Stochastic Multi-armed Bandits in Constant Space

David Liau davidliau@utexas.edu
Eric Price ecprice@cs.utexas.edu
Zhao Song zhaos@utexas.edu
Ger Yang geryang@utexas.edu
The University of Texas at Austin

Abstract

We consider the stochastic bandit problem in the sublinear space setting, where one cannot record the win-loss record for all K arms. We give an algorithm using $O(1)$ words of space with regret

$$\sum_{i=1}^{K} \frac{1}{\Delta_i} \log \frac{\Delta_i}{\Delta} \log T$$

where Δ_i is the gap between the best arm and arm i and Δ is the gap between the best and the second-best arms. If the rewards are bounded away from 0 and 1, this is within an $O(\log 1/\Delta)$ factor of the optimum regret possible without space constraints.

1 Introduction

In this paper, we study the multi-arm bandit problem in a sublinear space setting. In an instance of the bandit problem, there are K arms and a finite time horizon $1, \ldots, T$, where T could be unknown to us. At each time step, we pull one of the K arms, and receive a reward that depends on our choice. The goal is to find a strategy that would achieve a sublinear (with respect to time) *regret*, which is defined as the difference between the cumulative reward we received from our strategy and the reward we could have received if we always pulled the best arm in the hindsight.

There are many formulations of the bandit problem. In this paper we consider the stochastic setting specifically. In the stochastic setting, one assumes the rewards from the i-th arm are i.i.d. random variables, with mean μ_i and support $[0, 1]$. A well-known algorithm for the stochastic bandit is the UCB algorithm (Auer et al., 2002), and it is known that UCB achieves regret $O(K \log T)$.

The UCB algorithm requires $\Omega(K)$ space since it records the estimated rewards from all of the K arms. However, in settings with limited space such as streaming algorithms, or settings with infinitely many arms (Kleinberg, 2004), the requirement is problematic. There is a significant literature addressing this problem, but existing approaches assume structural properties on the set of arms, e.g. combinatorial structure (Cesa-Bianchi and Lugosi, 2012) or continuum arm with local Lipschitz condition (Kleinberg, 2004). A natural question is, what can we do without these structural assumptions given limited space?

A particular example is in a streaming algorithm setting, where space is much more limited than time, such as a router (Zhang, 2013). If the space constraint is $o(K)$ but the time constraint is $\Omega(K)$, one cannot run traditional UCB. In this case, $O(K)$ regret is still acceptable, and by accepting O(K) total regret, we can avoid requiring structural assumptions. In a router, complicated strategy would corresponds to a larger set K of possible strategies, which grants us the tradeoff: larger K will result in a higher regret with a better optimum. Since routers have strict space constraints, running UCB would result in an extremely small regret on average over time (K/T = space/time, which is acceptable for routers). Our algorithm provides more flexibility in this bias/variance tradeoff.

Our techniques. Our algorithm is based on fairly simple ideas. First, suppose we know the time horizon T and the expected value of the optimal arm μ^*. We could then make a single pass through the arms; for each arm i, flip it until we have high $(1 - 1/T^3)$

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

confidence that $\Delta_i = \mu^* - \mu_i > 0$, where μ_i is the expected value of arm i. Once this happens, move to the next arm. This will flip each arm $O(\frac{\log T}{\Delta_i^2})$ times, inducing regret $O(\frac{\log T}{\Delta_i^2} \cdot \Delta_i)$ from this arm. The total regret will then be $O(\sum_{i \neq i^*} \frac{\log T}{\Delta_i})$, which is ideal, with only constant space required. The problem is that we don't know T or μ^*. Not knowing T isn't a big deal – we can partition the time horizon into $\log \log T$ scales, and the last $\log T$ term will dominate (Auer and Ortner, 2010) – but not knowing μ^* is a serious problem.

We solve this problem by iteratively refining upper and lower bounds μ_{LB} and μ_{UB} on μ^*. In each pass through the data, we get new estimates that are half as far from each other. After $O(\log(1/\Delta))$ passes, where $\Delta = \min_{i:\Delta_i>0} \Delta_i$ is the minimal gap between the optimal and the suboptimal arms, only the best arm i^* will remain in the interval. This gives an algorithm that loses at most an $O(\log(1/\Delta))$ factor in the regret. In some cases, the loss is significantly smaller. Therefore, we can obtain the following result that improves the $O(\log(1/\Delta))$ factor into a $O(\log(\Delta_i/\Delta))$ factor,

Theorem 1.1. *Given a stochastic bandit instance with K arms and their expected values $\mu_1, \cdots \mu_k \in [0,1]$. Let $\mu_* = \max_{i\in[K]} \mu_i$, $\Delta_i = \mu_* - \mu_i$, and $\Delta = \min_{i:\Delta_i>0} \Delta_i$. For any $T > 0$, there exists an algorithm that uses $O(1)$ words of space and achieves regret*

$$O\left(\sum_{i:\Delta_i>0} \frac{1}{\Delta_i} \log \frac{\Delta_i}{\Delta} \log T\right).$$

Recall that the well-known UCB algorithm gives regret $O(\sum_{i:\Delta_i>0} \frac{\log T}{\Delta_i})$. Our algorithm is always within a $\log(\Delta_i/\Delta)$ factor of its space-unlimited version. In certain situations, we can do slightly better by refining our estimate of μ^* by more than a constant factor in each iteration. This gives us the following result

Theorem 1.2. *Under the same setting as Theorem 1.1, for any $\gamma > 0$, there exists an algorithm that uses $O(1)$ words of space and achieves regret*

$$O\left(\sum_{i:\Delta_i>0} \frac{1}{\Delta_i} \left(\log^{\gamma} \frac{1}{\Delta_i} + \frac{\log(\Delta_i/\Delta)}{\gamma \log\log(\Delta_i/\Delta)}\right) \log(T)\right).$$

In particular, if we set $\gamma = 1/2$, we can find that this algorithm is always within an $O\left(\frac{\log(1/\Delta)}{\log\log(1/\Delta)}\right)$ factor of the space-unlimited UCB algorithm.

