BATCHED LIPSCHITZ BANDITS

Anonymous authors
Paper under double-blind review

ABSTRACT

In this paper, we study the batched Lipschitz bandit problem, where the expected reward is Lipschitz and the reward observations are collected in batches. We introduce a novel landscape-aware algorithm, called Batched Lipschitz Narrowing (BLiN), that naturally fits into the batched feedback setting. In particular, we show that for a T-step problem with Lipschitz reward of zooming dimension d_z, our algorithm achieves theoretically optimal regret rate of $\widetilde{\mathcal{O}}\left(T^{\frac{d_z+1}{d_z+2}}\right)$ using only $\mathcal{O}\left(\frac{\log T}{d_z}\right)$ batches. For the lower bound, we show that in an environment with B-batches, for any policy π, there exists a problem instance such that the expected regret is lower bounded by $\widetilde{\Omega}\left(R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}}\right)$, where $R_z(T)$ is the regret lower bound for vanilla Lipschitz bandits that depends on the zooming dimension d_z, and d is the dimension of the arm space.

1 INTRODUCTION

Multi-Armed Bandit (MAB) algorithms aim to exploit the good options while explore the decision space. These algorithms and methodologies find successful applications in artificial intelligence and reinforcement learning (e.g., Silver et al., 2016). While the classic MAB setting assumes that the rewards are immediately observed after each arm pull, real-world data often arrives in different patterns. For example, data from field experiments are often be collected in a batched fashion (Pocock, 1977), since the field researchers can hardly collect back every questionnaire immediately after it is filled out. Another example is from distributed computing. In such settings, each leaf node stores its observations locally and communicate the data every once in a while. From the central node's perspective, the observation arrives in a batched version. In such cases, any observation-dependent decision-making should comply with this data-arriving pattern, including MAB algorithms.

In this paper, we study the batched Lipschitz bandit problem – a MAB problem where the expected reward is Lipschitz and the observations are collected in batches. The model assumes that the time horizon T is divided into B batches by a grid $\mathcal{T} = \{t_0, \cdots, t_B\}$, where $0 = t_0 < t_1 < \cdots < t_B = T$. For any $t_{j-1} < t \le t_j$, the reward at time t cannot be observed until time t_j, and the decision made at time t depends only on rewards up to time t_{j-1}. Under the batched setting, the player may have significantly fewer observations when making decisions (Gao et al., 2019). This limitation of data interaction causes difficulty for both the exploration and the exploitation procedure. Due to this difficulty, existing algorithms for Lipschitz bandit (e.g., Kleinberg et al., 2008; Bubeck et al., 2009) fails to solve the batched setting.

To this end, we present a novel adaptive algorithm for batched Lipschitz bandit problems, named *Batched Lipschitz Narrowing* (BLiN). BLiN is adaptive in two respects. First, the algorithm learns the landscape of the reward by adaptively narrowing the arm set, so that regions of high reward are more frequently played. Second, the algorithm determines the data collection procedure adaptively, so that only very few data communications (number of batches) are needed. We show that for a T-step problem with Lipschitz reward of zooming dimension d_z, BLiN only needs $\mathcal{O}\left(\frac{\log T}{d_z}\right)$ batches to achieve the theoretically optimal regret rate of $\widetilde{\mathcal{O}}\left(T^{\frac{d_z+1}{d_z+2}}\right)$. In other words, BLiN achieves the regret rate of the best existing Lipschitz bandit methods (Kleinberg et al., 2008; Bubeck et al., 2009) using only $\mathcal{O}\left(\frac{\log T}{d_z}\right)$ batches. Since all existing Lipschitz bandit methods requires T batches, BLiN significantly saves the data communication cost for Lipschitz bandit problems.

We also provide complexity analysis for this problem. More precisely, we show that if the observations are collected in B batches, then for any policy π, there exists a problem instance such that the expected regret is at best $\widetilde{\Omega}\left(R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}}\right)$, where $R_z(T)$ is the regret lower bound for vanilla Lipschitz bandits that depends on the zooming dimension d_z (see Section 2 for the exact definition), and d is the dimension of the arm space. This results implies that $\Omega(\frac{\log \log T}{\log d})$ batches are necessary to achieve optimal regret rate.

1.1 Related Works

The history of the Multi-Armed Bandit (MAB) problem can date back to Thompson (1933). Solvers for this problems include the UCB algorithms (Lai & Robbins, 1985; Agrawal, 1995b; Auer et al., 2002a), the arm elimination method (Even-Dar et al., 2006; Perchet & Rigollet, 2013), the ϵ-greedy strategy (Auer et al., 2002a; Sutton & Barto, 2018), the exponential weights and mirror descent framework (Auer et al., 2002b).

Recently, with the prevalence of distributed computing and large-scale field experiments, the setting of batched feedback has captured attention (e.g., Cesa-Bianchi et al., 2013). Perchet et al. (2016) mainly consider batched bandit with two arms, and a matching lower bound for static grid is proved. It was then generalized by Gao et al. (2019) to finite-armed bandit problems. In their work, the authors designed an elimination method for finite-armed bandit problem and proved matching lower bounds for both static and adaptive grid. Soon afterwards, Ruan et al. (2021) provides a solution for batched bandit with linear reward. Parallel to the regret control regime, best arm identification with limited number of batches was studied in Agarwal et al. (2017) and Jun et al. (2016). Top-k arm identification in the collaborative learning framework is also closely related to the batched setting, where the goal is to minimize the number of iterations (or communication steps) between agents. In this setting, tight bounds are obtained in the recent works by Tao et al. (2019); Karpov et al. (2020). Yet the problem batched bandit problem with Lipschitz reward remains unsolved.

The Lipschitz bandit problem is important in its own stand. The Lipschitz bandit problem was introduced as "continuum-armed bandits" (Agrawal, 1995a), where the arm space is a compact interval. Along this line, bandits that are Lipschitz (or Hölder) continuous have been studied. For this problem, Kleinberg (2005) proves a $\Omega(T^{2/3})$ lower bound and introduced a matching algorithm. Under extra conditions on top of Lipschitzness, regret rate of $\widetilde{\mathcal{O}}(T^{1/2})$ was achieved (Cope, 2009; Auer et al., 2007). For general (doubling) metric spaces, the Zooming bandit algorithm (Kleinberg et al., 2008) and the Hierarchical Optimistic Optimization (HOO) algorithm (Bubeck et al., 2011a) were developed. In more recent years, some attention has been focused on Lipschitz bandit problems with certain extra structures. To name a few, Bubeck et al. (2011b) study Lipschitz bandits for differentiable rewards, which enables algorithms to run without explicitly knowing the Lipschitz constants. The idea of robust mean estimators (Bubeck et al., 2013; Bickel, 1965; Alon et al., 1999) was applied to the Lipschitz bandit problem to cope with heavy-tail rewards, leading to the development of a near-optimal algorithm for Lipschitz bandit with heavy-tailed reward (Lu et al., 2019). Lipschitz bandits where a clustering is used to infer the underlying metric, has been studied by Wanigasekara & Yu (2019). Contextual Lipschitz bandits have also been studied by Slivkins (2014) and Krishnamurthy et al. (2019). Yet all of the existing works for Lipschitz bandits assume that the reward sample is immediately observed after each arm pull, and none of them solve the batched Lipschitz bandit problem.

2 Settings & Preliminaries

For a (batched) Lipschitz bandit problem, the arm set is a compact doubling metric space $(\mathcal{A}, d_{\mathcal{A}})$. The expected reward $\mu : \mathcal{A} \to \mathbb{R}$ is 1-Lipschitz with respect to the metric $d_{\mathcal{A}}$, that is

$$|\mu(x_1) - \mu(x_2)| \leq d_{\mathcal{A}}(x_1, x_2), \quad \forall x_1, x_2 \in \mathcal{A}.$$

In each round $t \leq T$, the learning agent pulls an arm $x_t \in \mathcal{A}$ that yields a reward sample $y_t = \mu(x_t) + \epsilon_t$, where ϵ_t is a mean-zero independent sub-Gaussian noise. Without loss of generality, we assume that $\epsilon_t \sim \mathcal{N}(0, 1)$, since generalizations to other sub-Gaussian distributions are not hard.

For a batched problem, the time horizon T is divided into B batches, with respect to a grid $\mathcal{T} = \{t_0, \cdots, t_B\}$. In this environment, the reward sample for time t cannot be observed until the current batch is finished. Formally, the reward sample y_t can not be observed until $t_m = \min_s\{s \in \mathcal{T} : s \geq t\}$.

Similar to most bandit learning problems. the agent's goal is to minimize the regret:

$$R(T) = \sum_{t=1}^{T} \left(\mu^* - \mu(x_t)\right),$$

where μ^* denotes $\max_{x \in \mathcal{A}} \mu(x)$. For simplicity, we define $\Delta_x = \mu^* - \mu(x)$ (called optimality gap of x) for all $x \in \mathcal{A}$.

2.1 Doubling Metric Spaces and the $([0, 1]^d, \|\cdot\|_\infty)$ Metric Space

By the Assouad's embedding theorem (Assouad, 1983), the (compact) doubling metric space $(\mathcal{A}, d_{\mathcal{A}})$ can be embedded into a Euclidean space with some distortion of the metric; See (Wang & Rudin, 2020) for more discussions in a machine learning context. Due to existence of such embedding, the metric space $([0, 1]^d, \|\cdot\|_\infty)$, where metric balls are cubes, is sufficient for the purpose of our paper. For the rest of the paper, we will use hypercubes in algorithm design for simplicity, while our algorithmic idea generalizes to other doubling metric spaces.

2.2 Zooming Numbers and Zooming Dimensions

An important concept for bandit problems in metric spaces is the zooming number and the zooming dimension (Kleinberg et al., 2008; Bubeck et al., 2009; Slivkins, 2014), which we discuss now.

