an idle server. Another example is in wireless communication, where transmitting in a good channel has a higher success probability than in a bad channel.

The next assumption is that all Markov chains have a birth-death structure, which are common in a wide range of problems including queueing systems [24] and wireless communications [46]. We also note that the birth-death Markov chain generalizes the two-state Markov chain assumption which was used in prior works of restless bandit [17, 28].

Assumption 2. *$P_i(j, k) = 0$ for any action i and state $|j - k| > 1$, where $P_i(j, k)$ is the probability that M_i transits from state j to k in one time step.*

The following assumption is a generalization of the positive-correlated assumption often made in the restless bandit literature, e.g., [2, 45].

Assumption 3. *For any action i, state $1 \leq k \leq M-1$, we have that $P_i(k, k+1) + P_i(k+1, k) \leq 1$.*

Finally, we assume that for any state j, the probability of going to any neighbor state is lower bounded by a constant. This assumption is rather mild, especially when we only have a finite number of states.

Assumption 4. *For any action i, state $|j - k| \leq 1$, we have that $P_i(j, k) \geq c_1$ for some constant $c_1 \in (0, 1)$.*

Under these assumptions, it is easy to verify that the Markov chains are ergodic. We thus denote the unique stationary distribution of M_i by $\boldsymbol{d}^{(i)} = [d_1^{(i)}, \cdots, d_M^{(i)}]$, and denote $d_{\min} \triangleq \min_{i,k} d_k^{(i)}$. For each transition matrix $\boldsymbol{P}_i$, we also define a neighbor space $\mathcal{P}_i$ of transition matrices as:

$$\mathcal{P}_i = \left\{ \tilde{\boldsymbol{P}}_i : \forall |j - k| \leq 1, |\tilde{P}_i(j, k) - P_i(j, k)| \leq \frac{2c_1}{3} \right\}.$$

Notice that any Markov chain $\tilde{M}_i$ with transition matrices $\tilde{\boldsymbol{P}}_i \in \mathcal{P}_i$ must be ergodic. Thus, the absolute value of the second largest eigenvalues of $\tilde{\boldsymbol{P}}_i$, denoted by $\lambda_{\tilde{\boldsymbol{P}}_i}$, is smaller than 1, which means $\lambda^i \triangleq \sup_{\tilde{\boldsymbol{P}}_i \in \mathcal{P}_i} \lambda_{\tilde{\boldsymbol{P}}_i} < 1$ and $\lambda_{\max} \triangleq \max_i \lambda^i < 1$.

3 Restless-UCB Policy

In this section, we present our Restless-UCB policy, whose pseudo-code is presented in Algorithm 1.

Algorithm 1 Restless-UCB Policy

1: **Input:** Time horizon T, learning function $m(T)$.
2: **for** $i = 1, 2, \cdots, N$ **do**
3: Choose arm i until there are $m(T)$ times we observe $s_i(t) = k$ for all states k.
4: **end for**
5: Let $\hat{P}_i(j,k)$'s and $\hat{r}(i,k)$'s be the empirical values of $P_i(j,k)$'s and $r(i,k)$'s.
6: Construct instance $\mathcal{R}'$ with $r'(i,k) = \hat{r}(i,k) + rad(T)$, $P_i'(k,k+1) = \hat{P}_i(k,k+1) - rad(T)$, $P_i'(k,k) = \hat{P}_i(k,k)$, $P_i'(k,k-1) = \hat{P}_i(k,k-1) + rad(T)$. Specifically, $P_i'(1,1) = \hat{P}_i(1,1) + rad(T)$ and $P_i'(M,M) = \hat{P}_i(M,M) - rad(T)$.
7: Find the optimal policy $\pi^{*\prime}$ for problem $\mathcal{R}'$, i.e., $\pi^{*\prime} = \texttt{Oracle}(\mathcal{R}')$.
8: **while true do**
9: Follow $\pi^{*\prime}$ for the rest of the game.
10: **end while**

Restless-UCB contains two phases: (i) the exploration phase (lines 2-4) and (ii) the exploitation phase (lines 5-10). The goal of the exploration phase is to learn the parameters $\{\boldsymbol{P}_i, i = 1, 2, ..., N\}$ and $\{r(i,s), \forall\, i, s\}$ as accurate as possible. To do so, Restless-UCB pulls each arm until there are sufficient observations, i.e., for any action i and state k, we observe the next transition and the given reward for at least $m(T)$ number of times ($m(T)$ to be specified later). Once there are $m(T)$ observations, the empirical values of $\{\boldsymbol{P}_i, i = 1, 2, ..., N\}$ and $\{r(i,s), \forall\, i, s\}$ (represented by $\hat{P}_i(j,k)$ and $\hat{r}(i,k)$) have a bias within $rad(T) \triangleq \sqrt{\frac{\log T}{2m(T)}}$ with high probability [19]. The key here is to choose the right $m(T)$ to balance accuracy and complexity.

In the exploitation phase, we first use an offline oracle `Oracle` (using oracle is a common approach in bandit problems [10, 11, 49]) to construct the optimal policy for our estimated model instance based on empirical data, and then apply this policy for the rest of the game. The key to guarantee good performance of our algorithm is that, instead of using the empirical data directly, in Restless-UCB, *we carefully construct an offline problem instance to guide our policy search*. Specifically, we use the *upper* confidence bound values for $P_i(k,k-1)$'s and $r(i,k)$'s, the *lower* confidence bounds for $P_i(k,k+1)$'s, and the empirical values for $P_i(k,k)$'s. As shown in line 6 of Algorithm 1, we set $r'(i,k) = \hat{r}(i,k) + rad(T)$, $P_i'(k,k+1) = \hat{P}_i(k,k+1) - rad(T)$, $P_i'(k,k) = \hat{P}_i(k,k)$, $P_i'(k,k-1) = \hat{P}_i(k,k-1) + rad(T)$ in the estimated offline instance $\mathcal{R}'$. This method allows us to use $O(N)$ complexity to construct a good offline instance, which is greatly better than the exponential cost in [33].

Next, we view the offline restless bandit instance as a MDP. The state of the MDP (referred to as belief state in POMDP [16]), is defined as to be $z = \{(s_i, \tau_i)\}_{i=1}^N$, where s_i is the last observed state of M_i, τ_i is the number of time steps elapsed since the last time we observe M_i, and the action set is $\{1, 2, \cdots, N\}$. Once we choose action i under belief state z, the belief state will transit to z_k^i with probability p_k, where p_k equals to the k-th term in vector $\boldsymbol{e}_{s_i}\boldsymbol{P}_i^{\tau_i}$ ($\boldsymbol{e}_{s_i}$ represents the one hot vector with only the s_i-th term equals to 1 and the rest are 0), and $z_k^i = \{(s_j, \tau_j + 1)\}_{j \neq i} \cup \{k, 1\}$, i.e., $\{s_i, \tau_i\}$ is updated by $\{k, 1\}$ according to the observation, while other actions only have their τ_j values increase by one. We then use `Oracle` to find out the optimal policy $\pi^{*\prime}$ for the empirical offline instance $\mathcal{R}'$. After that, Algorithm 1 uses policy $\pi^{*\prime}$ for the rest of the game, even though the actual model is $\mathcal{R}$ rather than $\mathcal{R}'$.

Note that by choosing $m(T) = o(T)$, the major part of the game will be in the exploitation phase, whose size is close to T. Thus, the regret in the exploitation phase is about $T(\mu(\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R}))$, where π^* is the best policy of the origin problem $\mathcal{R}$. To bound the regret, we divide the gap into two parts and analyze them separately, i.e., $T[\mu(\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R})] = T[\mu(\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R}')] + T[\mu(\pi^{*\prime}, \mathcal{R}') - \mu(\pi^{*\prime}, \mathcal{R})]$.

Roughly speaking, our estimation in the exploration phase makes the probability of transitioning to a lower index state (which has a higher expected reward) larger in any Markov chain M_i. Thus, the corresponding estimated reward is also larger. This way, one can guarantee that with high probability, the average reward of applying $\pi^{*\prime}$ in $\mathcal{R}'$ is larger than that of applying π^* in the original model $\mathcal{R}$, i.e., the first term $T[\mu(\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R}')]$ is less than or equal to zero. The second term is the average reward gap between applying the same policy in different problem instance $\mathcal{R}$ and $\mathcal{R}'$. Since

our estimation ensures that $\mathcal{R}'$ and $\mathcal{R}$ only has a bounded gap, this term can also be bounded. The regret bound and theoretical analysis are shown in details in Section 4.

We emphasize that although Restless-UCB has a similar form as an "explore-then-commit" policy, e.g., [29, 36, 14], the key novelty of our scheme lies in the method of constructing offline problem instances from empirical data. Our method also only requires $O(N)$ time to search for a better problem instance within the confidence set, which greatly reduces the running time of our algorithm. In contrast, existing algorithms, e.g., [33], take exponential time for this step and incur an exponential (in N) implementation cost.

As described before, our method chooses to use upper (lower) confidence bounds of the transition probabilities in $\boldsymbol{P}_i$. If observations for different arms are interleaved with each other, it will be very difficult to utilize them directly for updating the confidence interval of $\boldsymbol{P}_i$, since they are observations of transition matrix $\boldsymbol{P}_i^\tau$ with $\tau > 1$ but not $\boldsymbol{P}_i$. Indeed, according to [18], it is already difficult to calculate the τ-th roots of stochastic matrices, let alone finding the confidence intervals. Therefore, we construct Markov chain M_i' in the offline instance $\mathcal{R}'$ by continuously pulling a single arm i for long time. This is why we choose to use an "explore-then-commit" framework instead of a UCRL framework.

4 Theoretical Analysis

In this section, we present our main theoretical results. The complete proofs are referred to the Appendix A in supplementary file.

4.1 Restless-UCB with Efficient Offline Oracles

We first consider the case when there exists an efficient `Oracle` for the offline restless bandit problem.

Theorem 1. *If* `Oracle` *returns the optimal policy, then under Assumptions 1, 2, 3 and 4, the regret of Restless-UCB with $m(T) = T^{\frac{2}{3}}$ in an online restless bandit problem $\mathcal{R}$ satisfies:*

$$Reg(T) = \tilde{O}\left(\left(\frac{N}{d_{\min}} + \frac{M^3}{(1-\lambda_{\max})^2}\right) T^{\frac{2}{3}}\right).$$

Although our setting focuses on birth-death Markov state transitions, we note that it is still general and applies to a wide range of important applications, e.g., communications [3, 2], recommendation systems [30] and queueing systems [4]. Focusing on this setting allows us to design algorithms with much lower complexity, which can be of great interest in practice. Existing low-complexity policies, though applicable to general MDP problems, do not have theoretical guarantees. For example, the UCB-based policies in [42, 13, 26] suffer from a $\Theta(T)$ regret (although their weak regret upper bound is $O(\log T)$), while the Thompson Sampling policy only has a sub-linear regret upper bound in Bayesian setting. The colored-UCRL2 policy [33], which possesses a sub-linear regret bound of $O(D\sqrt{T})$ with respect to T, suffers from an exponential implementation complexity (in N), even with an efficient oracle. Our Restless-UCB policy only requires a polynomial complexity (refer to Line 6 in Algorithm 1) with an efficient offline oracle, and achieves a rigorous sub-linear regret upper bound.