The paper is presented in the following manner. Section 2 reviews the related work. Section 3 provides detailed preliminaries of problem formulation and the background needed for our result. Section 4 and 5 contains the algorithm that gives the result (I) and (II) of Theorem 1.1 with known time horizon T, respectively. Section 6 demonstrates how to extend the algorithms to the case with unknown time horizon. The full version is available at https://arxiv.org/pdf/1712.09007.

2 Related Works

For stochastic bandits, the seminal work by Lai and Robbins (1985) demonstrated the idea of using the confidence intervals to solve the problem, and it showed that the lower bound of the regret is $\Omega(\sum \frac{\Delta_i \log T}{\mathrm{KL}(\mu_i,\mu_*)})$. The UCB algorithm, which is a simple solution to stochastic bandits, was analyzed in Auer et al. (2002). The UCB algorithm is based on Hoeffding's inequality, which is optimal when $\mathrm{KL}(\mu_i, \mu_*) \approx \Delta_i^2$. In certain situations this can be improved using different types of concentration inequalities; for example, Audibert et al. (2009) used Bernstein's inequality to derive an algorithm with regret depending on the second moments. Later, Garivier and Cappé (2011) and Maillard et al. (2011) independently proposed the KL-UCB algorithm that matches the lower bound. We refer to the reader the comprehensive survey by Bubeck and Cesa-Bianchi (2012) for general bandit problems.

In addition to regret analysis for online decision making, there is a set of papers that discuss the sample complexity for the pure exploration problem, i.e. how to identify the best arm (Mannor and Tsitsiklis, 2004; Even-Dar et al., 2002; Jamieson et al., 2014; Karnin et al., 2013; Kaufmann et al., 2015; Even-Dar et al., 2006). Similar algorithms has been used in the regime of online decision making (Bui et al., 2011; Auer and Ortner, 2010). With the idea of the best arm identification, the explore-then-commit (ETC) policy is designed to first performs some tests to identify the best arm, and then commit to it in the remaining time horizon. The ETC policy is shown to be suboptimal (Garivier et al., 2016) but simplifies the analysis. In particular, our algorithm is based on the framework by Auer and Ortner (2010), but our algorithm takes only $O(1)$ space while the method by Auer and Ortner (2010) takes $O(K)$ space.

Moreover, there is a small set of papers that integrates the sketching techniques from streaming and online learning (Hazan and Seshadhri, 2009; Luo et al., 2016). Hazan and Seshadhri (2009) considered

the problem of minimizing α-exp-concave losses, and the regret is required to be $O(\log T)$ uniformly over time. They used the idea from streaming to keep a small active set of experts. Luo et al. (2016) considered the online convex optimization problem, and they used the ideas of sketching to reduce the efficiency for computing online Newton steps, however, the complexity is still $\Omega(K)$.

3 Preliminary

Notations For any positive integer n, we use $[n]$ to denote the set $\{1, 2, \cdots, n\}$. For random variable X, let $\mathbb{E}[X]$ denote its expectation of X (If this quantity exists). In addition to $O(\cdot)$ notation, for two functions f, g, we use the shorthand $f \lesssim g$ (resp. $\gtrsim$) to indicate that $f \leq Cg$ (resp. $\geq$) for an absolute constant C. We use $f \eqsim g$ to mean $cf \leq g \leq Cf$ for constants c, C.

We measure space in words using the word RAM model, so that the input values (such as K, T, and rewards) and variables can each be expressed in $O(1)$ word of space in $O(\log(KT))$ bits. For more details of word RAM model, we refer the readers to Aho et al. (1974); Cormen et al. (2009).

3.1 Problem Formulations

Definition 3.1. *For a* multi-armed bandit problem*, there are K arms in total, and a finite time horizon $1, 2, \ldots, T$. At each time step $t \in [T]$, the player has to choose an arm $I_t \in [K]$ to play, and receives a reward $X_{i,t}$ associate to that arm. Without loss of generality, assume that for each arm $i \in [K]$ and each time step $t \in [T]$, $X_{i,t} \in [0, 1]$. We denote the arm that player chooses at time t as I_t. The goal of the player is to maximize the total reward he is getting. We will measure the performance of an algorithm via its* regret*, which is defined as the difference between the best reward in the hindsight and the reward received with the algorithm:*

$$\Psi_T = \max_{i \in [K]} \left(\sum_{t=1}^{T} X_{i,t} - \sum_{t=1}^{T} X_{I_t,t} \right).$$

In this paper, we consider the stochastic setting, where we assume the rewards are coming from some stochastic processes.

Definition 3.2. *In a* stochastic bandit*, we assume each arm $i \in [K]$ is associated with a distribution $\mathcal{D}_i$ over $[0, 1]$, with mean μ_i. The reward $X_{i,t}$ at time $t \in [T]$ is drawn from $\mathcal{D}_i$ independently.*

For stochastic bandits, instead of using the regret defined above, we will consider the *pseudo regret*:

$$\overline{\Psi}_T = \max_{i \in [K]} \left(\mathbb{E}\left[\sum_{t=1}^{T} X_{i,t} \right] - \mathbb{E}\left[\sum_{t=1}^{T} X_{I_t,t} \right] \right).$$

We can rewrite the pseudo regret using Wald's identity:

$$\overline{\Psi}_T = \max_{i \in [K]} \left(\sum_{j=1}^{K} \mathbb{E}\left[N_{j,T} \Delta_{ij} \right] \right), \tag{1}$$

where $N_{j,T}$ is the number of times arm j is chosen up to time T, and we define $\Delta_{ij} = \mu_i - \mu_j$ to be the gap between the means of arm i and arm j. We use μ_* to denote the mean reward for the arm with the highest mean, i.e., $\mu_* = \max_{i \in [K]} \mu_i$.