Define the set of r-optimal arms as $S(r) = \{x \in \mathcal{A} : \Delta_x \le r\}$. For any $r = 2^{-i}$, the decision space $[0,1]^d$ can be equally divided into 2^{di} cubes with edge length r, which we call *standard cubes* (more commonly referred to as dyadic cubes). The r-zooming number is defined as

$$N_r := \# \{C : C \text{ is a standard cube with edge length } r \text{ and } C \subset S(16r)\}.$$

In words, N_r is the r-packing number of the set $S(16r)$ in terms of standard cubes. The zooming dimension is then defined as

$$d_z := \min_d \{d : N_r \le r^{-d}, \quad \forall r \in (0,1]\}.$$

Previously, seminal works (Kleinberg et al., 2008; Slivkins, 2014; Bubeck et al., 2011a) show that the optimal regret bound for traditionally Lipschitz bandits, where the reward observations are immediately observable after each arm pull, is

$$\mathbb{E}[R(T)] \lesssim R_z(T) \triangleq \inf_{r_0} \left\{ 16 r_0 T + 256 \sum_{r=2^{-i}, r \ge r_0} \frac{N_r}{r} \log T \right\} \tag{1}$$

in terms of zooming number, and

$$\mathbb{E}[R(T)] \lesssim T^{1-\frac{1}{d_z+2}} \log T$$

in terms of zooming dimension.

This paper is organized as follows. In section 3, we introduce the BLiN algorithm and give a visual illustration of the algorithm procedure. In section 4, we prove that BLiN achieves the optimal upper bound using only $\mathcal{O}\left(\frac{\log T}{d_z}\right)$ batches. Section 5 provides regret lower bounds for batched Lipschitz bandit problems. An experimental result is presented in Section 6.

3 Algorithm

In a batched bandit environment, the agent's knowledge about the environment does not accumulate within each batch, since the reward samples can only be collected at the end of the batch. Due to this nature, there is no gain from changing strategies within each batch. This characteristic of the problem suggests a "uniform" type algorithm – we shall treat each step within the same batch equally. Following this intuition, in each batch, we uniformly play the remaining arms, and eliminate arms of low reward. Next we describe the uniform play rule and the arm elimination rule.

Uniform Play Rule: For each batch m, a collection of subsets of the arm space $\mathcal{A}_m = \{C_{m,1}, C_{m,2}, \cdots, C_{m,|\mathcal{A}_m|}\}$is constructed. This construction $\mathcal{A}_m$ consists of cubes, and all cubes in $\mathcal{A}_m$ have the same edge length $r_m = 2^{-(m-1)}$. We will detail the construction of $\mathcal{A}_m$ when we describe the arm elimination rule.

During batch m, each cube in $\mathcal{A}_m$ is played $n_m := \frac{16 \log T}{r_m^2}$ times, where T is the total time horizon. More specifically, within each $C \in \mathcal{A}_m$, arms $x_{C,1}, x_{C,2}, \cdots, x_{C,n_m} \in C$ are played[1]. The reward samples $\{y_{C,1}, y_{C,2}, \cdots, y_{C,n_m}\}_{C \in \mathcal{A}_m}$ corresponding to $\{x_{C,1}, x_{C,2}, \cdots, x_{C,n_m}\}_{C \in \mathcal{A}_m}$ will be collected at the end of the this batch. Once the reward feedback is collected, we can eliminate arms of low reward.

Arm Elimination Rule: At the end of batch m, information from the arm pulls is collected, and we estimate the reward of each $C \in \mathcal{A}_m$ by $\widehat{\mu}_m(C) = \frac{1}{n_m} \sum_{i=1}^{n_m} y_{C,i}$. Cubes of low estimated reward are then eliminated, according to the following rule:

- A cube $C \in \mathcal{A}_m$ is removed if $\widehat{\mu}_m^{\max} - \widehat{\mu}_m(C) \ge 4 r_m$, where $\widehat{\mu}_m^{\max} := \max_{C \in \mathcal{A}_m} \widehat{\mu}_m(C)$.

After necessary removal of "bad cubes", each cube in $\mathcal{A}_m$ that survives the elimination is divided into 2^d subcubes of edge length $r_m/2$. These cubes (of edge length $r_m/2$) are collected to construct $\mathcal{A}_{m+1}$, and the learning process moves on to the next batch.

Remark 1. *Note that only samples from batch m is used for computing $\widehat{\mu}_m$. This algorithm design is sufficient to achieve theoretically optimal regret bound. Yet in practice, one can well keep all historical samples.*

The learning process is summarized in Algorithm 1.

[1] One can arbitrarily play $x_{C,1}, x_{C,2}, \cdots, x_{C,n_m}$ as long as $x_{C,i} \in C$ for all i.

Algorithm 1 Batched Lipschitz Narrowing (BLiN)

Input. Arm set $\mathcal{A} = [0,1]^d$; time horizon T.
Initialization $\mathcal{A}_1 = \{[0,1]^d\}$; edge length $r_1 = 1$; An upper bound of number of batches B_0 (default value $B_0 = T$); The first grid point $t_0 = 0$.
for $m = 1, 2, \cdots, B_0$ **do**
  Compute $n_m = \frac{16 \log T}{r_m^2}$.
  Compute the next grid point $t_m = t_{m-1} + n_m \cdot |\mathcal{A}_m|$.
  For each cube $C \in \mathcal{A}_m$, play arms $x_{C,1}, \cdots x_{C,n_m}$ from C, and compute the average payoff $\widehat{\mu}_m(C) = \frac{\sum_{i=1}^{n_m} y_{C,i}}{n_m}$. Find $\widehat{\mu}_m^{max} = \max_{C \in \mathcal{A}_m} \widehat{\mu}(C)$.
  For each cube $C \in \mathcal{A}_m$, eliminate C if $\widehat{\mu}_m^{max} - \widehat{\mu}_m(C) > 4r_m$.
  Equally partition each remaining cube in $\mathcal{A}_m$ to 2^d subcubes and define $\mathcal{A}_{m+1}$ as the collection of these subcubes.
  Define $r_{m+1} = r_m/2$, and repeat.
  / During the algorithm execution, we immediately exits the for-loop terminates once all T pulls are exhausted. */*
  / In practice, we can choose a large B_0 so that all T pulls are exhausted within the for-loop. */*
end for
Cleanup: Arbitrarily play the remaining arms until all T steps are used.

A visualization of a BLiN run is in Figure 1. In the i-th subgraph, the white cubes are those remaining after the i-th batch. In this experiment, we set $\mathcal{A} = [0,1]^2$, and the optimal arm is $x^* = (0.8, 0.7)$. Note that x^* is not eliminated during the game. More detailed description and results of this experiment are presented in Section 6.

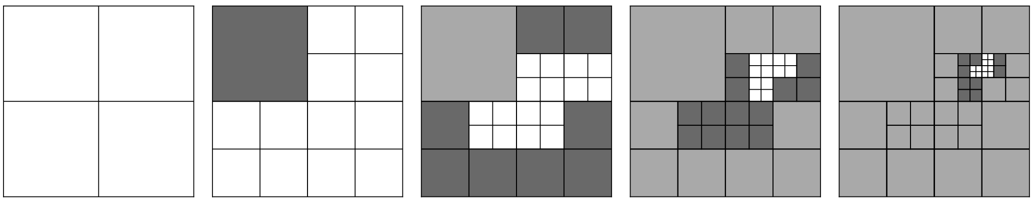


Figure 1: Partition and elimination process of BLiN (Algorithm 1). The i-th subfigure shows the pattern after the i-th batch for $1 \le i \le 5$. Shaded cubes are those eliminated, while darkgray ones are eliminated at the current batch. For the total time horizon $T = 80000$, BLiN needs only 6 batches.

4 REGRET ANALYSIS OF ALGORITHM 1

In this section, we provide regret analysis for Algorithm 1. The highlight of the finding is that $2 + \frac{\log T}{d_z+2}$ batches are sufficient to achieve optimal regret rate of $\widetilde{\mathcal{O}}\left(T^{\frac{d_z+1}{d_z+2}}\right)$, as summarized in Theorem 1.

Theorem 1. *With probability exceeding $1 - \frac{2}{T^6}$, the T-step total regret $R(T)$ of Algorithm 1 satisfies*

$$R(T) \le 528 T^{\frac{d_z+1}{d_z+2}} \log T, \tag{2}$$

where d_z is the zooming dimension of the problem instance. In addition, Algorithm 1 only needs $2 + \frac{\log_2 T}{d_z+2}$ batches to achieve this regret rate.

To prove Theorem 1, we first show that the estimator $\widehat{\mu}$ is concentrated to the true expected reward μ. This result is stated in Lemma 1 and can be quickly verified with the Hoeffding's inequality and Lipschitzness of the expected reward. In the following analysis, we let B_{stop} be the batch where the last arm is pulled in BLiN.

Lemma 1. *Define*

$$\mathcal{E} := \left\{ |\mu(x) - \widehat{\mu}_m(C)| \le r_m + \sqrt{\frac{16 \log T}{n_m}}, \quad \forall 1 \le m \le B_{\text{stop}},\ \forall C \in \mathcal{A}_m,\ \forall x \in C \right\}.$$

It holds that $\mathbb{P}(\mathcal{E}) \ge 1 - 2T^{-6}$.

The proof of Lemma 1 is in Appendix A. Next, we show that under event $\mathcal{E}$, the optimal arm is not removed (Lemma 2), and the cubes that survive elimination are of high reward (Lemma 3).

Lemma 2. *Under event $\mathcal{E}$, the optimal arm $x^* = \arg\max \mu(x)$ is not eliminated.*

Proof. We use C_m^* to denote the cube containing x^* in $\mathcal{A}_m$. Here we proof that C_m^* is not eliminated in round m. Under event $\mathcal{E}$, for any cube $C \in \mathcal{A}_m$ and $x \in C$, we have

$$\widehat{\mu}(C) - \widehat{\mu}(C_m^*) \leq \mu(x) + \sqrt{\frac{16 \log T}{n_m}} + r_m - \mu(x^*) + \sqrt{\frac{16 \log T}{n_m}} + r_m \leq 4r_m.$$

Then from the elimination rule, C_m^* is not eliminated. □

Lemma 3. *Under event $\mathcal{E}$, for any $1 \leq m \leq B_{\text{stop}}$, any $C \in \mathcal{A}_m$ and any $x \in C$, Δ_x satisfies*

$$\Delta_x \leq 16r_m. \tag{3}$$

Proof. For $m = 1$, (3) holds directly from the Lipschitzness of μ. For $m > 1$, let C_{m-1}^* be the cube in $\mathcal{A}_{m-1}$ such that $x^* \in C_{m-1}^*$. From Lemma 2, this cube C_{m-1}^* is well-defined under $\mathcal{E}$. For any cube $C \in \mathcal{A}_m$ and $x \in C$, it is obvious that x is also in the parent of C, which is denoted by C_{par}. Thus for any $x \in C$, it holds that

$$\Delta_x = \mu^* - \mu(x) \leq \widehat{\mu}_{m-1}(C_{m-1}^*) + \sqrt{\frac{16 \log T}{n_{m-1}}} + r_{m-1} - \widehat{\mu}_{m-1}(C_{par}) + \sqrt{\frac{16 \log T}{n_{m-1}}} + r_{m-1},$$

where the inequality uses Lemma 1.

