Thresholded Linear Bandits

Nishant A. Mehta[1], **Junpei Komiyama**[2], **Vamsi K. Potluru**[3], **Andrea Nguyen**[1], **Mica Grant-Hagen**[1]
[1]University of Victoria [2]New York University [3]J.P. Morgan AI Research
nmehta@uvic.ca, junpei@komiyama.info, vamsi.k.potluru@jpmchase.com
trangn@uvic.ca, micag@uvic.ca

Abstract

We introduce the thresholded linear bandit problem, a novel sequential decision making problem at the interface of structured stochastic multi-armed bandits and learning halfspaces. The set of arms is $[0,1]^d$, the expected Bernoulli reward is piecewise constant with a jump at a separating hyperplane, and each arm is associated with a cost that is a positive linear combination of the arm's components. This problem is motivated by several practical applications. For instance, imagine tuning the continuous features of an offer to a consumer; higher values incur higher cost to the vendor but result in a more attractive offer. At some threshold, the offer is attractive enough for a random consumer to accept at the higher probability level. For the one-dimensional case, we present Leftist, which enjoys $\log^2 T$ problem-dependent regret in favorable cases and has $\log(T)\sqrt{T}$ worst-case regret; we also give a lower bound suggesting this is unimprovable. We then present MD-Leftist, our extension of Leftist to the multi-dimensional case, which obtains similar regret bounds but with $d^{2.5}\log d$ and $d^{1.5}\log d$ dependence on dimension for the two types of bounds respectively. Finally, we experimentally evaluate Leftist.

1 INTRODUCTION

Much is known about how to sequentially maximize cumulative reward in stochastic sequential decision-making problems when the problem structure is finite — as in multi-armed bandit problems with finitely many arms — or continuous — as in linear bandits, Lipschitz bandits, or unimodal bandits. Simultaneously, the machine learning community has a rich body of results for inherently discontinuous problems like learning halfspaces, an instance of classification. Our work introduces a new problem, *thresholded linear bandits*, that lies in the intersection of multi-armed bandits and learning halfspaces. We introduce this problem via the following practically-motivated example.

Vending with an Outside Option Suppose a vendor is selling an essential good. Due to regulation or steep competition, the price is fixed at \$1. However, the vendor can specialize the good $a \in [0,1]^d$ by tuning each of d continuous features $a_1, a_2, \ldots, a_d$; each feature represents the quality of the good along a dimension. Naturally, offering a good a is associated with a (known) cost $c(a)$ that is increasing in each coordinate. In addition, there is an *unknown*, nonnegative linear utility function $a \mapsto \langle \theta^*, a \rangle$ that maps a good $a \in [0,1]^d$ to a utility $u \in \mathbb{R}_+$.

When presented a good, a consumer buys the good if its utility is the highest among the consumers' options. The consumers in the market of interest are of 3 unknown types:

(i) a $1 - p_1$ fraction does not want the good;

(ii) a p_0 fraction needs the good and the vendor is their only option, so they always buy the good;

(iii) a $\Delta = p_1 - p_0$ fraction needs the good but also has an outside option with utility $\tau = \langle \theta^*, a \rangle$, so they buy the good if and only if its linear utility is at least τ.

When considering a random consumer, there is a jump in the probability of buying the good as soon as the good is in the positive halfspace $\{a \in [0,1]^d : \langle \theta^*, a \rangle \geq \tau\}$. This type of discontinuous demand is known to be plausible when a consumer makes choices from a finite consideration set (Caplin et al., 2018); indeed, being the most attractive item among the consideration set is a significant sell to the consumer, which induces a discrete demand. What is the vendor's optimal strategy, the sequence of goods $a \in [0,1]^d$ to offer, in order to maximize expected cumulative profit?

The thresholded linear bandits problem can be viewed from two rather different perspectives: structured bandits and active learning (AL). We explore each perspective in turn.