The reason why the regret bound of Restless-UCB is slightly worse than colored-UCRL2 is because the observations in the exploitation phase are not used for updating the parameters of the game (the observations for different arms in the exploitation phase are interleaved with each other and are hard to be used as we discussed before). This means that only $\tilde{O}(T^{\frac{2}{3}})$ observations are used in estimating the offline problem instance, resulting in a bias of $\tilde{O}(T^{-\frac{1}{3}})$ according to [19]. Colored-UCRL2 tackles this problem (i.e., utilize all the observations) by updating transition vectors for all the possible (s_i, τ_i), and finding a policy based on all these transition vectors. As a result, it needs to work with an exponential space set and results in an exponential time complexity. We instead sacrifice a little on the regret to obtain a significant reduction in the implementation complexity. We also emphasize that the colored-UCRL2 policy cannot simplify its implementation even under our assumptions, due to its need to compute the best policy based on all transition vectors for all (exponentially many) (s_i, τ_i) pairs. Moreover, the factor in the Restless-UCB's regret bound is $\frac{N}{d_{\min}} + \frac{M^3}{(1-\lambda_{\max})^2}$,

which is polynomial with N, M, and is *exponentially* better than the factor D in regret bounds of colored-UCRL2 [33] or Thompson Sampling [22], which is the diameter of applying a learning policy to the problem and is exponential in N, M. In Appendix B of the supplementary file, we also prove that our analysis can be adopted to reduce the D factor in their regret bounds to polynomial values. This shows that our analysis approach can be of independent interest for analyzing online restless bandit problems.

Now we highlight two key ideas of our theoretical analysis, each summarized in one lemma. They enable us to achieve polynomial time complexity and reduce the exponential factor in regret bounds.

Lemma 1. *Conditioning on event $\mathcal{E}$, we have $\mu(\pi^*, \mathcal{R}) \leq \mu(\pi^{*\prime}, \mathcal{R}')$. Here*

$$\mathcal{E} = \{\forall i, |j-k| \leq 1, |P_i(j,k) - \hat{P}_i(j,k)| \leq rad(T), |r(i,k) - \hat{r}(i,k)| \leq rad(T)\}.$$

Remark 1. *Conditioning on event $\mathcal{E}$, which is guaranteed to happen with high probability, the probability of transitioning to a lower index state (which has a higher expected reward) in $\mathcal{R}'$ is always larger than that one in $\mathcal{R}$. Lemma 1 shows that we can obtain a higher average reward in $\mathcal{R}'$ than in $\mathcal{R}$. This implies that we can efficiently (with $O(N)$ complexity) construct a better instance within the confidence set, which is an important step in analyzing restless MDPs, e.g., [33], whereas the prior work takes an exponential time for the construction.*

Lemma 2. *Conditioning on event $\mathcal{E}$, $\mu(\pi^{*\prime}, \mathcal{R}') - \mu(\pi^{*\prime}, \mathcal{R}) = \tilde{O}\left(\frac{M^3}{(1-\lambda_{\max})^2} T^{-\frac{1}{3}}\right)$.*

Remark 2. *Existing results in [33, 22] treat the restless bandit game as a general MDP. Doing so results in the factor D in their regret upper bounds (D is the diameter of the game and is exponential in the game size). In our case, the factor $\frac{M^3}{(1-\lambda_{\max})^2}$ in Lemma 2 is polynomial with the the game size. This improvement comes from a novel exploitation in the sparsity in general restless bandit problems, i.e., each belief state can only transit to M other ones. This novel result can also help to reduce the exponential factors in previous regret bounds, e.g., [33, 22], to polynomial ones in general restless bandit problems, though their complexities remain exponential (see details in Appendix B of the supplementary file), and can be of independent interest in analyzing online restless bandit problems.*

4.2 Restless-UCB with Efficient Offline Approximate Oracles

In this section, we show how Restless-UCB can be combined with approximate oracles to achieve good regret performance. In practical applications, finding the optimal policy of the offline problem is in general NP-hard [35] and one can only obtain efficient approximate solutions [27, 17]. As a result, how to perform learning efficiently and achieve low regret when `Oracle` can only return an approximation policy, e.g., [10, 11, 49], is of great interest and importance in designing low-complexity and efficient learning policies.

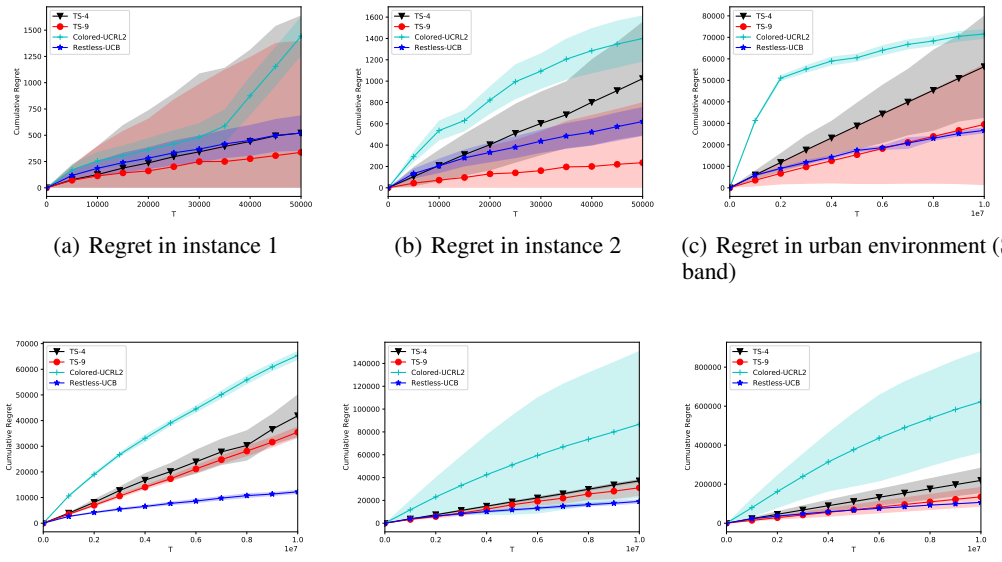
The definitions of approximate policies and approximation regret is given below.

Definition 1. *For an offline restless bandit instance $\mathcal{R}$, an approximate policy $\tilde{\pi}$ with approximate ratio $\lambda > 0$ satisfies that $\mu(\tilde{\pi}, \mathcal{R}) \geq \lambda \mu^*(\mathcal{R})$.*

Definition 2. *For an online restless bandit instance $\mathcal{R}$, the approximation regret with approximate ratio $\lambda > 0$ for learning policy π is defined as $Reg(T, \lambda) = \lambda \mu^*(\mathcal{R}) - \sum_{t=1}^{T} \mathbb{E}[x^{\pi}(t)]$.*

Theorem 2. *If `Oracle` returns an approximate policy with ratio λ, then under Assumptions 1, 2, 3 and 4, the approximation regret (with approximate ratio λ) of Restless-UCB with $m(T) = T^{\frac{2}{3}}$ in an online restless bandit problem $\mathcal{R}$ is upper bounded by $\tilde{O}(T^{\frac{2}{3}})$.*

Theorem 2 shows that Restless-UCB can be combined with approximate policies for the offline problem to achieve good performance, i.e., it can reduce the time complexity by applying an approximate oracle. This feature is not possessed by other policies such as colored-UCRL2 [23] or Thompson Sampling [22]. Specifically, colored-UCRL2 needs to solve a set of Bellman equations with exponential size to find out the better instance within the confidence set, thus using an approximate policy leads to a similar approximation regret but cannot reduce the time complexity. Thompson Sampling, on the other hand, can only apply the approximate approach in the Bayesian setting [47].



(a) Regret in instance 1

(b) Regret in instance 2

(c) Regret in urban environment (S-band)

(d) Regret in heavy tree-shadowed environment (S-band)

(e) Regret in suburban environment (L-band)

(f) Regret in urban environment (L-band)

Figure 1: Experiments: Comparison of regrets of different algorithms

5 Experiments

In this section, we present some of our experimental results. In all these experiments, we use the offline policy proposed by [28] as the offline oracle of restless bandit problems.

5.1 Experiments on Constructed Instance

We consider two problem instances, each instance contains two arms and each arm evolves according to a two-state Markov chain. In both instances, $r(i,2) = 0$ for any arm i, and all Markov chains start at state 2. In problem instance one, $r(1,1) = 1$, $r(2,1) = 0.8$, $P_1(1,1) = 0.7$, $P_1(2,2) = 0.8$, $P_2(1,1) = 0.5$ and $P_2(2,2) = 0.6$. In problem instance two, $r(1,1) = 0.8$, $r(2,1) = 0.4$, $P_1(1,1) = 0.7$, $P_1(2,2) = 0.9$, $P_2(1,1) = 0.7$ and $P_2(2,2) = 0.5$.

We compare three different algorithms, including Restless-UCB, state-of-the-art colored-UCRL2 [33] and Thompson Sampling policies with different priors, TS-9 and TS-4 [22]. In TS-9, the prior distributions of transition probability of Markov chain M_i are the uniform one on $[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]^2$ (for the values $P_i(1,1)$ and $P_i(2,2)$), and the prior distributions of different Markov chains are independent. In TS-4, the prior support is $[0.2, 0.4, 0.6, 0.8]^2$ instead.

The regrets of these algorithms are shown in Figures 1(a) and 1(b), which take average over 1000 independent runs. The expected regrets of Restless-UCB are smaller than TS-4 and colored-UCRL2, but larger than TS-9. This is because the support of prior distribution in TS-9 contains the real problem instance. As a result, its samples equal to the real problem instance with high probability. However, from the expected regrets of TS-4, one can see that when the support of its prior distribution is not close to the real instance, its expected regret grows linearly as T increases. Besides, compare with Restless-UCB, TS policy has a much larger variance on the regret, due to the high degree of randomness on the samples. Therefore, Restless-UCB is more robust against inaccurate estimation in the prior distributions, and has a more reliable theoretical guarantee due to its low variance on regret.

We also compare the average running times of the different algorithms. In this experiment, $T = 500000$. The four problem instances contain $N = 2, 3, 4, 5$ arms and each arm has a two-state Markov chain. The results in Table 1 take average over 50 runs of a single-threaded program (running on an Intel E5-2660 v3 workstation with 256GB RAM). They show that Restless-UCB policy is

much more efficient when there are more arms (particularly compared to colored-UCRL2). The time complexity of colored-UCRL2 grows exponentially as N grows up, while the time complexity of Restless-UCB only has a minimal increase.