Equality $\sqrt{\frac{16 \log T}{n_{m-1}}} = r_{m-1}$ gives that

$$\Delta_x \leq \widehat{\mu}_{m-1}(C_{m-1}^*) - \widehat{\mu}_{m-1}(C_{par}) + 4r_{m-1}.$$

Since the cube C_{par} is not eliminated, from the elimination rule we have

$$\widehat{\mu}_{m-1}(C_{m-1}^*) - \widehat{\mu}_{m-1}(C_{par}) \leq \widehat{\mu}_{m-1}^{\max} - \widehat{\mu}_{m-1}(C_{par}) \leq 4r_{m-1}.$$

Hence we have

$$\Delta_x \leq 8r_{m-1} = 16r_m.$$

□

With Lemmas 2 and 3, we can prove the regret guarantee for Algorithm 1. The regrouping argument is similar to the seminal ones by Kleinberg et al. (2008), but we can achieve such results with a smaller number of batches.

Lemma 4. *Under event $\mathcal{E}$, the T-step total regret $R(T)$ of Algorithm 1 satisfies*

$$R(T) \leq \inf_B \left\{ \sum_{m=1}^{B} N_{r_m} \cdot \frac{256 \log T}{r_m} + 16r_B T \right\}. \tag{4}$$

Proof. Lemma 3 shows that every cube $C \in \mathcal{A}_m$ is a subset of $S(16r_m)$. Thus from the definition of zooming number, we have

$$|\mathcal{A}_m| \leq N_{r_m}. \tag{5}$$

Fix a positive number B. Lemma 3 also implies that any arm played after batch B incurs a regret bounded by $16r_B$, since the cubes played after batch B have edge length no larger than r_B. Then the total regret occurs after the first B batch is bounded by $16r_B T$.

Thus the regret $R(T)$ can be bounded by

$$R(T) \leq \sum_{m=1}^{B} \sum_{C \in \mathcal{A}_m} \sum_{i=1}^{n_m} \Delta_{x_{C,i}} + 16r_B T, \tag{6}$$

where the first term bounds the regret in the first B batches in Algorithm 1, and the second term bounds the regret after the first B batches. If the algorithm stops at batch $B_{\text{stop}} < B$, we define $\mathcal{A}_m = \varnothing$ for any $B_{\text{stop}} < m \leq B$ and inequality (6) still holds.

By Lemma 3, we have $\Delta_{C,i} \leq 16r_m$ for all $C \in \mathcal{A}_m$. We can thus bound (6) by

$$\begin{aligned} R(T) &\leq \sum_{m=1}^{B} |\mathcal{A}_m| \cdot n_m \cdot 16r_m + 16r_B T \\ &\leq \sum_{m=1}^{B} N_{r_m} \cdot \frac{256 \log T}{r_m} + 16r_B T, \end{aligned}$$

where the second inequality uses (5) and $n_m = \frac{16 \log T}{r_m^2}$. Then by taking inf on all B, we have

$$R(T) \leq \inf_B \left\{ \sum_{m=1}^{B} N_{r_m} \cdot \frac{256 \log T}{r_m} + 16 r_B T \right\},$$

which finishes the proof of the Lemma. $\square$

Remark 2. *Note that the right hand of (4) equals to the optimal regret bound for traditional Lipschitz bandits in terms of zooming number*

$$R_z(T) = \inf_{r_0} \left\{ 16 r_0 T + 256 \sum_{r=2^{-i},\, r \geq r_0} \frac{N_r}{r} \log T \right\}.$$

Proof of Theorem 1. Since $r_m = 2^{-m+1}$ and $N_{r_m} \leq r_m^{-d_z} = 2^{(m-1)d_z}$, (4) yields that

$$R(T) \leq \inf_B \left\{ 256 \sum_{m=1}^{B} \left(2^{(m-1)(d_z+1)} \log T \right) + 16 \cdot 2^{-B+1} T \right\}.$$

By choosing $B^* = 1 + \frac{\log_2 T}{d_z+2}$, we have

$$\begin{aligned} R(T) \leq & 512 \cdot 2^{(B^*-1)(d_z+1)} \log T + 16 \cdot T \cdot 2^{-B^*+1} \\ \leq & 528 T^{\frac{d_z+1}{d_z+2}} \log T. \end{aligned}$$

The analysis in Theorem 1 implies that we can achieve the optimal regret rate $\widetilde{\mathcal{O}}\left(T^{\frac{d_z+1}{d_z+2}}\right)$ by letting the *for-loop* of Algorithm 1 run B^* times and finishing the remaining rounds in the *Cleanup* step. In other words, $B^* + 1$ batches are sufficient for Algorithm 1 to achieve the regret bound (2). $\square$

5 Lower Bounds

In this section, we present the lower bounds for the batched Lipschitz bandit problem. Our lower bounds depend on the number of batches B. When B is sufficiently large, our lower bounds match the lower bound for the vanilla Lipschitz bandit problem $\widetilde{\Theta}(R_z(T))$. More importantly, this dependency on B gives the minimal number of batches needed to achieve optimal regret rate.

The lower bound depends on the number of batches, as well as how the grid is determined. The grid can be static or adaptive. If the grid $\mathcal{T}$ is static, $\mathcal{T}$ is fixed and independent from the policy. If the grid is adaptive, every grid point $t_j \in \mathcal{T}$ can be chosen based on the choices and observations before t_{j-1}, and the determination of grid points is part of the policy.

We provide lower bounds for both static and adaptive grids, respectively in Theorem 2 and Theorem 3. The adaptive grid case is more difficult to analyse and more general. Therefore, we present the core part of the proof for the adaptive-grid lower bound in the main text. For static grid, we sketch its constructive proof in Theorem 2 in the main text and postpone the details to the Appendix. The proof is inspired by (Gao et al., 2019) and (Slivkins, 2014) but non-trivially extends both previous arguments.

Theorem 2 (Lower Bound for Static Grid)**.** *For B-batched Lipschitz bandit problem with time horizon T and any static grid $\mathcal{T}$, for any policy π, there exists a problem instance such that*

$$\mathbb{E}[R_T(\pi)] \geq c \cdot (\log T)^{-\frac{\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}} \cdot R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}},$$

where $c > 0$ is a numerical constant independent of B, T, π and $\mathcal{T}$, $R_z(T)$ is defined in (1), and d is the dimension of the arm space.

To prove Theorem 2, we first find $t_{k-1}, t_k \in \mathcal{T}$ such that

$$\frac{t_k}{t_{k-1}^{\frac{1}{d+2}}} \geq T^{\frac{1-\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}}. \tag{7}$$

Note that we can always find such a pair t_{k-1}, t_k since for any fixed B, $\min_{\mathcal{T}=\{t_0,\cdots,t_B\}} \max_{k'} \frac{t_{k'}}{t_{k'-1}^{\frac{1}{d+2}}} \geq T^{\frac{1-\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}}$.

Then we construct a set of problem instances that is difficult to distinguish. Let $r_k = \frac{1}{t_{k-1}^{\frac{1}{d+2}}}$, $M_k := t_{k-1} r_k^2 = \frac{1}{r_k^d}$, and $\mathcal{U} = \{u_1, \cdots, u_{M_k}\}$ be an arm-set such that $d_{\mathcal{A}}(u_i, u_j) \geq r_k$ for any $i \neq j$. Then we consider a set of problem instances $\mathcal{I} = \{I_1, \cdots, I_{M_k}\}$. The expected reward for I_1 is defined as

$$\mu_1(x) = \begin{cases} \frac{3}{4} r_k, & x = u_1, \\ \frac{5}{8} r_k, & x = u_j,\ j \neq 1, \\ \max\left\{\frac{r_k}{2}, \max_{u \in \mathcal{U}} \{\mu_1(u) - d_{\mathcal{A}}(x, u)\}\right\}, & \text{otherwise.} \end{cases}$$

For $2 \leq i \leq M_k$, the expected reward for I_i is defined as

$$\mu_i(x) = \begin{cases} \frac{3}{4} r_k, & x = u_1, \\ \frac{7}{8} r_k, & x = u_i, \\ \frac{5}{8} r_k, & x = u_j,\ j \neq 1 \text{ and } j \neq i, \\ \max\left\{\frac{r_k}{2}, \max_{u \in \mathcal{U}} \{\mu_i(u) - d_{\mathcal{A}}(x, u)\}\right\}, & \text{otherwise.} \end{cases}$$

Instance set $\mathcal{I}$ is based on index k which satisfies (7). In our construction, instance I_i has a "peak" located at u_i with height $\frac{7}{8} r_k$ (except I_1). Then we prove that no algorithm can distinguish an instance in $\mathcal{I}$ from the others in the first $(k-1)$ batches, so the total regret is at least $r_k t_k$, which gives the lower bound we need.

As a result of Theorem 2, we can derive the minimal number of batches needed to achieve optimal regret rate for Lipschitz bandit problem, which is stated in Corollary 1.

Corollary 1. *For a B-batched Lipschitz bandit problem with time horizon T with static grid, at least $\Omega(\frac{\log \log T}{\log d})$ batches are needed to achieve the regret rate $\widetilde{\Theta}(R_z(T))$.*

The detailed proof of Corollary 1 is deferred to Appendix.