3.2 Concentration Inequalities

In this paper, for simplicity, we will use Chernoff-Hoeffding inequality to analyze the concentration behavior for random variables with bounded support.

Fact 3.3 (Chernoff-Hoeffding Bound)**.** *Let x_1, x_2, $\ldots$, x_n be i.i.d. random variables in $[0, 1]$. Let $X = \frac{1}{n} \sum_{i=1}^{n} x_i$. Then for any $\epsilon > 0$,*

$$\Pr\left[|X - \mathbb{E}[X]| > \epsilon \right] \leq 2e^{-2n\epsilon^2}.$$

4 UCBConstSpace with known T

The original UCB-1 algorithm (Auer et al., 2002) needs $O(K)$ space to achieve $O(\sum_{i:\Delta_i > 0} \frac{1}{\Delta_i} \log T)$ regret. In this section, we propose a new algorithm which requires only $O(1)$ space in exchange for a slightly worse regret.

First, we consider the setting where T is known. The main result is presented in the following theorem.

Theorem 4.1. *Given a stochastic bandit instance with known T, let $\Delta_i = \mu_* - \mu_i$, and let $\Delta = \min_{i:\Delta_i > 0} \Delta_i$. Then for any $T > 0$, there exists an algorithm that uses $O(1)$ words of space and achieves regret*

$$O\left(\sum_{i:\Delta_i > 0} \frac{1}{\Delta_i} \log(\Delta_i / \Delta) \log T \right).$$

We present the method in Algorithm 1, where we iteratively improve our estimation of Δ. More precisely, we scan through the data multiple rounds. In the r-th round, we sample each arm up to some

Algorithm 1 UCB algorithm with constant space and known T (Theorem 4.1)

1: **procedure** UCBConstSpace(K, T)
2: Set $\delta \leftarrow 1/T^3$, initialize $g_1 \leftarrow \frac{1}{2}$, $t \leftarrow 1$
3: **Exploration Phase:**
4: **for** rounds $r = 1, 2, \ldots$ **do**
5: a': the best arm in the previous round, $\overline{\mu}'$: mean reward for arm a' in the previous round
6: $N \leftarrow \lceil 2\log(1/\delta)/g_r^2 \rceil$, which is the maximum number of plays for each arm in the current round
7: Initialize $a, b \leftarrow 0$, which are the best and the second best arm in this round
8: Initialize $\overline{\mu}_a, \overline{\mu}_b \leftarrow 0$, which are the means for arms a and b
9: **for** each arm $i = 1 \rightarrow K$ **do**
10: Set $\overline{\mu} \leftarrow 0$, which keeps the mean reward for arm i in the current round
11: **for** $n = 1 \rightarrow N$ **do**
12: Pull arm i and receive reward v
13: $t \leftarrow t + 1$
14: Update $\overline{\mu}$ with v: $\overline{\mu} \leftarrow (\overline{\mu} \cdot (n-1) + v)/n$
15: **if** $\overline{\mu} + \sqrt{\log(1/\delta)/2n} < \overline{\mu}' - g_{r-1}/2$ **then**
16: **break**, i.e. we *rule out* arm i for the current round
17: **end if**
18: **end for**
19: **if** $\overline{\mu} > \overline{\mu}_a$ **then** $b \leftarrow a$, $\overline{\mu}_b \leftarrow \overline{\mu}_a$, $a \leftarrow i$ and $\overline{\mu}_a \leftarrow \overline{\mu}$ ▷ Update the best and the 2nd best arms
20: **else if** $\overline{\mu} > \overline{\mu}_b$ **then** $b \leftarrow i$ and $\overline{\mu}_b \leftarrow \overline{\mu}$ ▷ Update the 2nd best arm
21: **end for**
22: **Stopping Criterion: if** $\overline{\mu}_a - g_r/2 > \overline{\mu}_b + g_r/2$ or $t > T$ **then break**
23: Update $a' = a$ and $\overline{\mu}' = \overline{\mu}_a$
24: **Set new precision:** $g_{r+1} = g_r/2$
25: **end for**
26: **Exploitation Phase:**
27: Pull arm a for the remaining time steps.
28: **end procedure**

precision g_r. The desired precision g_r is halved after each round. In this sampling process, we only keep the information of the best arm and the second best arm seen in the current and the previous round, instead of saving those from all arms. With the information of the best arm and the current precision g_r, we can refine the upper and lower bound μ_{UB} and μ_{LB} on μ_*. If an arm whose upper confidence value is less than μ_{LB}, we can rule it out without continuing to g_r precision. This process is terminated if we are able to determine the best arm with the rest arms.

We define $a^{(r)}$ and $b^{(r)}$ as the best arm and the second best arm stored at the end of the r-th round. Also, we let $\overline{\mu}_i^{(r)}$ to be the recorded empirical mean at the end of the r-th round for arm i. Denote $n_i^{(r)}$ as the total number of pulls of arm i at the r-th round. Then, we define $\overline{\mu}_{i,n}^{(r)}$ as the empirical mean $\overline{\mu}_i$ stored for arm i after pulling it for n times in round r. Further, we define $r_{\max}$ as the value of $r-1$ at the moment the algorithm exits the loop in Line 22.

Definition 4.2. *For each $r \in [r_{\max}]$, define the event ξ_r to be the event: $\exists r' \in [r], \exists i \in [K], \exists n \in [n_i^{(r')}]$ such that $|\overline{\mu}_{i,n}^{(r')} - \mu_i| > \sqrt{\log(1/\delta)/(2n)}$, i.e., there exists some estimate of $\overline{\mu}_{i,n}^{(r')}$ that is not within our desired confidence interval up to round r.*

Throughout the first part of our analysis, we focus on the case when $\neg\xi_r$ holds when we are discussing the state of the algorithm at round r, i.e., all estimates are within our desired confidence interval.