Table 1: Average running times of different algorithms

Algorithm	2 arms	3 arms	4 arms	5 arms
Restless-UCB	4s	5s	5s	5s
TS-9	4s	6s	8s	10s
TS-4	3s	4s	6s	9s
Colored-UCRL2	5s	326s	8892s	129387s

5.2 Experiments with Real Data Set

We also use real datasets to compare the behavior of different algorithms.

Here we use the wireless communication dataset in [37]. It is a setting on digital video broadcasting satellite services to handheld devices via land mobile satellite. [37] provided the parameters of two-state Markov chain representations on the channel model in three different environments, including urban, suburban and heavy tree-shadowed environments. In this experiment, different elevation angles of antenna are represented as arms, and different elevation angles correspond to different channel parameters, including the transition matrices and propagation impairments. Our goal is to correctly transmit as many data packets as possible within a time horizon T. We use the transition probability matrices in Tables IV and VI in [37]. We also use the average direct signal mean given in Tables III and V in [37] as the expected reward. In Figure 1(c), we consider communicating via S-band under the urban environment, and one can choose the elevation angle to be either 40° or 80°. In Figure 1(d), we consider communicating via S-band under the heavy tree-shadowed environment, and one can choose the elevation angle to be either 40° or 80°. In Figure 1(e), we consider communicating via L-band under the suburban environment, and one can choose the elevation angle to be 50°, 60° or 70°. In Figure 1(f), one consider communicating via L-band under the urban environment, and we can choose the elevation angle to be either 10°, 20°, 30° or 40°. All of these results take average over 200 independent runs.

One can see that Restless-UCB performs the best in all these experiments, it achieves the smallest expected regret and the smallest variance on regret. As mentioned before, the TS policy suffers from a linear expected regret since its support does not contain the real transition matrices, and it has a large variance on regret at the same time. Although colored-UCRL2 achieves a sub-linear regret, it suffers from a large constant factor and performs worse than Restless-UCB. When there are more arms (see Figures 1(e) and 1(f)), colored-UCRL2 also suffers from a large variance on regret. These results demonstrate the effectiveness of Restless-UCB.

6 Conclusion

In this paper, we propose a low-complexity and efficient algorithm, called Restless-UCB, for online restless bandit. We show that Restless-UCB achieves a sublinear regret upper bound $\tilde{O}(T^{\frac{2}{3}})$, with a polynomial time implementation complexity. Our novel analysis technique also helps to reduce both the time complexity of the policy and the exponential factor in existing regret upper bounds. We conduct experiments based on real-world datasets to show that Restless-UCB outperforms existing benchmarks and has a much shorter running time.

Broader Impact

Online restless bandit model has found applications in many important areas such as wireless communications [3, 2], recommendation systems [30] and queueing systems [4]. Existing results face challenges including exponential implementation-complexity and regret bounds that are exponential in the size of the game [33, 22, 23]. Our Restless-UCB algorithm offers a novel approach that enjoys $O(N)$ time-complexity to implement, which greatly reduces the running time in real applications.

Moreover, our analysis reduces the exponential factor in the regret upper bound to a polynomial one. Our work contributes to designing low-complexity and efficient learning policies for online restless bandit problem and can likely find applications in a wide range of areas.

Acknowledgments and Disclosure of Funding

The work of Siwei Wang and Longbo Huang was supported in part by the National Natural Science Foundation of China Grant 61672316, the Zhongguancun Haihua Institute for Frontier Information Technology and the Turing AI Institute of Nanjing.

The work of John C.S. Lui is supported in part by the GRF 14201819.

References

[1] S. Agrawal and R. Jia. Optimistic posterior sampling for reinforcement learning: worst-case regret bounds. In *Advances in Neural Information Processing Systems*, pages 1184–1194, 2017.

[2] S. H. A. Ahmad and M. Liu. Multi-channel opportunistic access: A case of restless bandits with multiple plays. In *2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1361–1368. IEEE, 2009.

[3] S. H. A. Ahmad, M. Liu, T. Javidi, Q. Zhao, and B. Krishnamachari. Optimality of myopic sensing in multichannel opportunistic access. *IEEE Transactions on Information Theory*, 55(9):4040–4050, 2009.

[4] P. Ansell, K. D. Glazebrook, J. Nino-Mora, and M. O'Keeffe. Whittle's index policy for a multi-class queueing system with convex holding costs. *Mathematical Methods of Operations Research*, 57(1):21–39, 2003.

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

[6] P. L. Bartlett and A. Tewari. Regal: a regularization based algorithm for reinforcement learning in weakly communicating mdps. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 35–42, 2009.

[7] D. A. Berry and B. Fristedt. *Bandit problems: sequential allocation of experiments (Monographs on statistics and applied probability)*. Springer, 1985.

[8] D. Bertsimas and J. Niño-Mora. Restless bandits, linear programming relaxations, and a primal-dual index heuristic. *Operations Research*, 48(1):80–90, 2000.

[9] O. Besbes, Y. Gur, and A. Zeevi. Stochastic multi-armed-bandit problem with non-stationary rewards. In *Advances in Neural Information Processing Systems*, pages 199–207, 2014.

[10] W. Chen, Y. Wang, and Y. Yuan. Combinatorial multi-armed bandit: General framework and applications. In *International Conference on Machine Learning*, pages 151–159, 2013.

[11] W. Chen, Y. Wang, Y. Yuan, and Q. Wang. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *The Journal of Machine Learning Research*, 17(1):1746–1778, 2016.

[12] G. E. Cho and C. D. Meyer. Comparison of perturbation bounds for the stationary distribution of a markov chain. *Linear Algebra and its Applications*, 335(1-3):137–150, 2001.

[13] W. Dai, Y. Gai, B. Krishnamachari, and Q. Zhao. The non-bayesian restless multi-armed bandit: A case of near-logarithmic regret. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2940–2943. IEEE, 2011.

[14] A. Garivier, T. Lattimore, and E. Kaufmann. On explore-then-commit strategies. In *Advances in Neural Information Processing Systems*, pages 784–792, 2016.

[15] A. Garivier and E. Moulines. On upper-confidence bound policies for non-stationary bandit problems. *arXiv preprint arXiv:0805.3415*, 2008.

[16] B. Givan and R. Parr. An introduction to markov decision processes. *Purdue University*, 2001.

[17] S. Guha, K. Munagala, and P. Shi. Approximation algorithms for restless bandit problems. *Journal of the ACM (JACM)*, 58(1):3, 2010.

[18] N. J. Higham and L. Lin. On p th roots of stochastic matrices. *Linear Algebra and its Applications*, 435(3):448–463, 2011.

[19] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[20] P. Jacko. Restless bandits approach to the job scheduling problem and its extensions. *Modern trends in controlled stochastic processes: theory and applications*, pages 248–267, 2010.

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

[22] Y. H. Jung, M. Abeille, and A. Tewari. Thompson sampling in non-episodic restless bandits. *arXiv preprint arXiv:1910.05654*, 2019.

[23] Y. H. Jung and A. Tewari. Regret bounds for thompson sampling in episodic restless bandit problems. In *Advances in Neural Information Processing Systems*, pages 9005–9014, 2019.

[24] L. Kleinrock. *Queueing systems, volume 2: Computer applications*, volume 66. Wiley New York, 1976.

[25] T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[26] H. Liu, K. Liu, and Q. Zhao. Logarithmic weak regret of non-bayesian restless multi-armed bandit. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1968–1971. IEEE, 2011.

[27] K. Liu and Q. Zhao. On the myopic policy for a class of restless bandit problems with applications in dynamic multichannel access. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 3592–3597. IEEE, 2009.

[28] K. Liu and Q. Zhao. Indexability of restless bandit problems and optimality of whittle index for dynamic multichannel access. *IEEE Transactions on Information Theory*, 56(11):5547–5567, 2010.

[29] R. J. Maurice. A minimax procedure for choosing between two populations using sequential sampling. *Journal of the Royal Statistical Society: Series B (Methodological)*, 19(2):255–261, 1957.

[30] R. Meshram, D. Manjunath, and A. Gopalan. A restless bandit with no observable states for recommendation systems and communication link scheduling. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 7820–7825. IEEE, 2015.

[31] J. Nino-Mora. Restless bandits, partial conservation laws and indexability. *Advances in Applied Probability*, 33(1):76–98, 2001.

[32] J. Niño-Mora. Dynamic priority allocation via restless bandit marginal productivity indices. *Top*, 15(2):161–198, 2007.

[33] R. Ortner, D. Ryabko, P. Auer, and R. Munos. Regret bounds for restless markov bandits. In *International Conference on Algorithmic Learning Theory*, pages 214–228. Springer, 2012.

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

[35] C. H. Papadimitriou and J. N. Tsitsiklis. The complexity of optimal queuing network control. *Mathematics of Operations Research*, 24(2):293–305, 1999.

[36] V. Perchet, P. Rigollet, S. Chassang, and E. Snowberg. Batched bandit problems. In *Conference on Learning Theory*, pages 1456–1456, 2015.

[37] R. Prieto-Cerdeira, F. Perez-Fontan, P. Burzigotti, A. Bolea-Alamañac, and I. Sanchez-Lago. Versatile two-state land mobile satellite channel model with first application to dvb-sh analysis. *International Journal of Satellite Communications and Networking*, 28(5-6):291–315, 2010.

[38] J. S. Rosenthal. Convergence rates for markov chains. *Siam Review*, 37(3):387–405, 1995.

[39] E. Seneta. Sensitivity analysis, ergodicity coefficients, and rank-one updates for finite markov chains. *Numerical Solutions of Markov Chains*, pages 121–129, 1991.

[40] S. Sheng, M. Liu, and R. Saigal. Data-driven channel modeling using spectrum measurement. *IEEE Transactions on Mobile Computing*, 14(9):1794–1805, 2014.

[41] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

[42] C. Tekin and M. Liu. Online learning of rested and restless bandits. *IEEE Transactions on Information Theory*, 58(8):5588–5611, 2012.

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

[44] I. M. Verloop et al. Asymptotically optimal priority policies for indexable and nonindexable restless bandits. *The Annals of Applied Probability*, 26(4):1947–1995, 2016.

[45] P.-J. Wan and X. Xu. Weighted restless bandit and its applications. In *2015 IEEE 35th International Conference on Distributed Computing Systems*, pages 507–516. IEEE, 2015.

[46] H. S. Wang and N. Moayeri. Finite-state markov channel-a useful model for radio communication channels. *IEEE transactions on vehicular technology*, 44(1):163–171, 1995.

[47] S. Wang and W. Chen. Thompson sampling for combinatorial semi-bandits. In *International Conference on Machine Learning*, pages 5101–5109, 2018.