Theorem 3 (Lower Bound for Adaptive Grid)**.** *For a B-batched Lipschitz bandit problem with time horizon T, for any adaptive grid $\mathcal{T}$ and any policy π, there exists an instance such that*

$$\mathbb{E}\left[R_T(\pi)\right] \geq c \frac{1}{B^2} (\log T)^{-\frac{\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}} R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}},$$

where $c > 0$ is a numerical constant independent of B, T, π and $\mathcal{T}$, $R_z(T)$ is defined in (1), and d is the dimension of the arm space.

Proof. In this proof, we construct a series of 'worlds' (sets of problem instances) based on sequences $\{r_j\}$, $\{M_j\}$ and fixed grid $\{T_j\}$, which are defined below. In each world, we construct a set of instances using the needle-in-the-haystack technique. After the policy is given, we can find a world such that the worst-case regret of the policy in this world is lower bounded by $\widetilde{\Omega}\left(R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}}\right)$.

To prove this result, we first define the following fixed grid $\mathcal{T} = \{T_0, T_1, \cdots, T_B\}$, where

$$T_j = T^{\frac{1-\varepsilon^j}{1-\varepsilon^B}},$$

and $\varepsilon = \frac{1}{d+2}$. We also define $r_j = \frac{1}{T_{j-1}^{\varepsilon} B}$ and $M_j = \frac{1}{r_j^d}$. From the definition, we have

$$T_{j-1} r_j^2 = \frac{1}{r_j^d B^2} = \frac{M_j}{B^2}. \tag{8}$$

For $1 \leq j \leq B$, we can find sets of arms $\mathcal{U}_j = \{u_{j,1}, \cdots, u_{j,M_j}\}$ such that (a) $d_{\mathcal{A}}(u_{j,m}, u_{j,n}) \geq r_j$ for any $m \neq n$, and (b) $u_{1,M_1} = \cdots = u_{B,M_B}$. Then we construct problem instances based on arm sets $\mathcal{U}_1, \cdots, \mathcal{U}_B$.

Now we construct B different worlds, denoted by $\mathcal{I}_1, \cdots, \mathcal{I}_B$. For $1 \le j \le B-1$, $\mathcal{I}_j = \{I_{j,k}\}_{k=1}^{M_j-1}$, and the expected reward for $I_{j,k}$ is defined as

$$\mu_{j,k}(x) = \begin{cases} \frac{r_1}{2} + \frac{r_j}{16} + \frac{r_B}{16}, & x = u_{j,k}, \\ \frac{r_1}{2} + \frac{r_B}{16}, & x = u_{j,M_j}, \\ \max\left\{\frac{r_1}{2}, \max_{u \in \mathcal{U}_j} \{\mu_{j,k}(u) - d_{\mathcal{A}}(x,u)\}\right\}, & \text{otherwise.} \end{cases} \tag{9}$$

For $j = B$, $\mathcal{I}_B = \{I_B\}$. The expected reward for I_B is defined as

$$\mu_B(x) = \begin{cases} \frac{r_1}{2} + \frac{r_B}{16}, & x = u_{B,M_B}, \\ \max\left\{\frac{r_1}{2}, \mu_B(u_{B,M_B}) - d_{\mathcal{A}}(x, u_{B,M_B})\right\}, & \text{otherwise.} \end{cases} \tag{10}$$

For all arm pulls in all problem instances, an Gaussian noise sampled from $\mathcal{N}(0,1)$ is added to the observed reward. This noise corruption is independent from all other randomness.

To link the fixed worlds $\{\mathcal{I}_1, \cdots, \mathcal{I}_B\}$ to the adaptive grid setting, we first note that for any adaptive grid $\mathcal{T} = \{t_1, \cdots, t_B\}$, there exists an index j such that in world $\mathcal{I}_j$, $(t_{j-1}, t_j]$ is sufficiently large. More formally, for each $j \in [B]$, we define the event $A_j = \{t_{j-1} < T_{j-1},\ t_j \ge T_j\}$ and the following quantities

$$p_j := \frac{1}{M_j - 1} \sum_{k=1}^{M_j-1} \mathbb{P}_{j,k}(A_j)$$

for $j \le B-1$ and

$$p_B := \mathbb{P}_B(A_B),$$

where $\mathbb{P}_{j,k}(A_j)$ denotes the probability of the event A_j under instance $I_{j,k}$ and policy π. For these quantities, we have the following lemma.

Lemma 5. *For any adaptive grid $\mathcal{T}$ and policy π, it holds that $\sum_{j=1}^{B} p_j \ge \frac{7}{8}$.*

Lemma 5 implies that there exists some j such that $p_j > \frac{7}{8B}$. This Lemma is detailedly proved in the Appendix.

Next we proceed with the case where $p_j > \frac{7}{8B}$ for some $j \le B-1$. The case for $j = B$ can be proved analogously. In world $\mathcal{I}_j$, we use similar method to Theorem 2 to construct a series of instance sets that are difficult to differentiate. More precisely, for any $1 \le k \le M_j - 1$, we construct a set of problem instances $\mathcal{I}_{j,k} = (I_{j,k,l})_{1 \le l \le M_j}$ based on the constructions in (9) and (10). For $l \ne k$, the expected reward of $I_{j,k,l}$ is defined as

$$\mu_{j,k,l}(x) = \begin{cases} \mu_{j,k}(x) + \frac{3r_j}{16}, & x = u_{j,l}, \\ \mu_{j,k}(x), & x \in \mathcal{U}_j \text{ and } x \ne u_{j,l}, \\ \max\left\{\frac{r_1}{2}, \max_{u \in \mathcal{U}_j} \{\mu_{j,k,l}(u) - d_{\mathcal{A}}(x,u)\}\right\}, & otherwise. \end{cases}$$

For $l = k$, we let $\mu_{j,k,k} = \mu_{j,k}$ where $\mu_{j,k}$ is defined as in (9). For all arm pulls in all problem instances, an Gaussian noise sampled from $\mathcal{N}(0,1)$ is added to the observed reward. This noise corruption is independent from all other randomness. Based on this construction, the following lemma gives the lower bound of expected regret, whose proof is in the Appendix.

Lemma 6. *For any adaptive grid $\mathcal{T}$ and policy π, if index j satisfies $p_j \ge \frac{7}{8B}$, then there exists a problem instance $I \in \cup_{k \le M_j - 1} \mathcal{I}_{j,k}$ such that*

$$\mathbb{E}\left[R_T(\pi)\right] \ge \frac{1}{256B^2} T^{\frac{1-\varepsilon}{1-\varepsilon^B}}.$$

Since $N_r \le r^{-d}$ holds for all instances,

$$\sum_{r=2^{-i},\ r \ge r_0} \frac{N_r}{r} \log T \le 2r_0^{-d-1} \log T.$$

Then we have

$$R_z(T) \le \inf_{r_0} \left\{16 r_0 T + 512 \frac{1}{r_0^{d+1}} \log T\right\} \le 512 (\log T)^{\frac{1}{d+2}} T^{1 - \frac{1}{d+2}}.$$

Consequently, for any policy π and adaptive grid $\mathcal{T}$, Lemma 5, Lemma 6 and the above inequality shows that there exists an instance I such that

$$\mathbb{E}\left[R_T(\pi)\right] \geq \frac{1}{256B^2} T^{\frac{1-\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}}$$
$$\geq \frac{1}{256 \cdot 512^{\frac{1}{1-\frac{1}{d+2}}}} \frac{1}{B^2} (\log T)^{-\frac{\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}} R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}},$$

which concludes the proof. □

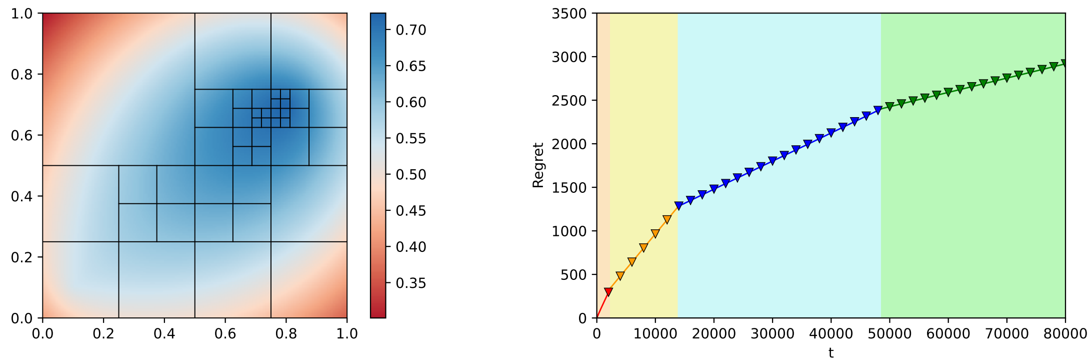
Similar to Corollary 1, we can prove that at least $\Omega(\frac{\log\log T}{\log d})$ batches are needed to achieve optimal regret rate. This result is formally summarized in Corollary 2.

Corollary 2. *For a B-batched Lipschitz bandit problem with time horizon T with adaptive grid, at least $\Omega(\frac{\log\log T}{\log d})$ batches are needed to achieve the regret bound $\widetilde{\Theta}(R_z(T))$.*

6 Experiments

In this section, we present numerical studies of BLiN. In the experiments, we use the arm space $\mathcal{A} = [0,1]^2$ and the expected reward function $\mu(x) = 1 - \frac{1}{2}\|x - x_1\|_2 - \frac{3}{10}\|x - x_2\|_2$, where $x_1 = (0.8, 0.7)$ and $x_2 = (0.1, 0.1)$. The landscape of μ is shown in Figure 2(a).

We let the time horizon $T = 80000$, and report the accumulated regret in Figure 2(b). The regret curve is sublinear, which agrees with the regret bound (2). Besides, different background colors in Figure 2(b) represent different batches. For the total time horizon $T = 80000$, BLiN only needs 6 batches (the first two batches are too small and are combined with the third batch in the visulization).