The halfspace structure of the problem and the ability to

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

decide which arm in all of $[0,1]^d$ to pull in each round suggests viewing the thresholded linear bandits problem as an instance of pool-based AL (Lewis and Gale, 1994; McCallum and Nigam, 1998); here, the pool is the entire *input space* $\mathcal{X} = [0,1]^d$. However, whereas in pool-based AL querying any example's label has the same cost, the queries in our model vary according to our linear cost functional $a \mapsto \langle v, a \rangle$. The goals also differ: pool-based AL seeks to to identify a separating hyperplane of minimum risk under 0-1 loss; in thresholded linear bandits, we wish to eventually commit to a single arm (example) in either the negative or positive halfspace whose cost is minimum (as we explain in Section 2, such a point must either be arm $\mathbf{0}$ or a minimum cost point on the separating hyperplane). It is plausible to use AL strategies (Zhang et al., 2014) to try to first learn (θ^*, τ) and, using this knowledge and a suitable estimate of Δ, to commit to the optimal arm. Yet, we believe direct estimation of (θ^*, τ) is not always necessary.

The thresholded linear bandits problem also can be viewed as a structured multi-armed bandit problem. As opposed to the classical stochastic bandit problem, in a structured bandit problem reward observations for one arm can reveal some information about the expected rewards of other arms (Van Parys and Golrezaei, 2020). Linear bandits (Abbasi-Yadkori et al., 2011) are a typical parametric instance, wherein each arm a is associated with a known d-dimensional feature vector $x_a \in \mathbb{R}^d$, and the expected reward of each arm is $\langle \theta^*, x_a \rangle$ for an unknown parameter vector $\theta^* \in \mathbb{R}^d$. More generally, we can consider nonlinear models via the use of a nonlinearity $\sigma : \mathbb{R} \to \mathbb{R}$, enabling an extension of linear bandits to generalized linear bandits (Filippi et al., 2010) by selecting σ to be the inverse link function for a generalized linear model.

Generalized linear models have widespread appeal, especially in statistics, but they rely upon modeling assumptions such as the inverse link being differentiable and, of course, invertible. These assumptions can be restrictive. Indeed, from a classification perspective, we may wish to take σ to (essentially) be a linear threshold function $x_a \mapsto \mathbf{1}\left[\langle \theta^*, x_a \rangle \geq \tau\right]$ for unknown θ^* as before and unknown threshold $\tau \in \mathbb{R}$, in which case both invertibility and differentiability (and even continuity!) are violated. As a result, existing algorithms for generalized linear bandit, such as GLM-UCB (Filippi et al., 2010) do not apply to thresholded linear bandits. Also, critically, the lack of a cost for pulling arms in generalized linear bandits means that when there is a positive arm that is optimal, this arm does not necessarily lie on the separating hyperplane. Yet, as we show in Section 4, the structure of an optimal solution for thresholded linear bandits is quite different from that of generalized linear bandits.

To see our main idea, consider the one-dimensional case. If we normalize $v = 1$, τ is the minimum arm in the region of p_1. Our main algorithm called Leftist first learns the scale of $\Delta := p_1 - p_0$ compared with τ, with Δ representing the return on the investment. Interestingly, we do not seek for a perfect identification because if $\tau > \Delta$, then arm 0 is optimal — the investment of going from p_0 to p_1 is not worth it. If investing is worthwhile, then the algorithm learns the minimum cost investment (i.e, optimal arm) by using a robust binary search. For the multi-dimensional case, we formulate a subproblem that boils down to the dual formulation of the fractional knapsack problem, for which there is a natural order of the coordinates in terms of the return on investment. In summary, our algorithmic approach to the problem comes with several novel ideas.

Our contributions are as follows:

- We introduce a novel structured bandit problem, thresholded linear bandits, closing a gap in the literature on structured bandits.
- For the one-dimensional case, we introduce two algorithms that we call Explore-the-Gap and Leftist.
- We devise an algorithm that we called MultiDim-Leftist (MD-Leftist) that extends the results of Leftist to the multi-dimensional case.
- We give both problem-dependent and worst-case guarantees on the pseudo-regret for each algorithm.
- Finally, we evaluate Leftist with simulations.

The next section formally introduces the thresholded linear bandits problem. We give algorithms and regret guarantees for the one-dimensional case in Section 3 and for the multi-dimensional case in Section 4. In Section 5, we present some experimental results for the one-dimensional case. Finally, Section 6 concludes the paper with a discussion.