[48] R. R. Weber and G. Weiss. On an index policy for restless bandits. *Journal of Applied Probability*, 27(3):637–648, 1990.

[49] Z. Wen, B. Kveton, and A. Ashkan. Efficient learning in large-scale combinatorial semi-bandits. In *International Conference on Machine Learning*, pages 1113–1122, 2015.

[50] P. Whittle. Restless bandits: Activity allocation in a changing world. *Journal of applied probability*, 25(A):287–298, 1988.

Supplementary Material

A Proof of Theorem 1

In this section, we first propose the proofs of our two key lemmas, i.e., Lemma 1 and Lemma 2. Then we state other lemmas and facts that are helpful in the proof of Theorem 1. Finally, we show the complete proof of Theorem 1.

In the following, we concentrate on the case that $rad(T) \leq \frac{c_1}{3}$. If $rad(T) \geq \frac{c_1}{3}$, then $T = \tilde{O}(\frac{1}{c_1^3})$, which implies that the regret is at most $\tilde{O}(\frac{1}{c_1^3})$.

A.1 Proof of Lemma 1

We first introduce a useful definition:

Definition 3. *For two vectors $\boldsymbol{v}$ and $\boldsymbol{v}'$ with M dimensions, $\boldsymbol{v} \gtrsim \boldsymbol{v}'$ if for all $k \leq M$, $\sum_{j=1}^{k} v_j \geq \sum_{j=1}^{k} v'_j$.*

Based on this definition, we have the following lemmas.

Lemma 3. *Under Assumptions 2, 4 and conditioning on event $\mathcal{E}$, we have that $\boldsymbol{P}'_i(k) \gtrsim \boldsymbol{P}_i(k)$, where $\boldsymbol{P}'_i(k)$ and $\boldsymbol{P}_i(k)$ represent the transition vectors of arm i under state k in our estimated model $\mathcal{R}'$ and origin model $\mathcal{R}$, respectively.*

Proof. Note that there are only three non-zero values in $\boldsymbol{P}_i(k)$ (and $\boldsymbol{P}'_i(k)$, respectively), i.e., $P_i(k, k-1)$, $P_i(k,k)$ and $P_i(k,k+1)$ ($P'_i(k,k-1)$, $P'_i(k,k)$ and $P'_i(k,k+1)$, respectively). Then we only need to prove that conditioning on event $\mathcal{E}$, we have that $P'_i(k,k+1) \leq P_i(k,k+1)$ and $P'_i(k,k-1) \geq P_i(k,k-1)$. This is given by definition of $\mathcal{E}$ and $\boldsymbol{P}'_i(k)$'s directly. □

Lemma 4. *Under Assumptions 2, 3, for any arm i and state k, we have that $\boldsymbol{P}_i(k) \gtrsim \boldsymbol{P}_i(k+1)$.*

Proof. Note that $P_i(k,k-1) \geq 0 = P_i(k+1,k-1)$ and $P_i(k,k-1) + P_i(k,k) + P_i(k,k+1) = 1 \geq 1 - P_i(k+1,k+2) = P_i(k+1,k-1) + P_i(k+1,k) + P_i(k+1,k+1)$. Thus, the only thing we need to prove is that $P_i(k,k-1) + P_i(k,k) \geq P_i(k+1,k)$.

Since we have

$$P_i(k,k-1) + P_i(k,k) = 1 - P_i(k,k+1) \geq P_i(k+1,k),$$

where the inequality is because of Assumption 3, we finish the proof of this lemma. □

Lemma 5. *Under Assumptions 2, 4 and conditioning on event $\mathcal{E}$, for any arm i and probability vector $\boldsymbol{v}$, we have that $\boldsymbol{v}\boldsymbol{P}'_i \gtrsim \boldsymbol{v}\boldsymbol{P}_i$.*

Proof. Define $(\boldsymbol{v})_1^k = \sum_{j=1}^{k} v_j$, then we only need to prove that for any arm i and state k, we have that $(\boldsymbol{v}\boldsymbol{P}'_i)_1^k \geq (\boldsymbol{v}\boldsymbol{P}_i)_1^k$.

Note that

$$\begin{aligned}
(\boldsymbol{v}\boldsymbol{P}'_i)_1^k &= \sum_{j=1}^{M} (v_j \boldsymbol{P}'_i(j))_1^k \\
&= \sum_{j=1}^{M} v_j (\boldsymbol{P}'_i(j))_1^k \\
&\geq \sum_{j=1}^{M} v_j (\boldsymbol{P}_i(j))_1^k \qquad (1) \\
&= (\boldsymbol{v}\boldsymbol{P}_i)_1^k,
\end{aligned}$$

where Eq. (1) is because that conditioning on event $\mathcal{E}$, we have that $\boldsymbol{P}'_i(k) \gtrsim \boldsymbol{P}_i(k)$ (Lemma 3). □

Algorithm 2 $\hat{\pi}^*$ based on π^*

Init: The real belief state is $z' = \{(s'_i, \tau'_i)\}_{i=1}^N$, and set the virtual belief state $z = z'$.
while true do
  Choose action $a(t)$ as π^* choose under z, and observe state $s'_i(t)$, reward $r'(a(t), s'_i(t))$.
  Set $\boldsymbol{v}' = \boldsymbol{e}_{s'_{a(t)}}(\boldsymbol{P}'_{a(t)})^{\tau_{a(t)}}$ and $\boldsymbol{v} = \boldsymbol{e}_{s_{a(t)}}(\boldsymbol{P}_{a(t)})^{\tau_{a(t)}}$.
  Update the $a(t)$-th term in z to be $(s_i(t), 1)$, where $s_i(t) = \mathsf{Correspond}(a(t), \boldsymbol{v}, \boldsymbol{v}', s'_i(t))$. For other terms $j \neq a(t)$, set $\tau_j = \tau_j + 1$. Observes virtual reward $r(a(t), s_i(t))$.
  Update the $a(t)$-th term in z' to be $(s'_i(t), 1)$, for other terms $j \neq a(t)$, set $\tau'_j = \tau'_j + 1$.
end while

Lemma 6. *Under Assumptions 2, 3 and conditioning on event $\mathcal{E}$, for any arm i and any probability vectors $\boldsymbol{v}$ and $\boldsymbol{v}'$ such that $\boldsymbol{v} \gtrsim \boldsymbol{v}'$, we have that $\boldsymbol{v}\boldsymbol{P}_i \gtrsim \boldsymbol{v}'\boldsymbol{P}_i$.*

Proof. We only need to prove that for any arm i and state k, we have that $(\boldsymbol{v}\boldsymbol{P}_i)_1^k \geq (\boldsymbol{v}'\boldsymbol{P}_i)_1^k$.

Note that

$$\begin{aligned}
(\boldsymbol{v}\boldsymbol{P}_i)_1^k - (\boldsymbol{v}'\boldsymbol{P}_i)_1^k &= \sum_{j=1}^{M}(v_j - v_j)(\boldsymbol{P}_i(j))_1^k \\
&= \left(\sum_{j'=1}^{M}(v_{j'} - v'_{j'})\right)(\boldsymbol{P}_i(M))_1^k + \sum_{j=1}^{M-1}\left(\sum_{j'=1}^{j}(v_{j'} - v'_{j'})\right)(\boldsymbol{P}_i(j) - \boldsymbol{P}_i(j+1))_1^k \\
&= 0 + \sum_{j=1}^{M-1}\left(\sum_{j'=1}^{j}(v_{j'} - v'_{j'})\right)(\boldsymbol{P}_i(j) - \boldsymbol{P}_i(j+1))_1^k \\
&= 0 + \sum_{j=1}^{M-1}(\boldsymbol{v} - \boldsymbol{v})_1^j(\boldsymbol{P}_i(j) - \boldsymbol{P}_i(j+1))_1^k \\
&\geq 0, \qquad (2)
\end{aligned}$$

where Eq. (2) is because that $(\boldsymbol{v} - \boldsymbol{v})_1^j \geq 0$ and $(\boldsymbol{P}_i(j) - \boldsymbol{P}_i(j+1))_1^k \geq 0$, since we have that $\boldsymbol{v} \gtrsim \boldsymbol{v}'$ and $\boldsymbol{P}_i(j) \gtrsim \boldsymbol{P}_i(j+1)$ by Lemma 4. □

Lemma 7. *Under Assumptions 2, 3, 4 and conditioning on event $\mathcal{E}$, for any arm i. any integer $\tau \geq 0$ and any probability vectors $\boldsymbol{v}$ and $\boldsymbol{v}'$ such that $\boldsymbol{v} \gtrsim \boldsymbol{v}'$, we have that $\boldsymbol{v}(\boldsymbol{P}'_i)^\tau \gtrsim \boldsymbol{v}'(\boldsymbol{P}_i)^\tau$.*

Proof. This is given by applying Lemmas 5 and 6 together. □

Based on these lemmas, we provide the proof of Lemma 1 here.

Lemma 1. *Conditioning on event $\mathcal{E}$, we have $\mu(\pi^*, \mathcal{R}) \leq \mu(\pi^{*\prime}, \mathcal{R}')$. Here*

$$\mathcal{E} = \{\forall i, |j - k| \leq 1, |P_i(j,k) - \hat{P}_i(j,k)| \leq rad(T), |r(i,k) - \hat{r}(i,k)| \leq rad(T)\}.$$

Proof. The key proof idea is to simulate policy π^*, i.e., emulate it as close as possible in $\mathcal{R}'$, using a fictitious policy $\hat{\pi}^*$ shown in Algorithm 2 , where $\boldsymbol{e}_k$ denotes the probability vector with the k-th term equals to 1.

At the beginning, $\hat{\pi}^*$ chooses the same action i as π^* does. However, since $\mathcal{R}$ and $\mathcal{R}'$ have different parameters (hence different transitions), the observed state $s'_i(t)$ in $\mathcal{R}'$ does not follow the same distribution as the observed state $s_i(t)$ in $\mathcal{R}$. Thus, to carry out the simulation, we need to record not only the actual observed state $s'_i(t)$, but also a virtual state $s_i(t)$ which follows the same distribution as choosing action i in $\mathcal{R}$. The virtual state $s_i(t)$ and the actual state $s'_i(t)$ are used to update the virtual belief state z and the actual belief state z', respectively. Then, we can pretend to observe the virtual state $s_i(t)$ in our policy $\hat{\pi}^*$ to imitate the trajectory of applying policy π^* in problem $\mathcal{R}$ precisely. Specifically, in our simulated policy $\hat{\pi}^*$, we choose the next action according to the virtual

Algorithm 3 Correspond$(i, \boldsymbol{v}, \boldsymbol{v}', k)$

1: $start = \sum_{j=1}^{k-1} v'_j$, $end = \sum_{j=1}^{k} v'_j$.
2: **for all** j **do**
3: $\quad p_j = \sum_{j'=1}^{j} v_{j'}$
4: $\quad$ **if** $p_j < start$ **then**
5: $\quad\quad q_j = 0$.
6: $\quad$ **else**
7: $\quad\quad q_j = \frac{\min\{p_{j+1}, end\} - p_j}{start - end}$.
8: $\quad$ **end if**
9: **end for**
10: **Return** j with probability q_j.

belief state $z = \{(s_i, \tau_i)\}_{i=1}^N$ instead of the actual belief state $z' = \{(s'_i, \tau'_i)\}_{i=1}^N$. For each time slot t, after we record the virtual state $s_i(t)$, we also construct a corresponding virtual reward $r(i, s_i(t))$, while the actual received reward is $r'(i, s'_i(t))$. Since the virtual belief state z follows the trajectory of applying π^* in $\mathcal{R}$ precisely, the cumulative virtual reward of applying $\hat{\pi}^*$ in $\mathcal{R}'$ is the same as the cumulative reward of applying π^* in $\mathcal{R}$.