(a) Partition (b) Regret

Figure 2: Resulting partition and regret of BLiN (Algorithm 1). In Figure 2(a), we show the resulting partition of Algorithm 1. The background color denotes the true value of expected reward μ, and blue means high values. The figure shows that the partition is finer for larger values of μ. In Figure 2(b), we show accumulated regret of BLiN. In the figure, different background colors represent different batches. For the total time horizon $T = 80000$, BLiN needs only 6 batches (the first two batches are too small and are combined with the third batch in the plot).

7 Conclusion

In this paper, we study the Batched Lipschitz Bandit problem, and propose the BLiN algorithm as a solution. We prove that BLiN only need $\mathcal{O}\left(\frac{\log T}{d_z}\right)$ batches to achieve the optimal regret rate of best previous Lipschitz bandit algorithms (Kleinberg et al., 2008; Bubeck et al., 2009) that need T batches. This improvement in number of the batches significantly saves data communication costs. We also provide complexity analysis for this problem. We show that if the observations are collected in B batches, then for any policy π, there exists a problem instance such that the expected regret is at best $\widetilde{\Omega}\left(R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}}\right)$, where $R_z(T)$ is the regret lower bound for vanilla Lipschitz bandits that depends on the zooming dimension d_z. This results implies that $\Omega(\frac{\log\log T}{\log d})$ batches are necessary to achieve the optimal regret rate.

REFERENCES

Arpit Agarwal, Shivani Agarwal, Sepehr Assadi, and Sanjeev Khanna. Learning with limited rounds of adaptivity: Coin tossing, multi-armed bandits, and ranking from pairwise comparisons. In *Conference on Learning Theory*, pp. 39–75. PMLR, 2017.

Rajeev Agrawal. The continuum-armed bandit problem. *SIAM Journal on Control and Optimization*, 33(6):1926–1951, 1995a.

Rajeev Agrawal. Sample mean based index policies by $\mathcal{O}(\log n)$ regret for the multi-armed bandit problem. *Advances in Applied Probability*, 27(4):1054–1078, 1995b.

Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *Journal of Computer and System Sciences*, 58(1):137–147, 1999.

Patrice Assouad. Plongements Lipschitziens dans $\mathbb{R}^n$. *Bulletin de la Société Mathématique de France*, 111:429–448, 1983.

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

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002b.

Peter Auer, Ronald Ortner, and Csaba Szepesvári. Improved rates for the stochastic continuum-armed bandit problem. In *Conference on Computational Learning Theory*, pp. 454–468. Springer, 2007.

Peter J. Bickel. On some robust estimates of location. *The Annals of Mathematical Statistics*, pp. 847–858, 1965.

Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. Online optimization in $\mathcal{X}$-armed bandits. *Advances in Neural Information Processing Systems*, 22:201–208, 2009.

Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. $\mathcal{X}$-armed bandits. *Journal of Machine Learning Research*, 12(5):1655–1695, 2011a.

Sébastien Bubeck, Gilles Stoltz, and Jia Yuan Yu. Lipschitz bandits without the Lipschitz constant. In *International Conference on Algorithmic Learning Theory*, pp. 144–158. Springer, 2011b.

Sébastien Bubeck, Nicolo Cesa-Bianchi, and Gábor Lugosi. Bandits with heavy tail. *IEEE Transactions on Information Theory*, 59(11):7711–7717, 2013.

Nicolo Cesa-Bianchi, Ofer Dekel, and Ohad Shamir. Online learning with switching costs and other adaptive adversaries. *Advances in Neural Information Processing Systems*, 26:1160–1168, 2013.

Eric W. Cope. Regret and convergence bounds for a class of continuum-armed bandit problems. *IEEE Transactions on Automatic Control*, 54(6):1243–1253, 2009.

Eyal Even-Dar, Shie Mannor, Yishay Mansour, and Sridhar Mahadevan. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7(6):1079–1105, 2006.

Zijun Gao, Yanjun Han, Zhimei Ren, and Zhengqing Zhou. Batched Multi-armed Bandits Problem. *Advances in Neural Information Processing Systems*, 32:503–513, 2019.

Kwang-Sung Jun, Kevin Jamieson, Robert Nowak, and Xiaojin Zhu. Top arm identification in multi-armed bandits with batch arm pulls. In *Artificial Intelligence and Statistics*, pp. 139–148. PMLR, 2016.

Nikolai Karpov, Qin Zhang, and Yuan Zhou. Collaborative top distribution identifications with limited interaction. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 160–171. IEEE, 2020.

Robert Kleinberg. Nearly tight bounds for the continuum-armed bandit problem. *Advances in Neural Information Processing Systems*, 18:697–704, 2005.

Robert Kleinberg, Aleksandrs Slivkins, and Eli Upfal. Multi-armed bandits in metric spaces. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp. 681–690, 2008.

Akshay Krishnamurthy, John Langford, Aleksandrs Slivkins, and Chicheng Zhang. Contextual bandits with continuous actions: Smoothing, zooming, and adapting. In *Conference on Learning Theory*, pp. 2025–2027. PMLR, 2019.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.

Shiyin Lu, Guanghui Wang, Yao Hu, and Lijun Zhang. Optimal algorithms for Lipschitz bandits with heavy-tailed rewards. In *International Conference on Machine Learning*, pp. 4154–4163, 2019.

Vianney Perchet and Philippe Rigollet. The multi-armed bandit problem with covariates. *The Annals of Statistics*, 41(2): 693–721, 2013.

Vianney Perchet, Philippe Rigollet, Sylvain Chassang, and Erik Snowberg. Batched bandit problems. *The Annals of Statistics*, 44(2):660–681, 2016.

Stuart J. Pocock. Group sequential methods in the design and analysis of clinical trials. *Biometrika*, 64(2):191–199, 1977.

Yufei Ruan, Jiaqi Yang, and Yuan Zhou. Linear bandits with limited adaptivity and learning distributional optimal design. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 74–87, 2021.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

Sean Sinclair, Tianyu Wang, Gauri Jain, Siddhartha Banerjee, and Christina Yu. Adaptive discretization for model-based reinforcement learning. *Advances in Neural Information Processing Systems*, 33:3858–3871, 2020.

Aleksandrs Slivkins. Contextual bandits with similarity information. *Journal of Machine Learning Research*, 15(1): 2533–2568, 2014.

Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Chao Tao, Qin Zhang, and Yuan Zhou. Collaborative learning with limited interaction: Tight bounds for distributed exploration in multi-armed bandits. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 126–146. IEEE, 2019.

William 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.

Tianyu Wang and Cynthia Rudin. Bandits for BMO functions. In *International Conference on Machine Learning*, pp. 9996–10006. PMLR, 2020.

Nirandika Wanigasekara and Christina Yu. Nonparametric contextual bandits in an unknown metric space. In *Advances in Neural Information Processing Systems*, volume 32, pp. 14684–14694, 2019.

A PROOF OF LEMMA 1

Lemma 1. Define

$$\mathcal{E} := \left\{ |\mu(x) - \widehat{\mu}_m(C)| \le r_m + \sqrt{\frac{16 \log T}{n_m}}, \quad \forall 1 \le m \le B_{\text{stop}},\ \forall C \in \mathcal{A}_m,\ \forall x \in C \right\}.$$

It holds that $\mathbb{P}\left(\mathcal{E}\right) \ge 1 - 2T^{-6}$.

Proof. Fix a cube $C \in \mathcal{A}_m$. Recall the average payoff of cube $C \in \mathcal{A}_m$ is defined as

$$\widehat{\mu}_m(C) = \frac{\sum_{i=1}^{n_m} y_{C,i}}{n_m}.$$

We also have

$$\mathbb{E}\left[\widehat{\mu}_m(C)\right] = \frac{\sum_{i=1}^{n_m} \mu(x_{C,i})}{n_m}.$$

Since $\widehat{\mu}_m(C) - \mathbb{E}\left[\widehat{\mu}_m(C)\right]$ obeys normal distribution $\mathcal{N}\left(0, \frac{1}{n_m}\right)$, Hoeffding inequality gives

$$\mathbb{P}\left(\left|\widehat{\mu}_m(C) - \mathbb{E}\left[\widehat{\mu}_m(C)\right]\right| \ge \sqrt{\frac{16 \log T}{n_m}} \right) \le \frac{2}{T^8}.$$

On the other hand, by Lipschitzness of μ, it is obvious that

$$\left|\mathbb{E}\left[\widehat{\mu}_m(C)\right] - \mu(x)\right| \le r_m, \quad \forall x \in C.$$

Consequently, we have

$$\mathbb{P}\left(\sup_{x \in C} |\mu(x) - \widehat{\mu}_m(C)| \le r_m + \sqrt{\frac{16 \log T}{n_m}} \right) \ge 1 - \frac{2}{T^8}.$$

From here, by Lemma F.1 in (Sinclair et al., 2020), we know the number of cubes can never exceed T. Thus by similar argument to Claim 4 in Slivkins (2014), taking a union bound finishes the proof. □

B PROOF OF THEOREM 2

Theorem 2. For B-batched Lipschitz bandit problem with time horizon T and any static grid $\mathcal{T} = \{t_1 \cdots, t_B\}$, for any policy π, there exists a problem instance such that

$$\mathbb{E}[R_T(\pi)] \ge c \cdot (\log T)^{-\frac{\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}} \cdot R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}},$$

where $c > 0$ is a numerical constant independent of B, T, π and $\mathcal{T}$, $R_z(T)$ is defined in (1), and d is the dimension of the arm space.