2 PROBLEM SETTING

The thresholded linear bandits problem is a sequential game that takes place over T rounds. In round t, a learning agent (Learner) plays an action a_t from an action space $[0,1]^d$, receives a stochastic Bernoulli revenue μ_t, and pays a cost $c_t = \langle v, a_t \rangle$ according to a known cost vector $v \in (0,\infty)^d$. We assume that for each action a, the stochastic revenue in each round is i.i.d. according to a Bernoulli distribution with mean $\mu(a)$; the Learner's feedback is therefore binary. Our key modeling assumption is that the behavior of the expected revenue function μ is specified by a linear threshold function. Specifically, for a normal vector $\theta^* \in [0,\infty)^d \setminus \{\mathbf{0}\}$, threshold $\tau > 0$, and probabilities p_0 and p_1 satisfying $0 \leq p_0 < p_1 \leq 1$, we have

$$\mu(a) = p_0 + (p_1 - p_0) \cdot \mathbf{1}\left[\langle \theta^*, a \rangle \geq \tau\right];$$

here, $\mathbf{1}[E]$ is the indicator function with respect to event E, and we assume that all the problem parameters θ^*, τ, p_0, and p_1 are unknown. The vector θ^* is normal to the implicit separating hyperplane; without loss of generality we can

and will assume that (θ^*, τ) are scaled such that $\|\theta^*\|_2 = 1$. Defining $\Delta := p_1 - p_0$, the expected reward of an action a_t is now given as $\mu_c(a_t)$, which is defined as

$$\mu(a_t) - c_t = p_0 + \Delta \cdot \mathbf{1}\left[\langle \theta^*, a_t \rangle \geq \tau\right] - \langle v, a_t \rangle. \quad (1)$$

Learner's objective for this problem is to obtain low *pseudo-regret* (hereafter "regret"), defined as

$$\mathcal{R}_T := \max_{a \in [0,1]^d} \sum_{t=1}^{T} \mu_c(a) - \mathsf{E}\left[\sum_{t=1}^{T} \mu_c(a_t)\right].$$

To gain intuition, let us explore what an optimal arm looks like. First, it is easy to see that the expected revenue function is piecewise constant with two pieces, each piece being a halfspace defined by the normal vector θ^* and threshold τ. Since the expected revenue function is constant within each halfspace, the optimal arm within a halfspace is its arm of minimum cost. We say an arm is *positive* if it is in the positive halfspace and *negative* if it is in the negative halfspace. Clearly, the negative arm of minimum cost is arm $\mathbf{0}$. Next, suppose that a is a positive arm that does not lie on the separating hyperplane $H := H_{\theta^*,\tau}$, defined as

$$H_{\theta^*,\tau} := \{a \in \mathbb{R}^d \colon \langle \theta^*, a \rangle = \tau\}$$

and that does not lie on the boundary of $[0,1]^d$. Then since $\langle v, \theta^* \rangle > 0$, we may move a infinitesimally in the direction $-\theta^*$ while simultaneously decreasing the cost. Therefore, arms interior to $[0,1]^d$ that lie in the positive halfspace cannot be optimal unless they belong to H. This implies the following characterization of the set of optimal arms.

Proposition 1. *The optimal arm a^* belongs to the set $(H \cap [0,1]^d) \cup \{\mathbf{0}\}$. In particular, $a^* \in H$ if $\min_{a \in H \cap [0,1]^d} \langle v, a \rangle \leq \Delta$, and arm $\mathbf{0}$ is optimal otherwise.*

The geometry of the problem is illustrated in Figure 1. In the sequel, if $a^* \in H$ we say *H is optimal*.

Related Work As discussed earlier, thresholded linear bandits has some relation to generalized linear bandits. Yu and Mannor (2011) considered the unimodal bandit problem, where arms are associated with a graph and the mean reward of the arms is unimodal. Kleinberg et al. (2008) considered the Lipschitz bandit problem where the arms are associated with a metric space and mean reward of the arms are Lipschitz to the original metric. Compared with these problems, the thresholded linear bandit problem is more challenging in that the proximity to other arms does not give much information on the mean reward: any arm in the same halfspace has the same mean reward.