We then prove that conditioning on event $\mathcal{E}$, at any time, if we select action i, the observed state $s'_i(t)$ and the virtual state $s_i(t)$ satisfies $s_i(t) \geq s'_i(t)$. We use induction to prove it. At the beginning, we have that $s_i = s'_i$ for any i.

When we choose to pull arm i at time t, suppose that there are τ_i rounds after the last pull of arm i, the last real state of arm i is s'_i, and the last virtual state of arm i is s_i. If our claim holds at time $t-1$, i.e., $s_i \geq s'_i$, then by Lemma 7, we know that $\boldsymbol{e}_{s'_i}(\boldsymbol{P}'_i)^{\tau_i} \gtrsim \boldsymbol{e}_{s_i}(\boldsymbol{P}_i)^{\tau_i}$. Denote $\boldsymbol{v}' = \boldsymbol{e}_{s'_i}(\boldsymbol{P}'_i)^{\tau_i}$ and $\boldsymbol{v} = \boldsymbol{e}_{s_i}(\boldsymbol{P}_i)^{\tau_i}$ as the two input vectors of Correspond (line 5 in Algorithm 2), then the Correspond procedure in Algorithm 3, which is for generating an transition that follows distribution $\boldsymbol{v}$, always returns $j \geq k$ since $p_{k-1} \leq start$ in line 4. Thus, the returned virtual state $s_i(t)$ and the actual observed state $s'_i(t)$ satisfies $s_i(t) \geq s'_i(t)$. Thus we finish the proof of the claim.

Since the virtual state $s_i(t)$ follows the distribution under problem instance $\mathcal{R}$, and the next action only depends on the virtual belief state z, we know that the cumulative virtual reward is the same as the cumulative reward of applying π^* under $\mathcal{R}$.

As for the real reward, we know that conditioning on event $\mathcal{E}$, $r'(i, s'_i(t)) \geq r(i, s'_i(t)) \geq r(i, s_i(t))$. Thus the cumulative real reward of applying $\hat{\pi}^*$ under $\mathcal{R}'$ is larger than the cumulative virtual reward, which equals to the cumulative reward of applying π^* under $\mathcal{R}$. This implies that $\mu(\hat{\pi}^*, \mathcal{R}') \geq \mu(\pi^*, \mathcal{R})$.

On the other hand, since $\pi^{*\prime}$ is the best policy of $\mathcal{R}'$, we must have $\mu(\hat{\pi}^*, \mathcal{R}') \leq \mu(\pi^{*\prime}, \mathcal{R}')$. Thus we finish the proof of $\mu(\pi^*, \mathcal{R}) \leq \mu(\pi^{*\prime}, \mathcal{R}')$. □

A.2 Proof of Lemma 2

In the following, we only consider the case when all actions $i > 0$ are pulled infinitely often. Since a finitely pulled often action $i > 0$ can be replaced by action 0 with only a constant regret, we can safely ignore these actions.

Lemma 8. *Conditioning on event $\mathcal{E}$, we have that $||\boldsymbol{P}_i(k) - \boldsymbol{P}'_i(k)||_\infty \leq 2rad(T)$ and $|r(i,k) - r'(i,k)| \leq 2rad(T)$.*

Proof. Since conditioning on event $\mathcal{E}$ we have that $|\hat{P}_i(j,k) - P_i(j,k)| \leq rad(T)$ and $|\hat{P}_i(j,k) - P'_i(j,k)| \leq rad(T)$, we know that

$$|P'_i(j,k) - P_i(j,k)| \leq |\hat{P}_i(j,k) - P_i(j,k)| + |\hat{P}_i(j,k) - P'_i(j,k)| \leq 2rad(T).$$

Similarly we can prove that $|r(i,k) - r'(i,k)| \leq 2rad(T)$. □

Lemma 9. *Conditioning on event $\mathcal{E}$, for any τ, i, k and probability vector $\boldsymbol{v}$, we have that $||\boldsymbol{v}(\boldsymbol{P}_i)^\tau - \boldsymbol{v}(\boldsymbol{P}'_i)^\tau||_\infty \leq 2\tau \cdot rad(T)$.*

Proof. Note that

$$\begin{aligned} ||\boldsymbol{v}(\boldsymbol{P}_i)^\tau - \boldsymbol{v}(\boldsymbol{P}_i')^\tau||_\infty &\leq \sum_{\tau'=0}^{\tau-1} ||\boldsymbol{v}(\boldsymbol{P}_i')^{\tau'}(\boldsymbol{P}_i)^{\tau-\tau'-1}(\boldsymbol{P}_i - \boldsymbol{P}_i')||_\infty \\ &= \sum_{\tau'=0}^{\tau-1} ||\boldsymbol{v}(\tau')\boldsymbol{P}_i - \boldsymbol{v}(\tau')\boldsymbol{P}_i'||_\infty \\ &\leq \tau \cdot (2rad(T)), \end{aligned} \tag{3}$$

where $\boldsymbol{v}(\tau') = \boldsymbol{v}(\boldsymbol{P}_i')^{\tau'}(\boldsymbol{P}_i)^{\tau-\tau'-1}$, and Eq. (3) is because that by Lemma 8 we always have $||\boldsymbol{v}(\tau')\boldsymbol{P}_i - \boldsymbol{v}(\tau')\boldsymbol{P}_i'||_\infty \leq 2rad(T)$. □

Lemma 10. *Under Assumptions 2, 4 and conditioning on event $\mathcal{E}$, for any $\tau > \frac{\log T}{\log(1/\lambda_{\max})}, i, k$ and probability vector $\boldsymbol{v}$, we have that $||\boldsymbol{v}(\boldsymbol{P}_i)^\tau - \boldsymbol{v}(\boldsymbol{P}_i')^\tau||_\infty \leq \frac{2M}{1-\lambda_{\max}} \cdot rad(T) + \frac{2M}{T}$.*

Proof. Since under Assumptions 2, 4, both M_i (with transition matrix $\boldsymbol{P}_i$) and M_i' (with transition matrix $\boldsymbol{P}_i'$) are ergodic and have unique stationary distributions. After $\frac{\log T}{\log(1/\lambda_{\max})}$ time steps, both the two Markov chains converge to their stationary distributions. By results in [39, 12], conditioning on event $\mathcal{E}$, their stationary distributions satisfy that $\lim_{\tau\to\infty} ||\boldsymbol{v}(\boldsymbol{P}_i)^\tau - \boldsymbol{v}(\boldsymbol{P}_i')^\tau||_\infty \leq \frac{M}{1-\lambda_{\max}} \cdot 2rad(T)$. On the other hand, after $\frac{\log T}{\log(1/\lambda_{\max})}$ steps, the gap between $\boldsymbol{v}(\boldsymbol{P}_i)^\tau$ and the stationary distribution is upper bounded by $\frac{M}{T}$ (similar for $\boldsymbol{v}(\boldsymbol{P}_i')^\tau$) [38].

Thus $||\boldsymbol{v}(\boldsymbol{P}_i)^\tau - \boldsymbol{v}(\boldsymbol{P}_i')^\tau||_\infty \leq \frac{2M}{1-\lambda_{\max}} \cdot rad(T) + \frac{2M}{T}$ for any $\tau > \frac{\log T}{\log(1/\lambda_{\max})}, i, k$, and probability vector $\boldsymbol{v}$. □

Lemma 11. *Under Assumptions 2, 4 and conditioning on event $\mathcal{E}$, for any τ, i, k and probability vector v, we have that $||\boldsymbol{v}(\boldsymbol{P}_i)^\tau - \boldsymbol{v}(\boldsymbol{P}_i')^\tau||_\infty \leq \left(\frac{2\log T}{\log(1/\lambda_{\max})} + \frac{2M}{1-\lambda_{\max}}\right) \cdot rad(T) + \frac{2M}{T}$.*

Proof. This lemma is given by directly applying Lemmas 9 and 10. □

Based on the results in Lemma 11, we denote $gap(T) = \left(\frac{2\log T}{\log(1/\lambda_{\max})} + \frac{2M}{1-\lambda_{\max}}\right) \cdot rad(T) + \frac{2M}{T}$, which will be used frequently in the following analysis.

Lemma 12. *Under Assumptions 2, 4 and conditioning on event $\mathcal{E}$, we have that $r(z) - r'(z) \leq 2rad(T) + M \cdot gap(T)$, where $z = \{(\tau_i, s_i)\}_{i=1}^N$ is the belief state of the game, $r(z)$ and $r'(z)$ is the expected given reward of belief state z in problem $\mathcal{R}$ and $\mathcal{R}'$ respectively.*

Proof. Note that $r(z) = \sum_{k=1}^m r(i,k)v_k$ and $r'(z) = \sum_{k=1}^m r'(i,k)v_k'$, where i is the chosen action at belief state $z = \{(\tau_i, s_i)\}_{i=1}^N$, and $\boldsymbol{v} = \boldsymbol{e}_{s_i}(\boldsymbol{P}_i)^{\tau_i}$, $\boldsymbol{v}' = \boldsymbol{e}_{s_i}(\boldsymbol{P}_i')^{\tau_i}$.

Thus

$$\begin{aligned} |r(z) - r'(z)| &= |\sum_{k=1}^M r(i,k)v_k - \sum_{k=1}^M r'(i,k)v_k'| \\ &= |\sum_{k=1}^M (r(i,k) - r'(i,k))v_k + \sum_{k=1}^M r'(i,k)(v_k - v_k')| \\ &\leq |\sum_{k=1}^M (r(i,k) - r'(i,k))v_k| + |\sum_{k=1}^M r'(i,k)(v_k - v_k')| \\ &\leq 2rad(T) + M \cdot gap(T). \end{aligned} \tag{4}$$

where Eq. (4) is because that $|r(i,k) - r'(i,k)| \leq 2rad(T)$ (Lemma 8) and $|v_k - v_k'| \leq gap(T)$ (Lemma 11). □

Let $V_t(z)$ denote the expected cumulative reward that we start at belief state $z = \{\tau_i, s_i\}_{i=1}^N$ and implement policy $\pi^{*\prime}$ for t time steps in $\mathcal{R}$, and $\boldsymbol{V}_t$ denote the vector of $V_t(z)$. Similarly, let $V_t'(z)$ be the expected cumulative reward that we start at belief state z and implement $\pi^{*\prime}$ for t times in $\mathcal{R}'$ and $\boldsymbol{V}_t'$ be the vector of $V_t'(z)$. Then we know that $\mu(\pi^{*\prime}, \mathcal{R}') - \mu(\pi^{*\prime}, \mathcal{R}) \leq \lim_{t\to\infty} \frac{1}{t}||\boldsymbol{V}_t - \boldsymbol{V}_t'||_\infty$.