Lemma 4.3. *In Algorithm 1, at any round $r \in [r_{\max}]$, given $\neg\xi_r$, the following statements are true:*
1. $n_{a^{(r)}}^{(r)} = \lceil (2\log(1/\delta))/g_r^2 \rceil$, i.e. the claimed optimal arm cannot be ruled out early.
2. $n_*^{(r)} = \lceil (2\log(1/\delta))/g_r^2 \rceil$, i.e. the true optimal arm cannot be ruled out early.
3. $|\overline{\mu}_{a^{(r)}}^{(r)} - \mu_*| \le g_r/2$.

Proof. We prove this lemma by induction. For the base case, the first and the second statement are true because all arms have to be played for $\lceil \frac{2\log(1/\delta)}{(g_1)^2} \rceil$

times. For the third statement, we prove by contradiction. Assume the contrary, i.e. $\overline{\mu}^{(1)}_{a^{(1)}} - \mu_* > g_1/2$ or $\mu_* - \overline{\mu}^{(1)}_{a^{(1)}} > g_1/2$. If $\overline{\mu}^{(1)}_{a^{(1)}} - \mu_* > g_1/2$, then we have

$$\begin{aligned}\mu_* &< \overline{\mu}^{(1)}_{a^{(1)}} - g_1/2 \\ &\le \mu_{a^{(1)}} + \sqrt{\log(1/\delta)/(2n^{(1)}_{a^{(1)}})} - g_1/2 \\ &\le \mu_{a^{(1)}} + g_1/2 - g_1/2 \\ &= \mu_{a^{(1)}}\end{aligned}$$

where the second step follows by condition $\neg\xi_r$ and the third step follows by $n^{(1)}_{a^{(1)}} \ge (2\log(1/\delta))/g_1^2$.

The above equation leads to a contradiction because $\mu_* > \mu_i$ for any $i \neq *$. Similarly, if $\mu_* - \overline{\mu}^{(1)}_{a^{(1)}} > g_1/2$, then we have

$$\begin{aligned}\overline{\mu}^{(1)}_{a^{(1)}} &< \mu_* - g_1/2 \\ &\le \overline{\mu}^{(1)}_* + \sqrt{\log(1/\delta)/(2n^{(1)}_*)} - g_1/2 \\ &\le \overline{\mu}^{(1)}_* + g_1/2 - g_1/2 \\ &= \overline{\mu}^{(1)}_*\end{aligned}$$

where the second step follows by condition $\neg\xi_r$, and the third step follows by $n^{(1)}_* \ge (2\log(1/\delta))/g_1^2$.

The above equation also results in a contradiction because for any $i \neq *$ to be assigned as $a^{(1)}$, we must have $\overline{\mu}^{(1)}_{a^{(1)}} > \overline{\mu}^{(1)}_*$.

For the induction step, we assume these three statements are true for $r \le r' - 1$. Now consider $r = r'$. We first prove the second statement. Assume the contrary, i.e. the true optimal arm has been ruled out early, meaning

$$\overline{\mu}^{(r)}_* + \sqrt{\log(1/\delta)/(2n^{(r)}_*)} < \overline{\mu}^{(r-1)}_{a^{(r-1)}} - g_{r-1}/2 \quad (2)$$

Then, we can see that

$$\begin{aligned}\mu_* &\le \overline{\mu}^{(r)}_* + \sqrt{\log(1/\delta)/(2n^{(r)}_*)} \\ &< \overline{\mu}^{(r-1)}_{a^{(r-1)}} - g_{r-1}/2 \\ &\le \mu_{a^{(r-1)}} \qquad (3)\end{aligned}$$

where in the last inequality, we use the induction hypothesis, $n^{(r-1)}_{a^{(r-1)}} \ge \frac{2\log(1/\delta)}{g^2_{r-1}}$ and then

$$\mu_{a^{(r-1)}} \ge \overline{\mu}^{(r-1)}_{a^{(r-1)}} - \sqrt{\frac{\log(1/\delta)}{2n^{(r-1)}_{a^{(r-1)}}}} \ge \overline{\mu}^{(r-1)}_{a^{(r-1)}} - g_{r-1}/2$$

There is a contradiction in (3) because we must have $\mu_* \ge \mu_{a^{(r-1)}}$. Hence the second statement is true.

Next, we can see that the first statement is now clear because we have shown that there is at least one arm that is going to pull for $\lceil \frac{2\log(1/\delta)}{g_r^2} \rceil$ times at the r-th round (which is arm $*$ according to the second statement we have just shown). This means that if arm $a^{(r)}$ is not arm $*$, then it has to be pulled for $\lceil \frac{2\log(1/\delta)}{g_r^2} \rceil$ times as well.

For the third statement, the proof is similar to the base case, where we prove by contradiction. Assume the contrary, i.e. $\overline{\mu}^{(r)}_{a^{(r)}} - \mu_* > g_r/2$ or $\mu_* - \overline{\mu}^{(r)}_{a^{(r)}} > g_r/2$.

If $\overline{\mu}^{(r)}_{a^{(r)}} - \mu_* > g_r/2$, then we have

$$\begin{aligned}\mu_* &< \overline{\mu}^{(r)}_{a^{(r)}} - g_r/2 \\ &\le \mu_{a^{(r)}} + \sqrt{\log(1/\delta)/(2n^{(r)}_{a^{(r)}})} - g_r/2 \\ &\le \mu_{a^{(r)}} + g_r/2 - g_r/2 \\ &= \mu_{a^{(r)}}\end{aligned}$$

where the second step follows by condition $\neg\xi_r$, and the third step follows by $n^{(r)}_{a^{(r)}} \ge \frac{2\log(1/\delta)}{g_r^2}$(the first statement).