The thresholded linear bandit setting has some similarity to dynamic pricing (Boer and Keskin, 2022), but the comparison is limited to the one-dimensional case due to core differences between the two problems. The most similar

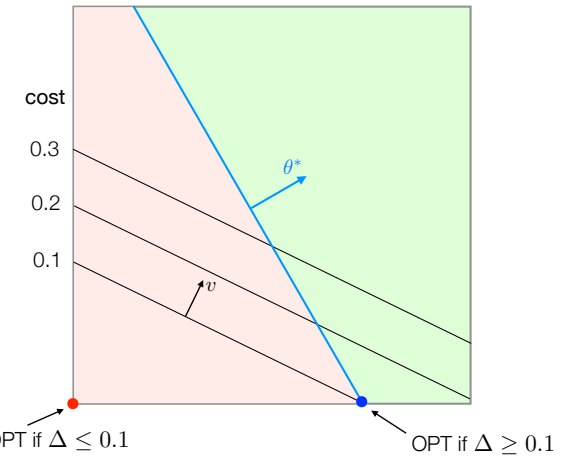


Figure 1: A two-dimensional problem with cost level sets shown by parallel black lines orthogonal to cost vector v. The minimum-cost arm in the positive halfspace (blue point) has cost 0.1 and is optimal if $\Delta \geq 0.1$, while arm $\mathbf{0}$ (red point) is optimal if $\Delta \leq 0.1$.

dynamic pricing works are those of den Boer and Keskin (2020) and Cesa-Bianchi et al. (2019); the latter also consider a piecewise constant expected reward function as it isolates the key difficulty of having a discontinuity. However, whereas in their case, the algorithm's choice is a price and hence is limited to a single dimension, in our case the algorithm's choice can be a collection of quality levels, making a multi-dimensional version natural. In this sense, our work transcends these dynamic pricing works. In both cases, the feedback is binary, but the notion of cost is unique to thresholded linear bandits and is always borne by the vendor, reminiscent of the newsvendor problem (Arrow et al., 1951; Choi, 2012).

We explore the one-dimensional version of the thresholded linear bandits problem in the next section. Before we present our algorithms, we mention that for simplicity and to keep the focus on the key ideas, in the sequel we assume that the failure probability δ is set as $1/T^2$, so that applying Hoeffding's inequality once per time step (which is certainly overcounting) still ensures that with probability at least $1 - 1/T$, all the relevant upper/lower confidence bounds simultaneously hold.

3 ONE-DIMENSIONAL CASE

We begin by considering the case of $d = 1$. Let us first discuss the set of potentially optimal arms. Since $d = 1$ and $\|\theta^*\|_2 = 1$, we implicitly have $\theta^* = 1$, and so the hyperplane H is the arm $\{\tau\}$. Proposition 1 therefore reduces to the following simple characterization of the set of potentially arms.

Corollary 2. *The optimal arm a^* belongs to the set $\{0, \tau\}$. Arm τ is optimal if $\tau \leq \frac{\Delta}{v}$ and arm 0 is optimal if $\tau \geq \frac{\Delta}{v}$.*

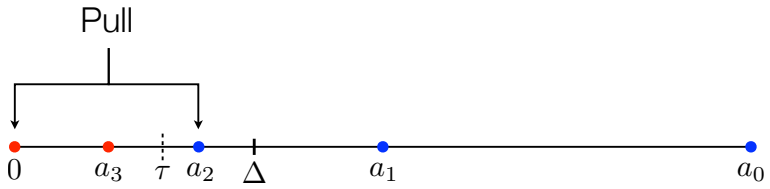


Figure 2: Leftist will pull geometrically decreasing arms until it identifies a lower bound on Δ or a_r is of order $O(\log(T)/\sqrt{T}$, in which case the algorithm will commit to pulling arm 0.