Proof. For any static grid $0 = t_0 < t_1 < \cdots < t_{B-1} < t_B = T$, there always exists an index $k \le B$ such that

$$\frac{t_k}{t_{k-1}^{\frac{1}{d+2}}} \ge T^{\frac{1-\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}}. \tag{11}$$

Let $r_k = \frac{1}{t_{k-1}^{\frac{1}{d+2}}}$ (then $t_{k-1} = \frac{1}{r_k^{d+2}}$), and $M_k := t_{k-1} r_k^2 = \frac{1}{r_k^d}$. We can find a set of arms $\mathcal{U} = \{u_1, \cdots, u_{M_k}\}$ such that $d_{\mathcal{A}}(u_i, u_j) \ge r_k$ for any $i \ne j$. Then we consider a set of problem instances $\mathcal{I} = \{I_1, \cdots, I_{M_k}\}$. The expected reward for I_1 is defined as

$$\mu_1(x) = \begin{cases} \frac{3}{4} r_k, & x = u_1, \\ \frac{5}{8} r_k, & x = u_j,\ j \ne 1, \\ \max\left\{ \frac{r_k}{2}, \max_{u \in \mathcal{U}} \{\mu_1(u) - d_{\mathcal{A}}(x, u)\} \right\}, & \textit{otherwise}. \end{cases} \tag{12}$$

For $2 \leq i \leq M_k$, the expected reward for I_i is defined as

$$\mu_i(x) = \begin{cases} \frac{3}{4} r_k, & x = u_1, \\ \frac{7}{8} r_k, & x = u_i, \\ \frac{5}{8} r_k, & x = u_j,\ j \neq 1 \textit{ and } j \neq i, \\ \max\left\{ \frac{r_k}{2}, \max_{u \in \mathcal{U}} \left\{ \mu_i(u) - d_{\mathcal{A}}(x, u) \right\} \right\}, & \textit{otherwise}. \end{cases} \tag{13}$$

For all arm pulls in all problem instances, an Gaussian noise sampled from $\mathcal{N}(0, 1)$ is added to the observed reward. This noise corruption is independent from all other randomness.

The lower bound of expected regret relies on the following lemma.

Lemma 7. *For any policy π, there exists a problem instance $I \in \mathcal{I}$ such that*

$$\mathbb{E}\left[R_T(\pi)\right] \geq \frac{r_k}{32} \cdot \sum_{j=1}^{B} (t_j - t_{j-1}) \exp\left\{ -\frac{t_{j-1} r_k^2}{32(M_k - 1)} \right\}.$$

Proof. Let $S_i = \mathbb{B}(u_i, \frac{3}{8} r_k)$ (the ball with center u_i and radius $\frac{3}{8} r_k$). It is easy to verify the following properties of construction (12) and (13):

1. For any $2 \leq i \leq M_k$, $\mu_i(x) = \mu_1(x)$ for any $x \in \mathcal{A} - S_i$;
2. For any $2 \leq i \leq M_k$, $\mu_1(x) \leq \mu_i(x) \leq \mu_1(x) + \frac{r_k}{4}$, for any $x \in S_i$;
3. For any $1 \leq i \leq M_k$, pulling an arm that is not in S_i incurs a regret at least $\frac{r_k}{8}$.

Let x_t denote the choices of policy π at time t, and y_t denote the reward. Additionally, for $t_{j-1} < t \leq t_j$, we define $\mathbb{P}_i^t$ as the distribution of sequence $\left(x_1, y_1, \cdots, x_{t_{j-1}}, y_{t_{j-1}}\right)$ under instance I_i and policy π. It holds that

$$\begin{aligned} \sup_{I \in \mathcal{I}} \mathbb{E} R_T(\pi) \geq & \frac{1}{M_k} \sum_{i=1}^{M_k} \mathbb{E}_{\mathbb{P}_i^T} \left[R_T(\pi) \right] \\ \geq & \frac{1}{M_k} \sum_{i=1}^{M_k} \sum_{t=1}^{T} \mathbb{E}_{\mathbb{P}_i^t} \left[R^t(\pi) \right] \\ \geq & \frac{r_k}{8} \sum_{t=1}^{T} \frac{1}{M_k} \sum_{i=1}^{M_k} \mathbb{P}_i^t (x_t \notin S_i), \end{aligned} \tag{14}$$

where $R^t(\pi)$ denotes the regret incurred by policy π at time t.

From our construction, it is easy to see that $S_i \cap S_j = \varnothing$, so we can construct a test Ψ such that $x_t \in S_i$ implies $\Psi = i$. Then from Lemma 9,

$$\frac{1}{M_k} \sum_{i=1}^{M_k} \mathbb{P}_i^t (x_t \notin S_i) \geq \frac{1}{M_k} \sum_{i=1}^{M_k} \mathbb{P}_i^t (\Psi \neq i) \geq \frac{1}{2M_k} \sum_{i=2}^{M_k} \exp\left\{ -D_{KL}\left(\mathbb{P}_1^t \| \mathbb{P}_i^t \right) \right\}.$$

Now we calculate $D_{KL}\left(\mathbb{P}_1^t \| \mathbb{P}_i^t\right)$. From the chain rule of KL-Divergence, we have

$$\begin{aligned}
D_{KL}\left(\mathbb{P}_1^t \| \mathbb{P}_i^t\right) =& D_{KL}\left(\mathbb{P}_1^t(x_1, y_1, \cdots, x_{t_{j-1}}, y_{t_{j-1}}) \| \mathbb{P}_i^t(x_1, y_1, \cdots, x_{t_{j-1}}, y_{t_{j-1}})\right) \\
=& D_{KL}\left(\mathbb{P}_1^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1}, x_{t_{j-1}}) \| \mathbb{P}_i^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1}, x_{t_{j-1}})\right) \\
&+ \mathbb{E}_{\mathbb{P}_1}\left(D_{KL}\left(\mathbb{P}_1^t(y_{t_{j-1}} | x_1, y_1, \cdots, x_{t_{j-1}}) \| \mathbb{P}_i^t(y_{t_{j-1}} | x_1, y_1, \cdots, x_{t_{j-1}})\right)\right) \\
\leq& D_{KL}\left(\mathbb{P}_1^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1}) \| \mathbb{P}_i^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1})\right) \\
&+ \mathbb{E}_{\mathbb{P}_1}\left(D_{KL}\left(\mathbb{P}_1^t(y_{t_{j-1}} | x_{t_{j-1}}) \| \mathbb{P}_i^t(y_{t_{j-1}} | x_{t_{j-1}})\right)\right), \qquad (15) \\
\leq& D_{KL}\left(\mathbb{P}_1^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1}) \| \mathbb{P}_i^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1})\right) \\
&+ \mathbb{E}_{\mathbb{P}_1}\left(D_{KL}\left(N(\mu_1(x_{t_{j-1}}), 1) \| N(\mu_i(x_{t_{j-1}}), 1)\right)\right), \qquad (16) \\
=& D_{KL}\left(\mathbb{P}_1^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1}) \| \mathbb{P}_i^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1})\right) \\
&+ \mathbb{E}_{\mathbb{P}_1}\left(\frac{1}{2}\left(\mu_1(x_{t_{j-1}}) - \mu_i(x_{t_{j-1}})\right)^2\right), \\
\leq& D_{KL}\left(\mathbb{P}_1^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1}) \| \mathbb{P}_i^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1})\right) \\
&+ \mathbb{E}_{\mathbb{P}_1}\left(\mathbf{1}_{\{x_{t_{j-1}} \in S_i\}} \cdot \frac{1}{2}\left(\frac{r_k}{4}\right)^2\right) \qquad (17) \\
=& D_{KL}\left(\mathbb{P}_1^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1}) \| \mathbb{P}_i^t(x_1, y_1, \cdots, x_{t_{j-1}-1}, y_{t_{j-1}-1})\right) \\
&+ \frac{r_k^2}{32} \cdot \mathbb{P}_1^t\left(x_{t_{j-1}} \in S_i\right), \qquad (18)
\end{aligned}$$

where (15) uses the non-negativity of KL-Divergence and the conditional independence of the reward, (16) uses that the rewards are corrupted by a standard normal noise, and (17) uses the first two properties of the construction.

From (18), we then decompose the KL-Divergence step by step and conclude that

$$D_{KL}\left(\mathbb{P}_1^t \| \mathbb{P}_i^t\right) \leq \frac{r_k^2}{32} \cdot \sum_{s \leq t_{j-1}} \mathbb{P}_1^t\left(x_s \in S_i\right) = \frac{r_k^2}{32} \mathbb{E}_{\mathbb{P}_1} \tau_i, \qquad (19)$$

where τ_i denotes the number of pulls of arms in S_i before the batch containing t. Then for all $t \in (t_{j-1}, t_j]$, we have

$$\begin{aligned}
\frac{1}{M_k} \sum_{i=1}^{M_k} \mathbb{P}_i^t(x_t \notin S_i) &\geq \frac{1}{2M_k} \sum_{i=2}^{M_k} \exp\left\{-\frac{r_k^2}{32} \mathbb{E}_{\mathbb{P}_1} \tau_i\right\} \\
&\geq \frac{M_k - 1}{2M_k} \exp\left\{-\frac{r_k^2}{32(M_k - 1)} \sum_{i=2}^{M_k} \mathbb{E}_{\mathbb{P}_1} \tau_i\right\} \qquad (20) \\
&\geq \frac{1}{4} \exp\left\{-\frac{r_k^2 t_{j-1}}{32(M_k - 1)}\right\}, \qquad (21)
\end{aligned}$$

where (20) uses the Jensen' inequality, and (21) uses the fact that $\sum_{i=2}^{M_k} \tau_i \leq t_{j-1}$. Finally, we substitute (21) to (14) to finish the proof. □

Since $M_k = t_{k-1} r_k^2$, the expected regret of π satisfies

$$\begin{aligned}
\mathbb{E}\left[R_T(\pi)\right] &\geq \frac{r_k}{32} \cdot \sum_{j=1}^{B} (t_j - t_{j-1}) \exp\left\{-\frac{t_{j-1} r_k^2}{32(M_k - 1)}\right\} \\
&\geq \frac{r_k}{32} \cdot \sum_{j=1}^{B} (t_j - t_{j-1}) \exp\left\{-\frac{t_{j-1} r_k^2}{16 M_k}\right\} \\
&\geq \frac{r_k}{32} \cdot \sum_{j=1}^{B} (t_j - t_{j-1}) \exp\left\{-\frac{t_{j-1}}{16 t_{k-1}}\right\}
\end{aligned}$$

on instance I in Lemma 7,

By omitting terms with $j > k$ in the above summation, we have

$$\begin{aligned}
\mathbb{E}[R_T(\pi)] &= \frac{r_k}{32} \cdot \sum_{j=1}^{B} (t_j - t_{j-1}) \exp\left\{-\frac{t_{j-1}}{16 t_{k-1}}\right\} \\
&\geq \frac{r_k}{32} \cdot \sum_{j=1}^{k} (t_j - t_{j-1}) \exp\left\{-\frac{1}{16}\right\} \\
&= \frac{1}{32 e^{\frac{1}{16}}} r_k t_k \\
&= \frac{1}{32 e^{\frac{1}{16}}} \cdot \frac{t_k}{t_{k-1}^{\frac{1}{d+2}}}.
\end{aligned}$$