This results in a contradiction because $\mu_* \ge \mu_i$ for any $i \in [K]$. Similarly, if $\mu_* - \overline{\mu}^{(r)}_{a^{(r)}} > g_r/2$, then we have

$$\begin{aligned}\overline{\mu}^{(r)}_{a^{(r)}} &< \mu_* - g_r/2 \\ &\le \overline{\mu}^{(r)}_* + \sqrt{\log(1/\delta)/(2n^{(r)}_*)} - g_r/2 \\ &\le \overline{\mu}^{(r)}_* + g_r/2 - g_r/2 \\ &= \overline{\mu}^{(r)}_*\end{aligned}$$

where the second step follows by condition $\neg\xi_r$ and the third step follows by $n^{(r)}_* \ge \frac{2\log(1/\delta)}{g_r^2}$(the second statement).

This results in a contradiction because for any $i \neq *$ to be assigned as $a^{(r)}$, we must have $\overline{\mu}^{(r)}_{a^{(r)}} > \overline{\mu}^{(r)}_*$, otherwise we will have $|\overline{\mu}^{(r)}_* - \mu_*| \le g_r/2$ by condition $\neg\xi_r$. □

Lemma 4.4. *In Algorithm 1, conditioning on event $\neg\xi_{r_{\max}}$ holds, we have $r_{\max} \le \lceil \log(2/\Delta) \rceil$.*

Proof. Assume the contrary, i.e. at the end of round $r = \lceil \log(2/\Delta) \rceil$, the best arm and the second best arm are still not differentiated, meaning we still have

$$\overline{\mu}^{(r)}_* - g_r/2 < \overline{\mu}^{(r)}_{a^{(r)}} + g_r/2$$

First note that $r > \log(2/\Delta)$ implies $2^{-r} = g_r < \Delta/2$. We have

$$\begin{aligned}\mu_* &\le \overline{\mu}_*^{(r)} + \sqrt{\log(1/\delta)/(2n_*^{(r)})} \\ &\le \overline{\mu}_*^{(r)} + g_r/2 < \overline{\mu}_{a^{(r)}}^{(r)} + 3g_r/2 \\ &< \overline{\mu}_{a^{(r)}}^{(r)} + 3\Delta/4\end{aligned}$$

Similarly, we have $\mu_{a^{(r)}} > \overline{\mu}_{a^{(r)}}^{(r)} - \Delta/4$. Then, we can show that

$$\Delta \le \mu_* - \mu_a \le (\overline{\mu}_{a^{(r)}}^{(r)} + 3\Delta/4) - (\overline{\mu}_{a^{(r)}}^{(r)} - \Delta/4) < \Delta$$

which results in a contradiction. This implies that given $\neg\xi_{r_{\max}}$, we must have $r_{\max} \le \lceil \log(2/\Delta) \rceil$. □

Lemma 4.5. *In Algorithm 1, at any round r, given $\neg\xi_r$, the number of plays for any arm $i \in [K]$ is upper-bounded by*

$$n_i^{(r)} \le \frac{2\log(1/\delta)}{(\Delta_i - g_{r-1})^2} + 1.$$

Proof. First, note that as long as an arm has not been ruled out, we have

$$\overline{\mu}^r_{i,n_i^{(r)}-1} + \sqrt{\frac{\log(1/\delta)}{2(n_i^{(r)}-1)}} \ge \overline{\mu}_{a^{(r-1)}}^{(r-1)} - \frac{g_{r-1}}{2} \tag{4}$$

Then, we can show

$$\begin{aligned}\Delta_i &= \mu_* - \mu_i \\ &\le \overline{\mu}_{a^{(r-1)}}^{(r-1)} + \frac{g_{r-1}}{2} - \mu_i \\ &\le \overline{\mu}_{a^{(r-1)}}^{(r-1)} + \frac{g_{r-1}}{2} - \left(\overline{\mu}^{(r)}_{i,n_i^{(r)}-1} - \sqrt{\frac{\log(1/\delta)}{2(n_i^{(r)}-1)}}\right) \\ &\le 2\left(\frac{g_{r-1}}{2} + \sqrt{\frac{\log(1/\delta)}{2(n_i^{(r)}-1)}}\right)\end{aligned}$$

where the second step follows from Lemma 4.3, the third step follows by $\neg\xi_r$, and the last step follows by (4). Reorganizing the above inequality proves the lemma. □

Proof of Theorem 4.1. Consider Algorithm 2. For each round $r \in [r_{\max}]$, conditioned on $\neg\xi_r$, i.e. the confidence interval is correct, we first recognize two bounds on the number of plays $n_i^{(r)}$ for each arm $i \in [K]$.

By the definition of Algorithm 1, we have

$$n_i^{(r)} \le 2\log(1/\delta)/g_r^2 + 1 \tag{5}$$

Also, from Lemma 4.5, we have

$$n_i^{(r)} \le 2\log(1/\delta)/(\Delta_i - g_{r-1})^2 + 1 \tag{6}$$

By combining (5) and (6), together with $r_{\max} \le \lceil \log(2/\Delta) \rceil$ by Lemma 4.4, we can upper bound the regret results from pulling arm i in the algorithm. Let $\alpha = \lceil \log(2/\Delta) \rceil$ and $\beta = \lceil \log(3/\Delta_i) \rceil$. Conditioning on event $\neg\xi_{r_{\max}}$ holds, we have,

$$\begin{aligned}\sum_{r=1}^{\alpha} \Delta_i n_i^{(r)} &\le \sum_{r=1}^{\alpha} \Delta_i \left(\frac{2\log(1/\delta)}{(\max\{g_r, \Delta_i - 2g_r\})^2} + 1\right) \\ &= \sum_{r=1}^{\alpha} \Delta_i \left(\frac{2\log(1/\delta)}{(\max\{2^{-r}, \Delta_i - 2\cdot 2^{-r}\})^2} + 1\right)\end{aligned}$$