Notice that under the fixed policy $\pi^{*\prime}$, $\mathcal{R}$ and $\mathcal{R}'$ are two Markov Chains. Let $\mathcal{T}$ and $\mathcal{T}'$ be the transition matrices of these two Markov Chains, and let $\boldsymbol{r}$ and $\boldsymbol{r}'$ to be the reward vectors of them, respectively. Then, $\boldsymbol{V}_t = \sum_{\tau=0}^{t-1} \mathcal{T}^\tau \boldsymbol{r}$, and $\boldsymbol{V}_t' = \sum_{\tau=0}^{t-1} (\mathcal{T}')^\tau \boldsymbol{r}'$, which implies that

$$\begin{aligned}\boldsymbol{V}_{t+1} - \boldsymbol{V}_{t+1}' &= (\mathcal{T}\boldsymbol{V}_t + \boldsymbol{r}) - (\mathcal{T}'\boldsymbol{V}_t' + \boldsymbol{r}') \\ &= \mathcal{T}\boldsymbol{V}_t - \mathcal{T}'\boldsymbol{V}_t' + (\boldsymbol{r} - \boldsymbol{r}') \\ &= \mathcal{T}(\boldsymbol{V}_t - \boldsymbol{V}_t') + (\mathcal{T} - \mathcal{T}')\boldsymbol{V}_t' + (\boldsymbol{r} - \boldsymbol{r}').\end{aligned}$$

Since $\mathcal{T}$ represents a transition matrix, $||\mathcal{T}(\boldsymbol{V}_t - \boldsymbol{V}_t')||_\infty \leq ||\boldsymbol{V}_t - \boldsymbol{V}_t'||_\infty$. This implies that

$$\frac{1}{t}||\boldsymbol{V}_t - \boldsymbol{V}_t'||_\infty \leq \frac{1}{t}\sum_{\tau=0}^{t-1} ||(\mathcal{T} - \mathcal{T}')\boldsymbol{V}_\tau'||_\infty + \frac{1}{t}\sum_{\tau=0}^{t-1} ||\boldsymbol{r} - \boldsymbol{r}'||_\infty. \tag{5}$$

Lemma 12 shows that $\frac{1}{t}\sum_{\tau=0}^{t-1} ||\boldsymbol{r} - \boldsymbol{r}'||_\infty \leq 2rad(T) + M \cdot gap(T)$.

As for the first term in Eq. (5), i.e., $||(\mathcal{T} - \mathcal{T}')\boldsymbol{V}_\tau'||_\infty$, notice that for any belief state $z = \{(s_i, \tau_i)\}_{i=1}^N$ that we select action $i \neq 0$, there are only M non-zero values in $\mathcal{T}$ and $\mathcal{T}'$, i.e., transitions to belief states $z_1', \cdots, z_M'$, where z_k' is given by substitute (s_i, τ_i) by $(k, 1)$, and other actions $i' \neq i$ will add 1 to their $\tau_{i'}$'s. Thus, the z-th term in $(\mathcal{T} - \mathcal{T}')\boldsymbol{V}_\tau'$ equals to $\sum_{k=1}^m (v_k(z) - v_k'(z))V_\tau'(z_k')$, where $\boldsymbol{v}(z)$ and $\boldsymbol{v}'(z)$ are probability distributions of the next observed state in $\mathcal{R}$ and $\mathcal{R}'$ under the belief state z. Under the event $\mathcal{E}$, $v_k(z) - v_k'(z)$ can be bounded by $\tilde{O}(\frac{M}{1-\lambda_{\max}} rad(T))$ (Lemma 11). On the other hand, since $\sum_{k=1}^M v_k(z) - v_k'(z) = 0$, we have that $\sum_{k=1}^m (v_k(z) - v_k'(z))V_\tau'(z_k') \leq \tilde{O}(\frac{M}{1-\lambda_{\max}} rad(T)) \cdot M \max_{j>k} |V_\tau'(z_j') - V_\tau'(z_k')|$. Therefore, the remaining issue is to bound $\max_{j>k} |V_\tau'(z_j') - V_\tau'(z_k')|$ for any belief state z.

Thus, in the following, we concentrate on a fixed tuple (z, j, k), and use the idea of simulated policy again. The simulated policy denoted by ${\pi^*_{z,j,k}}'$ is shown in Algorithm 4. Specifically, we pretend to observe a virtual state $s_i = k$ at the beginning, while the actual observed state is $s_i' = j$. Similar as Algorithm 2, in every time step, we observe the real state $s'_{a(t)}(t)$ and real reward $r'(a(t), s'_{a(t)}(t))$, but we also record a virtual state $s_{a(t)}(t)$ and a virtual reward $r'(a(t), s_{a(t)}(t))$, and the next action only depends on the virtual belief state, but not the real belief state. We use $V_\tau^{z,j,k\prime}(z_j')$ to denote the expected cumulative reward of applying ${\pi^*_{z,j,k}}'$ in $\mathcal{R}'$ starting at belief state z_j' (the cumulative real reward). Similarly as before, in ${\pi^*_{z,j,k}}'$, $V_\tau'(z_k')$ equals to the cumulative virtual reward.

Then we can bound the gap between $V_\tau^{z,j,k\prime}(z_j')$ and $V_\tau'(z_k')$ in the next lemma.

Lemma 13. *Under Assumptions 2, 4 and conditioning on event $\mathcal{E}$,*

$$|V_\tau^{z,j,k\prime}(z_j') - V_\tau'(z_k')| \leq \frac{2M}{1 - \lambda_{\max}}.$$

Proof. Recall that $V_\tau^{z,j,k\prime}(z_j')$ is the expected cumulative reward of applying ${\pi^*_{z,j,k}}'$ (Algorithm 4) in $\mathcal{R}'$ and start at belief state z_j'. Similar with the previous analysis, in Algorithm 4, the cumulative virtual reward is $V_\tau'(z_k)$.

If we do not pull arm i (the arm chosen in belief state z) at time t, then the real reward equals to the virtual reward in this time step, since they are under the same problem instance $\mathcal{R}'$. Thus the difference between real reward and virtual reward only appears when we pull arm i.

At the beginning, the probability distribution of s_i (the state of arm i in belief state y) is $\boldsymbol{e}_k$ and the probability distribution of s_i' (the state of arm i in belief state y') is $\boldsymbol{e}_j$. After τ_i time steps, the probability distribution of the next virtual state of arm i becomes $\boldsymbol{e}_k(\boldsymbol{P}_i')^{\tau_i}$ and the probability distribution of next real state of arm i becomes $\boldsymbol{e}_j(\boldsymbol{P}_i')^{\tau_i}$. Denote $\boldsymbol{v}(\tau_i) = \boldsymbol{e}_k(\boldsymbol{P}_i')^{\tau_i}$ and $\boldsymbol{v}'(\tau_i) =$

Algorithm 4 Simulated ${\pi^*_{z,j,k}}'$ start at state z'_j based on ${\pi^*}'$

Init: The real belief state $y' = z'_j$, and set the virtual belief state $y = z'_k$, let i be the chosen action under z.
while true do
  Choose action $a(t)$ as ${\pi^*}'$ choose under y, and observes state $s'_{a(t)}(t)$, reward $r'(a(t), s'(t))$.
  if $a(t) \neq i$ **then**
    Update the $a(t)$-th term in y and y' to be $(s'_{a(t)}(t), 1)$. For other terms $j \neq a(t)$, set $\tau_j = \tau_j + 1$ and $\tau'_j = \tau'_j + 1$. Observes virtual reward $r'(a(t), s'_{a(t)}(t))$.
  else
    Set $\boldsymbol{v}' = \boldsymbol{e}_{s'_{a(t)}}(\boldsymbol{P}'_{a(t)})^{\tau_{a(t)}}$ and $\boldsymbol{v} = \boldsymbol{e}_{s_{a(t)}}(\boldsymbol{P}'_{a(t)})^{\tau_{a(t)}}$.
    Update the $a(t)$-th term in z to be $(s_{a(t)}(t), 1)$, where $s_{a(t)}(t) = \mathsf{Correspond}(a(t), \boldsymbol{v}, \boldsymbol{v}', s'_{a(t)}(t))$. For other terms $j \neq a(t)$, set $\tau_j = \tau_j + 1$. Observes virtual reward $r'(a(t), s_{a(t)}(t))$.
    Update the $a(t)$-th term in z' to be $(s'_{a(t)}(t), 1)$, for other terms $j \neq a(t)$, set $\tau_j = \tau_j + 1$.
  end if
end while

$\boldsymbol{e}_j(\boldsymbol{P}'_i)^{\tau_i}$ Then we can upper bound the expected gap between real reward and virtual reward at time τ_i as $\sum_{\ell=1}^{M} |v_\ell(\tau_i) - v'_\ell(\tau_i)| r'(i,\ell) \leq ||\boldsymbol{v}(\tau_i) - \boldsymbol{v}'(\tau_i)||_1$.

Under Assumptions 2, 4, $\boldsymbol{V} = diag(1, \sqrt{\frac{P'_i(2,1)}{P'_i(1,2)}}, \sqrt{\frac{P'_i(2,1)P'_i(3,2)}{P'_i(1,2)P'_i(2,3)}}, \cdots, \sqrt{\prod_{\ell=1}^{M-1} \frac{P'_i(\ell+1,\ell)}{P'_i(\ell,\ell+1)}})$ satisfies that $\boldsymbol{V}^{-1}\boldsymbol{P}'_i\boldsymbol{V}$ is a symmetric matrix. Thus, the maximum Jordan block size of the Jordan normal form of $\boldsymbol{P}'_i$ equals to 1. According to Fact 3 in [38], such $\boldsymbol{P}'_i$ satisfies that the value of $||\boldsymbol{v}(\tau_i) - \boldsymbol{d}^{(i)'}||_1$ converges to 0 exponentially with rate at most $\lambda^{i'}$, where $\boldsymbol{d}^{(i)'}$ is the unique stationary distribution vector of M_i in $\mathcal{R}'$, and $\lambda^{i'}$ is the absolute value of the second largest eigenvalue of $\boldsymbol{P}'_i$.