Furthermore, from (11),

$$\mathbb{E}\left[R_T(\pi)\right] \geq \frac{1}{32 e^{\frac{1}{16}}} \cdot T^{\frac{1-\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}}. \tag{22}$$

Since $N_r \leq r^{-d}$ holds for all instances,

$$\sum_{r=2^{-i},\, r \geq r_0} \frac{N_r}{r} \log T \leq 2 r_0^{-d-1} \log T.$$

Then we have

$$\begin{aligned}
R_z(T) =& \inf_{r_0} 16 r_0 T + 256 \sum_{r=2^{-i},\, r \geq r_0} \frac{N_r}{r} \log T \\
\leq& \inf_{r_0} 16 r_0 T + 512 \frac{1}{r_0^{d+1}} \log T \\
=& 16 \cdot (32 \log T)^{\frac{1}{d+2}} T^{1-\frac{1}{d+2}} \\
\leq& 512 (\log T)^{\frac{1}{d+2}} T^{1-\frac{1}{d+2}}.
\end{aligned}$$

As a consequence, for instance I defined in Lemma 7,

$$\begin{aligned}
\mathbb{E}[R_T(\pi)] \geq& \frac{1}{32 e^{\frac{1}{16}}} \left(\frac{1}{512 (\log T)^{\frac{1}{d+2}}}\right)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}} \cdot R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}} \\
\geq& \frac{1}{32 e^{\frac{1}{16}} \cdot 512^{\frac{1}{1-\frac{1}{d+2}}}} \cdot (\log T)^{-\frac{\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}} \cdot R_z(T)^{\frac{1}{1-\left(\frac{1}{d+2}\right)^B}}.
\end{aligned}$$

Hence, the proof is completed. □

C Proof of Corollary 1

Corollary 1. For a B-batched Lipschitz bandit problem with time horizon T with static grid, at least $\Omega(\frac{\log\log T}{\log d})$ batches are needed to achieve the regret rate $\widetilde{\Theta}(R_z(T))$.

Proof. From (22), the expected regret is lower bounded by $\frac{1}{32 e^{\frac{1}{16}}} \cdot T^{\frac{1-\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}}$. When B is sufficiently large, the bound becomes $\frac{1}{32 e^{\frac{1}{16}}} \cdot T^{1-\frac{1}{d+2}}$. Here we seek for the minimum B such that

$$\frac{\frac{1}{32 e^{\frac{1}{16}}} \cdot T^{\frac{1-\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}}}{\frac{1}{32 e^{\frac{1}{16}}} \cdot T^{1-\frac{1}{d+2}}} \leq C \tag{23}$$

for some constant C.

Calculation shows that

$$\frac{\frac{1}{32e^{\frac{1}{16}}} \cdot T^{\frac{1-\frac{1}{d+2}}{1-\left(\frac{1}{d+2}\right)^B}}}{\frac{1}{32e^{\frac{1}{16}}} \cdot T^{1-\frac{1}{d+2}}} = \left(T^{\frac{d+1}{d+2}}\right)^{\frac{1}{(d+2)^B-1}}. \tag{24}$$

Substituting (24) to (23) and taking log on both sides yield that

$$\frac{d+1}{d+2} \cdot \frac{\log T}{(d+2)^B - 1} \leq \log C$$

and

$$(d+2)^B \geq \frac{d+1}{(d+2)\log C} \cdot \log T + 1.$$

Taking log on both sides again yields that

$$B \geq \frac{\log\left[\left(\frac{d+1}{(d+2)\log C}\right)\log T + 1\right]}{\log(d+2)}. \qquad \square$$

D Proof of Lemma 5

Lemma 5. For any adaptive grid $\mathcal{T}$ and policy π, it holds that $\sum_{j=1}^{B} p_j \geq \frac{7}{8}$.

Proof. For $1 \leq j \leq B-1$ and $1 \leq k \leq M_j - 1$, we define $S_{j,k} = \mathbb{B}(u_{j,k}, \frac{3}{8}r_j)$, which is the ball centered as $u_{j,k}$ with radius $\frac{3}{8}r_j$. It is easy to verify the following properties of our construction (9) and (10):

1. $\mu_{j,k}(x) = \mu_B(x)$ for any $x \notin S_{j,k}$;
2. $\mu_B(x) \leq \mu_{j,k}(x) \leq \mu_B(x) + \frac{r_j}{8}$, for any $x \in S_{j,k}$.

Let x_t denote the choices of policy π at time t, and y_t denote the reward. For $t_{j-1} < t \leq t_j$, we define $\mathbb{P}_{j,k}^t$ (resp. $\mathbb{P}_B^t$) as the distribution of sequence $\left(x_1, y_1, \cdots, x_{t_{j-1}}, y_{t_{j-1}}\right)$ under instance $I_{j,k}$ (resp. I_B) and policy π. Since event A_j can be completely described by the observations up to time T_{j-1} (A_j is an event in the σ-algebra where $\mathbb{P}_{j,k}^{T_{j-1}}$ and $\mathbb{P}_B^{T_{j-1}}$ are defined on), we can use the definition of total variation to get

$$|\mathbb{P}_B(A_j) - \mathbb{P}_{j,k}(A_j)| = |\mathbb{P}_B^{T_{j-1}}(A_j) - \mathbb{P}_{j,k}^{T_{j-1}}(A_j)| \leq TV\left(\mathbb{P}_B^{T_{j-1}}, \mathbb{P}_{j,k}^{T_{j-1}}\right).$$

For the total variation, we apply Lemma 8 to get

$$\frac{1}{M_j - 1}\sum_{k=1}^{M_j-1} TV\left(\mathbb{P}_B^{T_{j-1}}, \mathbb{P}_{j,k}^{T_{j-1}}\right) \leq \frac{1}{M_j - 1}\sum_{k=1}^{M_j-1}\sqrt{1 - \exp\left(-D_{KL}\left(\mathbb{P}_B^{T_{j-1}} \| \mathbb{P}_{j,k}^{T_{j-1}}\right)\right)}.$$

An argument similar to (19) yields that

$$D_{KL}\left(\mathbb{P}_B^{T_{j-1}} \| \mathbb{P}_{j,k}^{T_{j-1}}\right) \leq \frac{r_j^2}{128}\mathbb{E}_{\mathbb{P}_B}\tau_k,$$

where τ_k denotes the number of pulls which is in $S_{j,k}$ before the batch containing T_{j-1}. Combining the above two inequalities gives

$$\begin{aligned}
\frac{1}{M_j - 1}\sum_{k=1}^{M_j-1} TV\left(\mathbb{P}_B^{T_{j-1}}, \mathbb{P}_{j,k}^{T_{j-1}}\right) &\leq \frac{1}{M_j - 1}\sum_{k=1}^{M_j-1}\sqrt{1 - \exp\left(-\frac{r_j^2}{128}\mathbb{E}_{\mathbb{P}_B}\tau_k\right)} \\
&\leq \sqrt{1 - \exp\left(-\frac{r_j^2}{128(M_j-1)}\mathbb{E}_{\mathbb{P}_B}\left[\sum_{k=1}^{M_j-1}\tau_k\right]\right)} \qquad (25) \\
&\leq \sqrt{1 - \exp\left(-\frac{r_j^2 T_{j-1}}{128(M_j-1)}\right)} \qquad (26) \\
&\leq \sqrt{1 - \exp\left(-\frac{1}{64B^2}\right)} \qquad (27) \\
&\leq \frac{1}{8B},
\end{aligned}$$

where (25) uses Jensen's inequality, (26) uses the fact that $\sum_{k=1}^{M_j-1} \tau_k \leq T_{j-1}$, and (27) uses (8).

Plugging the above results implies that

$$|\mathbb{P}_B(A_j) - p_j| \leq \frac{1}{M_j - 1} \sum_{k=1}^{M_j-1} |\mathbb{P}_B(A_j) - \mathbb{P}_{j,k}(A_j)| \leq \frac{1}{8B}.$$

Since $\sum_{j=1}^{B} \mathbb{P}(A_j) \geq \mathbb{P}\left(\cup_{j=1}^{B} A_j\right) = 1$, it holds that

$$\sum_{j=1}^{B} p_j \geq \mathbb{P}_B(A_M) + \sum_{j=1}^{B-1} \left(\mathbb{P}_B(A_j) - \frac{1}{8B}\right) \geq \frac{7}{8}.$$

□

E Proof of Lemma 6

Lemma 6. For any adaptive grid $\mathcal{T}$ and policy π, if index j satisfies $p_j \geq \frac{7}{8B}$, then there exists a problem instance $I \in \cup_{k \leq M_j - 1} \mathcal{I}_{j,k}$ such that

$$\mathbb{E}\left[R_T(\pi)\right] \geq \frac{1}{256B^2} T^{\frac{1-\varepsilon}{1-\varepsilon^B}}.$$

Proof. We define $C_{j,k} = \mathbb{B}\left(u_{j,k}, \frac{r_j}{4}\right)$, and our construction $\mathcal{I}_{j,k}$ has the following properties:

1. For any $l \neq k$, $\mu_{j,k,l}(x) = \mu_{j,k,k}(x)$ for any $x \notin C_{j,l}$;
2. For any $l \neq k$, $\mu_{j,k,k}(x) \leq \mu_{j,k,l}(x) \leq \mu_{j,k,k}(x) + \frac{3r_j}{16}$ for any $x \in C_{j,l}$;
3. For any $1 \leq l \leq M_j$, under $I_{j,k,l}$, pulling an arm that is not in $C_{j,l}$ incurs a reward at least $\frac{r_j}{16}$.