Furthermore, we can obtain

$$\begin{aligned}&\sum_{r=1}^{\alpha} \Delta_i n_i^{(r)} \\ &\le \sum_{r=1}^{\beta} \Delta_i \cdot \frac{2\log(1/\delta)}{2^{-2r}} + \sum_{r=\beta+1}^{\alpha} \Delta_i \cdot \frac{2\log(1/\delta)}{(\Delta_i - 2\cdot 2^{-\log(3/\Delta_i)})^2} \\ &+ \Delta_i \cdot \lceil \log(2/\Delta) \rceil \\ &\le \frac{288\log(1/\delta)}{\Delta_i} + \frac{18\log(2\Delta_i/3\Delta)\log(1/\delta)}{\Delta_i} \\ &+ \Delta_i(\log(2/\Delta) + 1) \\ &\lesssim \frac{\log(\Delta_i/\Delta)\log(1/\delta)}{\Delta_i}\end{aligned} \tag{7}$$

For the next step, we find an upper bound for the probability of event $\xi_{r_{\max}} := \{\exists r \in [r_{\max}], \exists i \in [K], \exists n \in [n_i^{(r)}] \text{ s.t. } |\overline{\mu}_{i,n}^{(r)} - \mu_i| > \sqrt{\log(1/\delta)/(2n)}\}$:

$$\begin{aligned}&\Pr(\xi_{r_{\max}}) \\ &\le \sum_{r=1}^{T/K}\sum_{i=1}^{K}\sum_{n=1}^{T} \Pr\left(|\overline{\mu}_{i,n}^{(r)} - \mu_i| > \sqrt{\log(1/\delta)/(2n)}\right) \\ &\le 2T^2\delta\end{aligned} \tag{8}$$

Finally, by choosing $\delta = 1/T^3$, and combining (7) and (8), we have

$$\begin{aligned}\overline{\Psi}_T &\lesssim \sum_{i=1}^{K}\left(\frac{\log(\Delta_i/\Delta)\log(T)}{\Delta_i} + \Delta_i T \cdot 2T^2\delta\right) \\ &\lesssim \sum_{i=1}^{K}\frac{\log(\Delta_i/\Delta)\log(T)}{\Delta_i}\end{aligned}$$

which proves the theorem. □

5 Improved Algorithm for UCBConstSpace

The result in Theorem 2 gives an additional $O(\log(\Delta_i/\Delta))$ factor to the original UCB-1 algorithm by Auer et al. (2002). This means that in a

bad scenario, for example, if most of the arms have gap $\Delta_i = K\Delta$, the $O(\log(\Delta_i/\Delta))$ factor translates to an additional $\log K$ factor in the regret.

In this section, we show that we are able to improve the additional $\log(\Delta_i/\Delta)$ factor to a $\frac{\log(\Delta_i/\Delta)}{\log\log(\Delta_i/\Delta)}$ factor by slightly changing the update rule on the precision g_r. This means that in the bad example described above, we are improving the competitive ratio from $\log K$ to $\frac{\log K}{\log\log K}$. We present our result in the following theorem.

Theorem 5.1. *Given a stochastic bandit instance with known T, let $\Delta_i = \mu_* - \mu_i$, and let $\Delta = \min_{i:\Delta_i>0}\Delta_i$. For any $\gamma > 0$ and any $T > 0$, there exists an algorithm that uses $O(1)$ words of space and achieves regret*

$$O\left(\sum_{i:\Delta_i>0}\frac{1}{\Delta_i}\left(\log^\gamma\frac{1}{\Delta_i}+\frac{\log(\Delta_i/\Delta)}{\gamma\log\log(\Delta_i/\Delta)}\right)\log(T)\right).$$

We consider a modified version of Algorithm 1, where the update rule in Line 24 is replaced by

$$g_{r+1} = \frac{g_r}{2\left(\log(1/g_r)\right)^\epsilon} \tag{9}$$

where ϵ is some constant to be determined later. In the following lemma, we show that with this update rule, basically given any $D < 1$, it takes only $O\left(\frac{1}{\epsilon}\cdot\frac{\log(1/D)}{\log\log(1/D)}\right)$ steps to reach accuracy D.

Lemma 5.2. *Given any $g_0, D \in (0,1)$, $D < g_0$, let $r_0 = \frac{\log(g_0/D)}{\log\log(g_0/D)}$. If for any positive integer r, $g_r = \frac{g_{r-1}}{2(\log(1/g_{r-1}))^\epsilon}$. Then, for any $r \geq (\frac{2}{\epsilon}+1)r_0+2$, we have $g_r \leq D$.*

Proof. First, note that by definition of g_r, we have $g_r \leq g_0 2^{-r}$ for any $r \geq 1$. Therefore, for any $r \geq r_0$, we have

$$g_r \leq g_0 2^{-r} \leq g_0 2^{-r_0} = g_0(D/g_0)^{\frac{1}{\log\log(g_0/D)}}$$

Then, we can see that for any $r \geq r_0$, we have

$$g_{r+1} \leq \frac{g_r}{2\left(\frac{\log(g_0/D)}{\log\log(g_0/D)}\right)^\epsilon} \leq \frac{g_r}{2(\log(g_0/D))^{\epsilon/2}}$$

As a result, we have

$$\begin{aligned} g_{r+\lceil\frac{2}{\epsilon}r_0\rceil} &\leq \frac{g_r}{2^{\frac{2}{\epsilon}r_0}(\log(g_0/D))^{r_0}} \\ &\leq g_0(\log(g_0/D))^{-r_0} = D \end{aligned}$$

This implies that for any $r \geq (r_0+1)+(\frac{2}{\epsilon}r_0+1) \geq \lceil r_0\rceil + \lceil\frac{2}{\epsilon}r_0\rceil$, we have $g_r \leq D$. □

Note that we can apply Lemma 4.3 and Lemma 4.5 for Algorithm 2 with update rule (9) because they do not require specific update rules. Before we proceed to the proof of Theorem 5.1, we need the following lemma for an upper bound of $r_{\max}$.