Specifically, we have that:

$$\begin{aligned} ||\boldsymbol{v}(\tau_i) - \boldsymbol{d}^{(i)'}||_1 &\leq M(\lambda^{i'})^{\tau_i} ||\boldsymbol{v}(0) - \boldsymbol{d}^{(i)'}||_1 \\ &\leq 2M(\lambda_{\max})^{\tau_i}. \end{aligned}$$

Thus, the $|V_\tau^{z,j,k'}(z'_j) - V'_\tau(z'_k)|$ is upper bounded by $2M \sum_{\tau_i=0}^{\infty} (\lambda_{\max})^{\tau_i} \leq \frac{2M}{1-\lambda_{\max}}$. □

Lemma 14. *Under Assumptions 2, 4, if all the arms (expected for the default one) are pulled infinitely often, and applying ${\pi^*}'$ on $\mathcal{R}'$ is aperiodic, then the stationary distribution of applying ${\pi^*_{z,j,k}}'$ on $\mathcal{R}'$ and applying ${\pi^*}'$ on $\mathcal{R}'$ is the same.*

Proof. Note that when applying policy ${\pi^*}'$ (or ${\pi^*_{z,j,k}}'$), the arm i (the arm chosen by ${\pi^*}'$ under belief state z) is pulled infinitely often. Thus, we know that for any $T > 0$, there must be a pull of arm i after T.

On the other hand, Under Assumptions 2, 4, we know that the Markov Chain M'_i (with transition matrix $\boldsymbol{P}'_i$) exponentially converges to its unique stationary distribution. Thus the probability of $s_i(t) = s'_i(t)$ converges to 1 as time goes to infinity. Once $s_i = s'_i$, we know that after this time slot we always have $y = y'$, i.e., the virtual belief state equals to the real one. Thus, ${\pi^*_{z,j,k}}'$ and ${\pi^*}'$ are the same policy after this time step. This means that they will converge to the same stationary distribution. □

Lemma 15. *Under Assumptions 2, 4, if all the arms (expected for the default one) are pulled infinitely often, and applying ${\pi^*}'$ on $\mathcal{R}'$ is aperiodic, then*

$$\lim_{n\to\infty} (V_t^{z,j,k'}(z'_j) - V'_t(z'_j)) \leq 0. \tag{6}$$

Proof. Let's consider the Bellman equations of applying ${\pi^*}'$ on $\mathcal{R}'$. Denote $\mu^{*'} = \mu({\pi^*}', \mathcal{R}')$, and $Q(z)$ the Q-value of belief state z.

Then by Bellman equations, we have that:

$$\begin{aligned} Q(z'_j) &= \mathbb{E}[r_1(z'_j)] - \mu^{*\prime} + \mathbb{E}[Q(z_1(z'_j))] \\ &= \mathbb{E}[r_1(z'_j)] + \mathbb{E}[r_2(z'_j)] - 2\mu^{*\prime} + \mathbb{E}[Q(z_2(z'_j))] \\ &= \cdots \\ &= \sum_{\tau=1}^{t} \mathbb{E}[r_\tau(z'_j)] - t\mu^{*\prime} + \mathbb{E}[Q(z_t(z'_j))] \\ &= V'_t(z'_j) + \mathbb{E}[Q(z_t(z'_j))] - t\mu^{*\prime}, \end{aligned}$$

where $r_\tau(z'_j)$ and $z_\tau(z'_j)$ are the random reward and belief state in the τ-th time step of applying $\pi^{*\prime}$ in $\mathcal{R}'$ that starts at z'_j, respectively.

Now let's consider the policy ${\pi^*_{z,j,k}}'$, since it is not the best policy, we have that:

$$\begin{aligned} Q(z'_j) &\geq \mathbb{E}[r'_1(z'_j)] - \mu^{*\prime} + \mathbb{E}[Q(z'_1(z'_j))] \\ &\geq \mathbb{E}[r'_1(z'_j)] + \mathbb{E}[r'_2(z'_j)] - 2\mu^{*\prime} + \mathbb{E}[Q(z'_2(z'_j))] \\ &\geq \cdots \\ &\geq \sum_{\tau=1}^{t} \mathbb{E}[r'_\tau(z'_j)] - t\mu^{*\prime} + \mathbb{E}[Q(z'_t(z'_j))] \\ &= V_t^{z,j,k\prime}(z'_j) + \mathbb{E}[Q(z'_t(z'_j))] - t\mu^{*\prime} \end{aligned}$$

where $r'_\tau(z'_j)$ and $z'_\tau(z'_j)$ are the random reward and belief state in the τ-th time step of applying ${\pi^*_{z,j,k}}'$ in $\mathcal{R}'$ that starts at z'_j, respectively. The reason that here is greater than or equal to is because that when applying ${\pi^*_{z,j,k}}'$, sometimes we do not choose the best action.

Then we have that

$$V'_t(z'_j) + \mathbb{E}[Q(z_t(z_j))] - t\mu^{*\prime} \geq V_t^{z,j,k\prime}(z'_j) + \mathbb{E}[Q(z'_t(z_j))] - t\mu^{*\prime}.$$

When $t \to \infty$, by Lemma 14, we have that $\mathbb{E}[Q(z_t(z_j))] = \mathbb{E}[Q(z'_t(z_j))]$, which implies that $\lim_{t\to\infty}(V_t^{z,j,k\prime}(z'_j) - V'_t(z'_j)) \leq 0$. □

Lemma 16. *Under Assumptions 2, 4 and conditioning on event $\mathcal{E}$, if all the actions $i > 0$ are pulled infinitely often, then we have that*

$$\lim_{\tau\to\infty} \max_{j>k} |V'_\tau(z'_j) - V'_\tau(z'_k)| \leq \frac{2M}{1-\lambda_{\max}}. \tag{7}$$

Proof. If applying $\pi^{*\prime}$ on $\mathcal{R}'$ is aperiodic, then we can directly apply Lemmas 13 and 15 to get that for any $j > k$, $\lim_{\tau\to\infty} V'_\tau(z'_k) - V'_\tau(z'_j) \leq \frac{2M}{1-\lambda_{\max}}$. Similarly, we can prove that $\lim_{\tau\to\infty} V'_\tau(z'_j) - V'_\tau(z'_k) \leq \frac{2M}{1-\lambda_{\max}}$, which finish the proof in this case.

Then we consider the case that applying $\pi^{*\prime}$ on $\mathcal{R}'$ has a constant period larger than 1. Note that in proof of Lemma 15, we only require that $z'_t(z'_j)$ and $z_t(z'_j)$ has the same distribution when $t \to \infty$. On the other hand, $z'_t(z'_j)$ and $z_t(z'_j)$ start from the same state z'_j. Thus they are in the same set of states during one period. Because of this, even if applying $\pi^{*\prime}$ on $\mathcal{R}'$ has a constant period larger than 1, $z_t(z'_j)$ and $z'_t(z'_j)$ converges to the same distribution when $n \to \infty$. This implies that the result in Lemma 15 is still correct. Along with Lemma 13, we know that $\lim_{\tau\to\infty} \max_{j>k} |V'_\tau(z'_k) - V'_\tau(z'_j)| \leq \frac{2M}{1-\lambda_{\max}}$ still holds. □

Based on these lemmas, we propose the proof of Lemma 2 here.

Lemma 2. *Conditioning on event $\mathcal{E}$, $\mu(\pi^{*\prime}, \mathcal{R}') - \mu(\pi^{*\prime}, \mathcal{R}) = \tilde{O}\left(\frac{M^3}{(1-\lambda_{\max})^2} T^{-\frac{1}{3}}\right)$.*

Proof. Eq. (5) shows that $\mu(\pi^{*\prime}, \mathcal{R}') - \mu(\pi^{*\prime}, \mathcal{R}) \leq \lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} ||(\mathcal{T} - \mathcal{T}')\boldsymbol{V}'_\tau||_\infty + \lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} ||\boldsymbol{r} - \boldsymbol{r}'||_\infty$.

Lemma 12 shows that $\lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} ||\boldsymbol{r} - \boldsymbol{r}'||_\infty \leq 2rad(T) + M \cdot gap(T)$.

As for $||(\mathcal{T} - \mathcal{T}')\boldsymbol{V}'_\tau||_\infty$, note that for any belief state $z = \{(s_i, \tau_i)\}_{i=1}^N$ that we select action $i \neq 0$, there are only M non-zero values in $\mathcal{T}$ and $\mathcal{T}'$, i.e., transitions to belief states $z'_1, \cdots, z'_M$, where z'_k is given by substituting (s_i, τ_i) by $(k, 1)$, and other actions $j \neq i$ will increase their τ_j values by 1. Thus, the z-th term in $(\mathcal{T} - \mathcal{T}')\boldsymbol{V}'_\tau$ equals to $\sum_{k=1}^m (v_k(z) - v'_k(z))V'_\tau(z'_k)$, where $\boldsymbol{v}(z)$ and $\boldsymbol{v}'(z)$ are probability distributions of the next observed state in $\mathcal{R}$ and $\mathcal{R}'$ under the belief state z. That is,

$$\begin{aligned}
\lim_{t\to\infty} ||(\mathcal{T} - \mathcal{T}')\boldsymbol{V}'_\tau||_\infty &= \lim_{t\to\infty} \sum_{k=1}^m (v_k(z) - v'_k(z))V'_\tau(z'_k) \\
&\leq \lim_{t\to\infty} \frac{M}{2} ||\boldsymbol{v}(z) - \boldsymbol{v}'(z)||_\infty \max_{j>k} |V'_\tau(z'_j) - V'_\tau(z'_k)| \qquad (8) \\
&\leq \frac{M}{2} gap(T) \lim_{t\to\infty} \max_{j>k} |V'_\tau(z'_j) - V'_\tau(z'_k)| \\
&\leq \frac{M^2}{1-\lambda_{\max}} gap(T), \qquad (9)
\end{aligned}$$

where Eq. (8) is because that $\boldsymbol{v}(z)$ and $\boldsymbol{v}'(z)$ are probability vectors with dimension M, and Eq. 9 comes from Lemma 16.

Thus, $\lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1} ||(\mathcal{T} - \mathcal{T}')\boldsymbol{V}'_\tau||_\infty \leq \frac{M^2}{1-\lambda_{\max}} gap(T)$. Since $gap(T) = \tilde{O}\left(\frac{M}{1-\lambda_{\max}} rad(T)\right)$, we have that $\mu(\pi^{*\prime}, \mathcal{R}') - \mu(\pi^{*\prime}, \mathcal{R}) \leq \tilde{O}\left(\frac{M^3}{(1-\lambda_{\max})^2} rad(T)\right) + 2rad(T) + \tilde{O}\left(\frac{M^2}{1-\lambda_{\max}} rad(T)\right) = \tilde{O}\left(\frac{M^3}{(1-\lambda_{\max})^2} rad(T)\right)$.