Even in the case of $d = 1$, the threshold bandit problem has interesting structure. There are two challenges in thresholded linear bandits. First, the feedback is very limited: upon comparing two arms a_1, a_2, the learning algorithm only receives (noisy) feedback about whether a_1, a_2 lie in the same halfspace, and any explicit information regarding the distance of these arms from the separating hyperplane H (here, just the threshold τ) is unavailable. Second, each arm $a \in [0, 1]$ is associated with its own fixed cost $v \cdot a$, and thus we need to care about the exploration cost. Even though this problem is an online optimization with bandit (noisy partial) feedback, naive application of existing bandit algorithms such as upper confidence bound (UCB, (Lai et al., 1985; Auer et al., 2002)) or Thompson sampling (TS, (Thompson, 1933)) does not make immediate sense. In particular, a good algorithm must actively search for the hyperplane H. However, a naive binary search with a fixed number of comparisons does not work because the reward gap Δ is unknown, and the number of samples required to differentiate a positive arm from a negative arm depends on Δ. Also, some of the arms can be more costly than others, and naive exploration can result in $O(T)$ regret.

3.1 Lower Bounds of the Regret

We start with a regret lower bound in the one-dimensional thresholded linear bandit problem.

Theorem 3. *For any algorithm, there exists a set of parameters such that the regret of the algorithm is* $\mathcal{R}_T = \Omega(\sqrt{T})$.

At a glance, achieving $\tilde{\Omega}(\sqrt{T})$ regret[1] seems reasonable as it resembles the standard K-armed bandit problem where the worst-case (minimax) regret is $\Theta(\sqrt{T})$. In the standard bandit problem, well-known algorithms such as UCB and TS have $\tilde{O}(\sqrt{T})$ regret as well as a logarithmic regret.

3.2 Explore-the-Gap Algorithm

We first start with an algorithm that aims for a logarithmic regret bound, which we call Explore-the-Gap (EG). Due to space limitations, we only give a brief idea of EG and leave the detailed algorithm to Appendix E. First, EG repeatedly pulls each of arms 0 and 1 until it identifies the scale of Δ up to a constant factor. It then tries to identify the threshold τ by using a robust version of binary search that we call NoisyBinarySearch (Algorithm 2). After obtaining an estimate $\hat{\tau}$ of τ, it runs UCB on arms 0 and $\hat{\tau}$ in order to obtain low regret against the best of these two arms. The following theorem shows that it has a logarithmic regret.

Theorem 4. *The regret of EG is bounded as*

$$\mathcal{R}_T = O\left(\frac{\log^2 T}{\Delta^2} + \frac{\log T}{|\Delta - v \cdot \tau|}\right) \tag{2}$$

Theorem 4 states that Explore-the-Gap has a logarithmic regret bound. The bound appears to be reasonable because we are required to draw suboptimal arms at least $O(\log(T)/\Delta^2)$ times to identify the scale of Δ and $\frac{\log T}{|\Delta - v \cdot \tau|^2}$ times to identify the better arm among arm 0 and τ.[2]

Interestingly, EG performs arbitrarily badly in the worst case: in Appendix E.2, we explicitly construct an example where EG suffers $\Omega(T)$ (linear) regret. The ineffectiveness of EG comes from identifying the scale of Δ: it draws arms 0 and 1 for $O(\log T/\Delta^2)$ times, which is inefficient when Δ is very small. In such a case, exact identification of Δ is not necessary: it suffices to prove that Δ is not very large to conclude that arm τ is suboptimal. Moreover, if τ is small, arm τ is more cost-effective than arm 1. The next section proposes Leftist, which saves the cost of identifying Δ.

3.3 Leftist Algorithm

We now propose Leftist (Algorithm 1). Like EG, Leftist adopts an epoch-based approach. This algorithm involves several variables that change geometrically with epoch r, namely, $\varepsilon_r = 2^{-r}$, $a_r = O(2^{-r}/v)$, and $n_r = O(\log(T)4^r)$. The largest innovation of Leftist compared with EG is to progressively decrease the scale (as well as cost-per-round) of arm a_r that we compare with arm 0. See Figure 2 for an illustration of Leftist in the case of $v = 1$. In epoch r, Leftist (if still running) is able to restrict the search space to the region $[0, O(2^{-r}/v)]$, and the regret contribution per pull is $O(2^{-r})$, which results in an improved distribution-dependent regret of $O(\log(T)/\Delta)$ in total. Once Leftist identifies the value of Δ up to a constant

[1] The notation $\tilde{\Omega}$ ignores a polylogarithmic factor.