Lemma 5.3. *In Algorithm 2 with update rule (9), given $\neg\xi_{r_{\max}}$, we have $r_{\max} \leq \lceil(\frac{2}{\epsilon}+1)\frac{\log 2/\Delta}{\log\log 2/\Delta}+2\rceil$.*

Due to space contraint, we provide the detailed proof of the lemma in the full version of our paper (Liau et al., 2017).

Proof of Theorem 5.1. Consider Algorithm 2 with update rule (9). For each arm $i \in [K]$, if we condition on $\neg\xi_{r_{\max}}$, then by Lemma 4.5 and Lemma 5.3, we can upper bound the regret results from pulling arm i in the algorithm:

$$\begin{aligned} &\sum_{r=1}^{r_{\max}}\Delta_i n_i^{(r)} \\ &\leq \sum_{r=1}^{r_{\max}}\Delta_i\left(\frac{2\log(1/\delta)}{\left(\max\{g_r,\Delta_i-g_{r-1}\}\right)^2}+1\right) \\ &\leq \sum_{r=1}^{r_i}\Delta_i\cdot\frac{2\log(1/\delta)}{g_r^2}+\sum_{r=r_i+1}^{r_{\max}}\Delta_i\cdot\frac{2\log(1/\delta)}{(\Delta_i-g_{r-1})^2} \\ &\quad+\Delta_i\cdot r_{\max} \end{aligned} \tag{10}$$

where r_i be the minimal round r such that $g_r < \Delta_i/2$. For the first term of (10), since g_r decays super-exponentially, i.e. $g_{r+1} \leq g_r/2$, we have

$$\begin{aligned} \sum_{r=1}^{r_i}\frac{2\Delta_i\log(1/\delta)}{g_r^2} &\leq \frac{4\Delta_i\log(1/\delta)}{g_{r_i}^2} \\ &= \frac{4\Delta_i\left(\log\frac{1}{g_{r_i-1}}\right)^{2\epsilon}\log(1/\delta)}{g_{r_i-1}^2} \\ &\leq \frac{16\left(\log\frac{1}{\Delta_i}\right)^{2\epsilon}}{\Delta_i}\log(1/\delta) \end{aligned} \tag{11}$$

where the last step follows from the fact that $g_{r_i-1} \geq \Delta_i/2$ by the definition of r_i. For the second term of (10), we have

$$\begin{aligned} \sum_{r=r_i+1}^{r_{\max}}\frac{2\Delta_i\log(1/\delta)}{(\Delta_i-g_{r-1})^2} &\leq \sum_{r=r_i+1}^{r_{\max}}\frac{8\Delta_i\log(1/\delta)}{\Delta_i^2} \\ &\leq \sum_{r=r_i+1}^{r_{\max}}\frac{8\log(1/\delta)}{\Delta_i} \end{aligned} \tag{12}$$

By Lemma 5.2, we can find that it takes $\lceil(\frac{2}{\epsilon}+1)\frac{\log(\Delta_i/\Delta)}{\log\log(\Delta_i/\Delta)}+2\rceil$ rounds to get from $\Delta_i/2$ to $\Delta_i/2$.

As a result, we can upper bound (12) by

$$\begin{aligned}
&\sum_{r=r_i+1}^{r_{\max}} \frac{2\Delta_i \log(1/\delta)}{(\Delta_i - g_{r-1})^2} \\
&\leq \left(\left(\frac{2}{\epsilon}+1\right)\frac{\log(\Delta_i/\Delta)}{\log\log(\Delta_i/\Delta)}+3\right)\frac{8\log(1/\delta)}{\Delta_i} \\
&\leq \left(\frac{2}{\epsilon}+1\right)\frac{\log(\Delta_i/\Delta)}{\log\log(\Delta_i/\Delta)}\frac{16\log(1/\delta)}{\Delta_i} \qquad (13)
\end{aligned}$$

Using the similar argument as we have done in the proof of Theorem 4.1, we can find that

$$\Pr(\xi_{r_{\max}}) \leq 2T^2\delta \qquad (14)$$

Finally, by combining (11), (13), and (14), we can get

$$\begin{aligned}
\overline{\Psi}_T \leq 16 \sum_{i:\Delta_i>0} \frac{1}{\Delta_i}\left(\log^{2\epsilon}\frac{1}{\Delta_i} + \left(\frac{2}{\epsilon}+1\right)\frac{\log(\Delta_i/\Delta)}{\log\log(\Delta_i/\Delta)}\right) \\
\cdot \log(1/\delta) + \sum_{i:\Delta_i>0} \Delta_i T \cdot 2T^2\delta
\end{aligned}$$

By choosing $\delta = 1/T^3$ and $\epsilon = \gamma/2$ we can find that

$$\overline{\Psi}_T \lesssim \sum_{i:\Delta_i>0} \frac{1}{\Delta_i}\left(\log^{\gamma}\frac{1}{\Delta_i} + \frac{\log(\Delta_i/\Delta)}{\gamma\log\log(\Delta_i/\Delta)}\right)\log(T)$$

which proves the theorem. □

We conjecture below that the $O(\frac{\log(\Delta_i/\Delta)}{\log\log(\Delta_i/\Delta)})$ factor is not improvable given the $O(1)$ space constraint. The discussion for our conjectured hard instance is in the full version of our paper (Liau et al., 2017).