Let x_t denote the choices of policy π at time t, and y_t denote the reward. For $t_{j-1} < t \leq t_j$, we define $\mathbb{P}^t_{j,k,l}$ as the distribution of sequence $\left(x_1, y_1, \cdots, x_{t_{j-1}}, y_{t_{j-1}}\right)$ under instance $I_{i,j,k}$ and policy π. From similar argument in (14), it holds that

$$\sup_{I \in \mathcal{I}_{j,k}} \mathbb{E}\left[R_T(\pi)\right] \geq \frac{r_j}{16} \sum_{t=1}^{T} \frac{1}{M_j} \sum_{l=1}^{M_j} \mathbb{P}^t_{j,k,l}(x_t \notin C_{j,l}). \tag{28}$$

From our construction, it is easy to see that $C_{j,k_1} \cap C_{j,k_2} = \varnothing$ for any $k_1 \neq k_2$, so we can construct a test Ψ such that $x_t \notin C_{j,k}$ implies $\Psi \neq k$. By Lemma 9 with a star graph on $[K]$ with center k, we have

$$\frac{1}{M_j} \sum_{l=1}^{M_j} \mathbb{P}^t_{j,k,l}(x_t \notin C_{j,l}) \geq \frac{1}{M_j} \sum_{l \neq k} \int \min\left\{d\mathbb{P}^t_{j,k,k}, d\mathbb{P}^t_{j,k,l}\right\}. \tag{29}$$

Combining (28) and (29) gives

$$\begin{aligned}
\sup_{I \in \mathcal{I}_{j,k}} \mathbb{E}\left[R_T(\pi)\right] &\geq \frac{r_j}{16} \sum_{t=1}^{T} \frac{1}{M_j} \sum_{l \neq k} \int \min\left\{d\mathbb{P}^t_{j,k,k}, d\mathbb{P}^t_{j,k,l}\right\} \\
&\geq \frac{r_j}{16} \sum_{t=1}^{T_j} \frac{1}{M_j} \sum_{l \neq k} \int \min\left\{d\mathbb{P}^t_{j,k,k}, d\mathbb{P}^t_{j,k,l}\right\} \\
&\geq \frac{r_j T_j}{16} \cdot \frac{1}{M_j} \sum_{l \neq k} \int \min\left\{d\mathbb{P}^{T_j}_{j,k,k}, d\mathbb{P}^{T_j}_{j,k,l}\right\} &(30) \\
&\geq \frac{r_j T_j}{16} \cdot \frac{1}{M_j} \sum_{l \neq k} \int_{A_j} \min\left\{d\mathbb{P}^{T_j}_{j,k,k}, d\mathbb{P}^{T_j}_{j,k,l}\right\} &(31) \\
&\geq \frac{r_j T_j}{16} \cdot \frac{1}{M_j} \sum_{l \neq k} \int_{A_j} \min\left\{d\mathbb{P}^{T_{j-1}}_{j,k,k}, d\mathbb{P}^{T_{j-1}}_{j,k,l}\right\}, &(32)
\end{aligned}$$

where (30) follows from data processing inequality of total variation and the equation $\int \min\{dP, dQ\} = 1 - TV(P,Q)$, (31) restricts the integration to event A_j, and (32) holds because the observations at time T_j are the same as those at time T_{j-1} under event A_j.

For the term $\int_{A_j} \min\left\{d\mathbb{P}_{j,k,k}^{T_{j-1}}, d\mathbb{P}_{j,k,l}^{T_{j-1}}\right\}$, it holds that

$$
\begin{aligned}
\int_{A_j} \min\left\{d\mathbb{P}_{j,k,k}^{T_{j-1}}, d\mathbb{P}_{j,k,l}^{T_{j-1}}\right\} &= \int_{A_j} \frac{d\mathbb{P}_{j,k,k}^{T_{j-1}} + d\mathbb{P}_{j,k,l}^{T_{j-1}} - \left|d\mathbb{P}_{j,k,k}^{T_{j-1}} - d\mathbb{P}_{j,k,l}^{T_{j-1}}\right|}{2} \\
&= \frac{\mathbb{P}_{j,k,k}^{T_{j-1}}(A_j) + \mathbb{P}_{j,k,l}^{T_{j-1}}(A_j)}{2} - \frac{1}{2}\int_{A_j}\left|d\mathbb{P}_{j,k,k}^{T_{j-1}} - d\mathbb{P}_{j,k,l}^{T_{j-1}}\right| \\
&\geq \left(\mathbb{P}_{j,k,k}^{T_{j-1}}(A_j) - \frac{1}{2}TV\left(\mathbb{P}_{j,k,k}^{T_{j-1}}, \mathbb{P}_{j,k,l}^{T_{j-1}}\right)\right) - TV\left(\mathbb{P}_{j,k,k}^{T_{j-1}}, \mathbb{P}_{j,k,l}^{T_{j-1}}\right) \quad (33) \\
&= \mathbb{P}_{j,k}(A_j) - \frac{3}{2}TV\left(\mathbb{P}_{j,k,k}^{T_{j-1}}, \mathbb{P}_{j,k,l}^{T_{j-1}}\right), \quad (34)
\end{aligned}
$$

where (33) uses the inequality $|\mathbb{P}(A) - \mathbb{Q}(A)| \leq TV(\mathbb{P}, \mathbb{Q})$, and (34) holds because $I_{j,k} = I_{j,k,k}$ and A_j can be determined by the observations up to T_{j-1}.

We use an argument similar to (19) to get

$$
D_{KL}\left(\mathbb{P}_{j,k,k}^{T_{j-1}} \| \mathbb{P}_{j,k,l}^{T_{j-1}}\right) \leq \frac{1}{2} \cdot \left(\frac{3r_j}{16}\right)^2 \mathbb{E}_{\mathbb{P}_{j,k}} \tau_l \leq \frac{r_j^2}{32} \mathbb{E}_{\mathbb{P}_{j,k}} \tau_l,
$$

where τ_l denotes the number of pulls which is in $S_{j,l}$ before the batch of time T_{j-1}. Then from Lemma 8, we have

$$
\begin{aligned}
\frac{1}{M_j}\sum_{l\neq k} TV\left(\mathbb{P}_{j,k,k}^{T_{j-1}}, \mathbb{P}_{j,k,l}^{T_{j-1}}\right) &\leq \frac{1}{M_j}\sum_{l\neq k}\sqrt{1 - \exp\left(-D_{KL}\left(\mathbb{P}_{j,k,k}^{T_{j-1}} \| \mathbb{P}_{j,k,l}^{T_{j-1}}\right)\right)} \\
&\leq \frac{1}{M_j}\sum_{l\neq k}\sqrt{1 - \exp\left(-\frac{r_j^2}{32}\mathbb{E}_{\mathbb{P}_{j,k}}\tau_l\right)} \\
&\leq \frac{M_j - 1}{M_j}\sqrt{1 - \exp\left(-\frac{r_j^2}{32(M_j-1)}\sum_{l\neq k}\mathbb{E}_{\mathbb{P}_{j,k}}\tau_l\right)} \\
&\leq \frac{M_j - 1}{M_j}\sqrt{1 - \exp\left(-\frac{r_j^2 T_{j-1}}{32(M_j-1)}\right)} \\
&\leq \frac{M_j - 1}{M_j}\sqrt{1 - \exp\left(-\frac{M_j}{32(M_j-1)B^2}\right)} \quad (35) \\
&\leq \frac{M_j - 1}{M_j}\sqrt{\frac{M_j}{32(M_j-1)B^2}} \\
&\leq \frac{1}{4B}, \quad (36)
\end{aligned}
$$

where (35) uses (8).

Combining (32), (34) and (36) yields that

$$
\begin{aligned}
\sup_{I\in\mathcal{I}_{j,k}} \mathbb{E}\left[R_T(\pi)\right] &\geq \frac{1}{16} r_j T_j \left(\frac{\mathbb{P}_{j,k}(A_j)}{2} - \frac{3}{8B}\right) \\
&\geq \frac{1}{16B} T^{\frac{1-\varepsilon}{1-\varepsilon^B}} \left(\frac{\mathbb{P}_{j,k}(A_j)}{2} - \frac{3}{8B}\right).
\end{aligned}
$$

This inequality holds for any $k \leq M_j - 1$. Averaging over k yields

$$\begin{aligned}
\sup_{I \in \cup_{k \leq M_j - 1} \mathcal{I}_{j,k}} \mathbb{E}\left[R_T(\pi)\right] &\geq \frac{1}{16B} T^{\frac{1-\varepsilon}{1-\varepsilon^B}} \left(\frac{1}{2(M_j - 1)} \sum_{k=1}^{M_j - 1} \mathbb{P}_{j,k}(A_j) - \frac{3}{8B} \right) \\
&\geq \frac{1}{16B} T^{\frac{1-\varepsilon}{1-\varepsilon^B}} \left(\frac{7}{16B} - \frac{3}{8B} \right) \qquad (37) \\
&\geq \frac{1}{256B^2} T^{\frac{1-\varepsilon}{1-\varepsilon^B}},
\end{aligned}$$

where (37) holds from $p_j \geq \frac{7}{8B}$. Hence, the proof of Lemma 6 is completed. $\square$

F AUXILIARY TECHNICAL TOOLS

Lemma 8 (Bretagnolle-Huber Inequality)**.** *Let P and Q be any probability measures on the same probability space. It holds that*

$$TV(P, Q) \leq \sqrt{1 - \exp\left(-D_{KL}(P \| Q)\right)} \leq 1 - \frac{1}{2} \exp\left(-D_{KL}(P \| Q)\right).$$

Lemma 9 (Gao et al. (2019))**.** *Let $Q_1, \cdots, Q_n$ be probability measures over a common probability space $(\Omega, \mathcal{F})$, and $\Psi : \Omega \to [n]$ be any measurable function (i.e., test). Then for any tree $T = ([n], E)$ with vertex set $[n]$ and edge set E, we have*

1. $\frac{1}{n} \sum_{i=1}^{n} Q_i(\Psi \neq i) \geq \frac{1}{n} \sum_{(i,j) \in E} \int \min\{dQ_i, dQ_j\}$;
2. $\frac{1}{n} \sum_{i=1}^{n} Q_i(\Psi \neq i) \geq \frac{1}{2n} \sum_{(i,j) \in E} \exp\left(-D_{KL}(Q_i \| Q_j)\right)$.