[2] Note that square in Δ^2 is correctly placed. Each draw of arm 0 and 1 costs $O(1)$ regret. After we obtain estimator $\hat{\tau}$, the cost of drawing arm 0 or $\hat{\tau}$ is $O(|\Delta - v \cdot \tau|)$.

Algorithm 1: Leftist

Epoch $r \leftarrow 0$, $a_0 \leftarrow 1$, $\varepsilon_0 \leftarrow 1/8$
$n_0 \leftarrow \log(2/\delta)/(2\varepsilon_0^2)$
while $\varepsilon_r \geq \log(T) \cdot T^{-1/2}$ **do**
  Make n_r pulls of arm 0 to get empirical mean $\hat{p}_0$
  Make n_r pulls of arm a_r to get empirical mean $\hat{p}_1$
  $\hat{\Delta}_r \leftarrow \hat{p}_1 - \hat{p}_0$
  if $\hat{\Delta}_r - \varepsilon_r \geq \varepsilon_r$ **then**
    $\hat{\tau} \leftarrow$ NoisyBinarySearch(ε_r, 0, a_r)
    Run UCB on $\{0, \hat{\tau}\}$ until time T
  else
    if $v \cdot a_r \geq 8\varepsilon_r$ **then** $a_{r+1} \leftarrow a_r/2$
    else $a_{r+1} \leftarrow a_r$
    $\varepsilon_{r+1} \leftarrow \varepsilon_r/2$, $n_{r+1} \leftarrow 4 \cdot n_r$
    $r \leftarrow r + 1$
Commit to arm 0.

Algorithm 2: NoisyBinarySearch

Input: Lower bound ε_r satisfying $\varepsilon_r \leq \Delta$, Left-most arm L, Right-most arm R
$N \leftarrow 4\log(\frac{1}{\delta})/\varepsilon_r^2$
Make N pulls of arm L to get empirical mean $\hat{p}_{\text{left}}$
Make N pulls of arm R to get empirical mean $\hat{p}_{\text{right}}$
while $R - L \geq \varepsilon_{\text{NBS}}$ **do**
  $m \leftarrow \frac{L+R}{2}$
  Make N pulls of arm m to get empirical mean $\hat{p}_{\text{mid}}$
  if $\hat{p}_{\text{right}} - \hat{p}_{\text{mid}} \geq \frac{\varepsilon_r}{2}$ **then** $L \leftarrow m$, $\hat{p}_{\text{left}} \leftarrow \hat{p}_{\text{mid}}$
  else $R \leftarrow m$, $\hat{p}_{\text{right}} \leftarrow \hat{p}_{\text{mid}}$
return R

factor, Figure 2, it starts a noisy binary search to identify the boundary. When arm 0 is optimal, it need not identify a lower bound on Δ as a_r converges to an arm of order $O(\log(T)/\sqrt{T})$.

For the next two theorems, we make the very mild assumption that the known value v satisfies $v \geq \log(T) \cdot T^{-1/2}$. We first present a problem-dependent regret bound in the case that arm τ is optimal and Δ is suitably large.

Theorem 5. *Take $\varepsilon_{\text{NBS}} = 1/T$ and $\delta = 1/T^2$. Suppose that $\tau \leq \Delta/v$. Then the regret of Leftist is bounded as*

$$\mathcal{R}_T = O\left(\frac{\log^2 T}{\Delta} + \min\left\{\sqrt{T\log T}, \frac{\log T}{|\Delta - v\cdot\tau|}\right\}\right).$$

The next result applies generally; it is particularly useful for handling the cases that arm 0 is optimal or Δ is very small.

Theorem 6. *Take $\varepsilon_{\text{NBS}} = 1/T$ and $\delta = 1/T^2$. Then the regret of Leftist is bounded as*

$$\mathcal{R}_T = O\left(\log(T)\sqrt{T} + \min\left\{\sqrt{T\log T}, \frac{\log T}{|\Delta - v\cdot\tau|}\right\}\right).$$

If it further holds that $\Delta \leq \log(T) \cdot \frac{T^{-1/2}}{2}$, then the bound can be improved to $\mathcal{R}_T = O\left(\log(T)\sqrt{T}\right)$.