Conjecture 5.4. *There exists a distribution over stochastic bandit problems such that, for any algorithm taking $O(1)$ words of space will have regret*

$$\Omega\left(\sum_{i:\Delta_i>0}\frac{1}{\Delta_i}\left(\frac{\log(\Delta_i/\Delta)}{\log\log(\Delta_i/\Delta)}\right)\log(T)\right).$$

6 Unknown Horizon T

Now, we show that using the technique described in (Auer and Ortner, 2010), we are able to get the same regret as in Theorem 4.1 if T is unknown.

Theorem 6.1 (Restatement of Theorem 1.1)**.** *Given a stochastic bandit instance with unknown T, let $\Delta_i = \mu_* - \mu_i$, and let $\Delta = \min_{i:\Delta_i>0}\Delta_i$. For any $T > 0$, there exists an algorithm that uses $O(1)$ words of space and achieves regret*

$$O\left(\sum_{i:\Delta_i>0}\frac{\log(\Delta_i/\Delta)}{\Delta_i}\log T\right)$$

Algorithm 2 UCB algorithm with constant space and unknown T (Theorem 6.1 and Theorem 5.1)

```
procedure UCBCS-UnknownT(K)
    Initialize T_0 ← 10
    l ← 0, t ← 1
    while t ≤ T do
        Call UCBConstSpace(K, T_l),
        t ← t + T_l
        l ← l + 1
        T_l ← T_{l-1}^2
    end while
end procedure
```

Due to space constraints, we defer proof of this theorem to the full version of our paper (Liau et al., 2017). Similarly, we are able to use this trick for the improved algorithm in Section 5 and get the same regret as in Theorem 5.1.

Theorem 6.2 (Restatement of Theorem 1.2)**.** *Given a stochastic bandit instance with unknown T, let $\Delta_i = \mu_* - \mu_i$, and let $\Delta = \min_{i:\Delta_i>0}\Delta_i$. For any $\gamma > 0$ and any $T > 0$, there exists an algorithm that uses $O(1)$ words of space and achieves regret*

$$O\left(\sum_{i:\Delta_i>0}\frac{1}{\Delta_i}\left(\log^{\gamma}\frac{1}{\Delta_i} + \frac{\log(\Delta_i/\Delta)}{\gamma\log\log(\Delta_i/\Delta)}\right)\log(T)\right).$$

7 Conclusion

We proposed a constant space algorithm for the stochastic multi-armed bandits problem. Our algorithms proceeds by iteratively refining a confidence interval containing the best arm's value. In the simpler version of our algorithm, we refine the interval by a constant factor in each step, and each iteration only uses $O(\mathrm{OPT})$ regret. This gives an $O(\log\frac{1}{\Delta})$-competitive algorithm. We then showed how to improve this by an $O(\log\log\frac{1}{\Delta})$ factor in certain cases, by using fewer rounds that give more progress. Finally, we showed how to adapt our algorithms—which involve parameters that depend on the time horizon T—to situations with unknown time horizon.

References

A. V. Aho, J. Hopcroft, and J. D. Ullman. The design and analysis of computer algorithms. In *Addison-Wesley Series in Computer Science and Information Processing*, 1974.

J.-Y. Audibert, R. Munos, and C. Szepesvári. Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science*, 410(19):1876–1902, 2009.

P. Auer and R. Ortner. UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*, 61(1-2):55–65, 2010.

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

S. Bubeck and N. Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

L. X. Bui, R. Johari, and S. Mannor. Committing bandits. In *Advances in Neural Information Processing Systems*, pages 1557–1565, 2011.

N. Cesa-Bianchi and G. Lugosi. Combinatorial bandits. *Journal of Computer and System Sciences*, 78(5):1404–1422, 2012.

T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to algorithms*. MIT press, 2009.

E. Even-Dar, S. Mannor, and Y. Mansour. PAC bounds for multi-armed bandit and markov decision processes. In *International Conference on Computational Learning Theory*, pages 255–270. Springer, 2002.

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

A. Garivier and O. Cappé. The KL-UCB algorithm for bounded stochastic bandits and beyond. In *COLT*, pages 359–376, 2011.

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

E. Hazan and C. Seshadhri. Efficient learning algorithms for changing environments. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 393–400. ACM, 2009.

K. G. Jamieson, M. Malloy, R. D. Nowak, and S. Bubeck. lil'ucb: An optimal exploration algorithm for multi-armed bandits. In *COLT*, volume 35, pages 423–439, 2014.

Z. S. Karnin, T. Koren, and O. Somekh. Almost optimal exploration in multi-armed bandits. *ICML (3)*, 28:1238–1246, 2013.

E. Kaufmann, O. Cappé, and A. Garivier. On the complexity of best arm identification in multi-armed bandit models. *The Journal of Machine Learning Research*, 2015.

R. D. Kleinberg. Nearly tight bounds for the continuum-armed bandit problem. In *Advances in Neural Information Processing Systems*, pages 697–704, 2004.

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

D. Liau, E. Price, Z. Song, and G. Yang. Stochastic multi-armed bandits in constant space. In *arXiv preprint*. https://arxiv.org/pdf/1712.09007, 2017.

H. Luo, A. Agarwal, N. Cesa-Bianchi, and J. Langford. Efficient second order online learning by sketching. In *Advances in Neural Information Processing Systems*, pages 902–910, 2016.

O.-A. Maillard, R. Munos, G. Stoltz, et al. A finite-time analysis of multi-armed bandits problems with kullback-leibler divergences. In *COLT*, pages 497–514, 2011.

S. Mannor and J. N. Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*, 5(Jun):623–648, 2004.

Q. Zhang. Introduction. In *Lecture notes of Sublinear Algorithms for Big Data*. http://homes.soic.indiana.edu/qzhangcs/B669-13-fall-sublinear/slides/space-1-dist.pdf, 2013.