Note that when $m(T) = T^{\frac{2}{3}}$, we have that $rad(T) = \tilde{O}(T^{-\frac{1}{3}})$, therefore $\mu(\pi^{*\prime}, \mathcal{R}') - \mu(\pi^{*\prime}, \mathcal{R})$ is upper bounded by $\tilde{O}\left(\frac{M^3}{(1-\lambda_{\max})^2} T^{-\frac{1}{3}}\right)$. □

A.3 Other Lemmas and Facts

Fact 1. *The length T_1 of the exploration phase satisfies that*

$$\mathbb{E}[T_1] = \tilde{O}\left(\frac{N}{d_{\min}} m(T)\right).$$

Lemma 17. *With probability at least $1 - \frac{8NM}{T}$, $\mathcal{E}$ holds.*

Proof. Recall that

$$\mathcal{E} = \{\forall i, |j-k| \leq 1, |P_i(j,k) - \hat{P}_i(j,k)| \leq rad(T), |r(i,k) - \hat{r}(i,k)| \leq rad(T)\}.$$

For any action i and state j, k, we have that

$$\begin{aligned}
\Pr[|P_i(j,k) - \hat{P}_i(j,k)| \geq rad(T)] &\leq 2\exp(-2m(T)(rad(T))^2) \qquad (10) \\
&\leq 2\exp\left(-2m(T) \cdot \frac{\log T}{2m(T)}\right) \\
&\leq \frac{2}{T},
\end{aligned}$$

where Eq. (10) is given by Chernoff-Hoeffding inequality [19]. Similarly, $\Pr[|r(i,k) - \hat{r}(i,k)| \leq rad(T)] \leq \frac{2}{T}$.

Thus, by union bound, $\mathcal{E}$ holds with probability at least $1 - \frac{8NM}{T}$. □

A.4 Main Proof of Theorem 1

A.4.1 Regret in Exploration Phase

By Fact 1, we know the regret in exploration phase has upper bound $\tilde{O}(\frac{N}{d_{\min}}m(T))$.

A.4.2 Regret in Exploitation Phase

Let T_2 be the number of time steps in the exploitation phase. Note that applying $\pi^{*\prime}$ in $\mathcal{R}$ has an average reward $\mu(\pi^{*\prime}, \mathcal{R})$. According to results in [6], the cumulative reward of applying policy $\pi^{*\prime}$ in $\mathcal{R}$ for T_2 time steps is lower bounded by $T_2\mu(\pi^{*\prime}, \mathcal{R}) - \mathcal{C}$, where $\mathcal{C}$ is the diameter of applying policy $\pi^{*\prime}$ in $\mathcal{R}$, which does not depend on T.

Then we can write the upper bound of cumulative regret in exploitation phase as

$$T_2(\mu(\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R})) + \mathcal{C}. \tag{11}$$

Since $\mathcal{C}$ is a constant that does not depend on T, we can concentrate on the term depends on T (or T_2), i.e., $T_2(\mu(\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R}))$. To bound this term, we write $\mu(\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R})$ as:

$$\mu[\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R}) \quad = \quad [\mu(\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R}')] + [\mu(\pi^{*\prime}, \mathcal{R}') - \mu(\pi^{*\prime}, \mathcal{R})]. \tag{12}$$

By Lemma 1, conditioning on event $\mathcal{E}$, the first term is upper bounded by 0.

By Lemma 2, conditioning on event $\mathcal{E}$, the second term is upper bounded by $\tilde{O}\left(\frac{M^3}{(1-\lambda_{\max})^2}T^{-\frac{1}{3}}\right)$.

Therefore, conditioning on event $\mathcal{E}$, we have that

$$T_2(\mu(\pi^*, \mathcal{R}) - \mu(\pi^{*\prime}, \mathcal{R})) \leq \tilde{O}\left(\frac{M^3}{(1-\lambda_{\max})^2}T^{\frac{2}{3}}\right)$$

A.4.3 The Total Regret

From the above analysis, the total regret is upper bounded by:

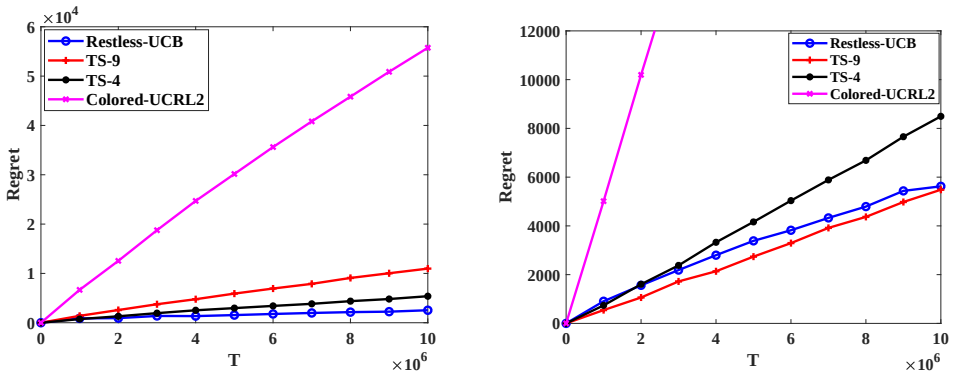
$$\begin{aligned} Reg(T) &\leq \mathbb{E}[T_1] + \tilde{O}\left(\frac{M^3}{(1-\lambda_{\max})^2}T^{\frac{2}{3}}\right) + 8NM + \mathcal{C} \\ &\leq \tilde{O}\left(\frac{N}{d_{\min}}m(T)\right) + \tilde{O}\left(\frac{M^3}{(1-\lambda_{\max})^2}T^{\frac{2}{3}}\right) + 8NM + \mathcal{C} \\ &= \tilde{O}\left(\left(\frac{N}{d_{\min}} + \frac{M^3}{(1-\lambda_{\max})^2}\right)T^{\frac{2}{3}}\right). \end{aligned}$$

B Reduced Regret Bound for Colored-UCRL2 or Thompson Sampling

Denote $\boldsymbol{h}'$ the bias vector of applying policy $\pi^{*\prime}$ in $\mathcal{R}'$, and $U = \max_{z,j>k} |h'(z'_j) - h'(z'_k)|$. The colored-UCRL2 policy [33] and Thompson Sampling policy [22] both achieve regret upper bounds of $O(U\sqrt{T})$, according to their analysis. To bound the value of U, they both directly apply an upper bound D, which is the diameter of applying policy $\pi^{*\prime}$ in $\mathcal{R}'$, according to [6].

However, note that $|h'(z'_j) - h'(z'_k)| = \lim_{t\to\infty} |V'_t(z'_j) - V'_t(z'_k)|$, and Lemma 16 states that $\lim_{t\to\infty} \max_{j>k} |V'_t(z'_j) - V'_t(z'_k)| \leq \frac{2M}{1-\lambda_{\max}}$ for any z. Therefore,

$$\begin{aligned} U &= \max_{z,j>k} |h'(z'_j) - h'(z'_k)| \\ &= \max_{z,j>k} \lim_{t\to\infty} |V'_t(z'_j) - V'_t(z'_k)| \\ &= \lim_{t\to\infty} \max_{z,j>k} |V'_t(z'_j) - V'_t(z'_k)| \\ &\leq \frac{2M}{1-\lambda_{\max}}. \end{aligned}$$



(a) Regret: two channels

(b) Regret: three channels

Figure 2: More experiments: Comparison of regrets of different algorithms

This implies that the regret upper bounds in [33, 22] can be reduced to $O(\frac{M}{1-\lambda_{\max}}\sqrt{T})$, whose constant factor is a polynomial one.

More importantly, in the proof of Lemma 16, we only use Assumptions 2, 4 to make sure that the both the original Markov chain M_i (with transition matrix $\boldsymbol{P}_i$) and the constructed Markov chain M_i' (with transition matrix $\boldsymbol{P}_i'$) are ergodic. Therefore, Lemma 16 is not limited to the setting in this paper, but instead can be applied to more general settings and reduce the exponential factors in the regret upper bounds under other learning policies.

C Approximation Oracle

Theorem 2. *If* `Oracle` *returns an approximate policy with ratio λ, then under Assumptions 1, 2, 3 and 4, the approximation regret (with approximate ratio λ) of Restless-UCB with $m(T) = T^{\frac{2}{3}}$ in an online restless bandit problem $\mathcal{R}$ is upper bounded by $\tilde{O}(T^{\frac{2}{3}})$.*

Proof. First, we see that the exploration phase still results in $\tilde{O}(T^{\frac{2}{3}})$ regret. Next, we come to the exploitation phase.

Similarly, the regret in the exploitation phase can be upper bounded by $T_2(\lambda\mu(\pi^*, \mathcal{R}) - \mu(\tilde{\pi}', \mathcal{R}))$. We can similarly write $\lambda\mu(\pi^*, \mathcal{R}) - \mu(\tilde{\pi}', \mathcal{R})$ as

$$\lambda\mu(\pi^*, \mathcal{R}) - \mu(\tilde{\pi}', \mathcal{R}) = [\lambda\mu(\pi^*, \mathcal{R}) - \mu(\tilde{\pi}', \mathcal{R}')] + [\mu(\tilde{\pi}', \mathcal{R}') - \mu(\tilde{\pi}', \mathcal{R})]$$

Note that Lemma 1 still works in this case. Thus, we must have $\mu(\tilde{\pi}', \mathcal{R}') \geq \lambda\mu(\pi^{*\prime}, \mathcal{R}') \geq \lambda\mu(\pi^*, \mathcal{R})$ with high probability. However, Lemma 16 does not work here since it relies on the optimality of $\pi^{*\prime}$.

According to the results in [6], we can use D, the diameter of applying $\tilde{\pi}'$ in the problem $\mathcal{R}'$ as an upper bound for $\lim_{\tau\to\infty} \max_{j>k} |V_\tau'(z_k') - V_\tau'(z_j')|$. Thus, the regret in exploitation phase is still $\tilde{O}(T^{\frac{2}{3}})$ while the constant factor here is much larger and probably exponential.

Together with the regret in exploration phase, we finish the proof. □

Note that although the constant factor in the regret bound can be exponential, the algorithm complexity is still polynomial and much better than the colored-UCRL2 policy [33].

D More Experiments on Real Datasets

We also use the dataset in [40] for packet transmission via a wireless link with different frequency channels. Table 3 of [40] provides the transition matrices of the two-state Markov Chains for different

channels. In this setting, a bad state results in a packet loss while a good state guarantees a successful transition. In Figure 2(a), one can use frequency bands of 551MHz and 665MHz. In Figure 2(b), one can use frequency bands of 551MHz, 629MHz and 665MHz.

One can see that Restless-UCB still outperforms other policies. The TS policy suffers from a linear regret, since its support does not contain the real transition matrices, and colored-UCRL2 performs worse than Restless-UCB as well. These results also demonstrate the effectiveness of Restless-UCB.