Remark 7. The bound in Theorem 6 is $O(\log(T)\sqrt{T})$, which is minimax optimal up to a logarithmic factor in view of Theorem 3. Moreover, the distribution-dependent regret of Leftist in Theorem 5, which holds when arm 1 is optimal and Δ is not too small, is

$$\frac{\log T}{\Delta} + \frac{\log T}{|\Delta - v\cdot\tau|}, \tag{3}$$

which is better than EG in the sense that dependence on Δ is improved from Δ^2 to Δ. Theorems 5 and 6 imply that when arm 0 is optimal and Δ is small, Leftist does *not* have a logarithmic regret bound. We consider this as not suboptimality but a result of Leftist achieving minimax regret: somewhat surprisingly, there is a case in which the identification of the optimal arm results in large regret, and Leftist successfully avoid this.

We exhibit the aforementioned case with a pair of interesting models that involve a trade-off.

Theorem 8. *Let $\eta \in (0, 1/2)$ be arbitrary. There exists a pair of models where any algorithm either* (i) *has $\Omega(T^{1-\eta})$ regret in one of the two models, or* (ii) *has $\Omega(\sqrt{T})$ regret and it draws arms of the suboptimal halfspace for $\Theta(T)$ times in one of the two models.*

Intuitively speaking, in the former case we identified the optimal arm (by paying the cost of $\Omega(T^{1-\eta})$, whereas in the latter case we skip the identification of the optimal arm and receives a smaller regret. Leftist chooses the latter and has an $\tilde{O}(\sqrt{T})$ regret bound.

Before showing a proof sketch of Theorem 5, we introduce some epoch-related concepts. First, we define ρ be the stopping epoch of Leftist; this is either the epoch in which NoisyBinarySearch (NBS) is called or, if the former is never called, epoch $r_{\max} = O(\log_2 \sqrt{T/(\log T)})$, which is the largest possible epoch. Next, when arm τ is optimal, we can show that with high probability, ρ is no greater than 3 epochs after the following critical epoch:

$$r_\Delta := \arg\max_{r\geq 0} \{\varepsilon_r > \Delta\}.$$

Lemma 9. *If $\tau \leq \Delta/v$, then $\rho \leq r_\Delta + 3$.*

Lemma 9 states that the algorithm stops in a proper round when τ is the optimal arm. The next lemma is instrumental in proving the previous claim.

Lemma 10. *Let $r^* = r_\Delta + 3$. If $\tau \leq \Delta/v$, then for all epochs $r \leq \min\{r^*, \rho\}$, arm a_r is positive. Also, with probability at least $1 - 1/T$, the lower confidence bound of Δ at epoch r^* satisfies $\Delta/2 \leq \hat{\Delta}_{r^*} - \varepsilon_{r^*}$.*

The interpretation of the above lemma is that either arm τ is optimal, in which case the algorithm collects informative

samples (a_r is positive), or a lower bound on Δ has been identified (which is how Lemma 9 upper bounds the stopping epoch ρ). Note that if arm 0 is optimal, arm a_r can be negative, but it will then converge to an arm of order $O(1/\sqrt{T})$; this case is handled by Theorem 6.

Proof Sketch (of Theorem 5). Leftist (Algorithm 1) tries to find τ that is optimal by assumption. We split the rounds into (A) before and (B) after entering NoisyBinarySearch, and decompose the proof into the following steps.

Regret in Rounds (A): At each epoch r, Leftist compares arm 0 and a_r. With high-probability, we have $\tau \in [0, a_r]$ for all epochs r. The cost of exploration for each epoch r is upper-bounded by $v \cdot a_r \cdot n_r = O((\log T) \cdot 2^r)$, and thus the total cost is on the order of

$$2^\rho = O\left(\log(T) \min\left\{\sqrt{\tfrac{T}{\log T}}, \tfrac{1}{\Delta}\right\}\right).$$

Regret in Rounds (B): If $\Delta = \Omega(2^{-r})$, then Leftist enters NBS. NBS requires $O(\log T)$ pairwise comparisons, and each pairwise comparison costs $O((\log T) \cdot 2^r)$. This suffices to find a point $\hat{\tau} : |\hat{\tau} - \tau| = O(T^{-1})$ with a high probability. As a result, the total cost of NBS is

$$O\left((\log T) \cdot (\log T) \cdot 2^r\right) = O\left(\tfrac{(\log T)^2}{\Delta}\right).$$

Finally, running UCB on arms 0 and $\hat{\tau}$ picks up regret at most $O\left(\frac{\log T}{|\Delta - v \cdot \tau|}\right)$. □

4 MULTI-DIMENSIONAL CASE

We now begin generalizing Leftist to the multi-dimensional setting; since $d > 1$, there is an unknown vector θ^* that is normal to the hyperplane. The multi-dimensional setting introduces several new challenges. We develop our algorithm for this setting, MultiDimLeftist (MD-Leftist, Algorithm 3), in the course of discussing these challenges.

First, our approach in the one-dimensional setting involved an exploration phase that achieved the following goal: if an arm on the separating hyperplane (there, the threshold τ) is optimal, then Leftist identified a positive arm whose cost is of order Δ. Because there was only a single dimension, all arms pulled by Leftist were along the line segment $[0, 1]$. The first challenge in MD-Leftist is to scale down arm a_r in the multi-dimensional setting. As the set of arms is now $[0, 1]^d$, it no longer is clear if there is a single line segment along which MD-Leftist can explore while keeping the regret under control.

As it turns out, there is such a single line segment. For an axis-parallel rectangle A, let $R(A)$ be the vertex for which all coordinates are maximized. This is the multi-dimensional analogue of the "right-most" point of a closed interval (a one-dimensional axis-parallel rectangle). Central to MD-Leftist's operation is the following type of axis-parallel rectangle. Let A_r be the axis-parallel rectangle whose j^{th} side is $2^{-r} \cdot v_{\min} \cdot [0, \frac{1}{v_j}]$, for $v_{\min} = \min\{\min_j v_j, 1\}$. We remark that A_0 is obtained by starting with a rectangle whose sides are $\frac{1}{v_j}$ and scaling down this rectangle (if needed) until it is contained within the unit cube. Notice that the cost of $R(A_r)$ is equal to

$$\langle v, R(A_r)\rangle = d \cdot 2^{-r} \cdot v_{\min}, \tag{4}$$

which, in the case of $v = \mathbf{1}$, reduces to $d \cdot 2^{-r}$. Arms of the type $R(A_r)$ satisfy an important property.

Proposition 11. *For any $r \geq 0$, if $R(A_r)$ is negative, then any arm with cost at most $2^{-r} \cdot v_{\min}$ also must be negative.*

Considering the contrapositive form of the above proposition, we have that if there is a positive arm with cost at most $2^{-r} \cdot v_{\min}$, then arm $R(A_r)$ must be positive. To make this algorithmic, suppose that we have discovered that Δ is at most $2^{-r} \cdot v_{\min}$ for some r, with high probability. Then it suffices to pull arm $R(A_r)$ to determine whether any arm on the hyperplane can be optimal. Moreover, the cost of a pull of this arm is d times our believed upper bound on Δ. This intuition essentially captures the behavior of MD-Leftist.

The second challenge is to find a point on the boundary H. If and when we find a first positive arm by searching along the line segment described above, we can still use NoisyBinarySearch (NBS) to find a positive arm a^f that is arbitrarily close to the separating hyperplane H (using $O(\log(1/\varepsilon_{\text{NBS}}))$ rounds of NBS; by searching along the same direction as MD-Leftist, the contribution to the regret per round of NBS will be essentially the same as the contribution from the pulls from MD-Leftist.

Before discussing the next key challenge, recall that either arm $\mathbf{0}$ is optimal or the minimum cost arm on H is optimal. Hence, we next consider how to go from our low-cost positive arm a^f that approximately is on H, to an arm approximately on H that also is approximately of minimum cost *among all arms on H*. For this, we consider a characterization of the optimal solution a^H to the problem of finding an arm $a \in [0, 1]^d$ that minimizes the cost $\langle v, a\rangle$ subject to the constraint that $\langle \theta^*, a\rangle = \tau$. The constraint is feasible since we assume that arm $\mathbf{0}$ is negative and arm $\mathbf{1}$ is positive. The above problem is a dual formulation of the fractional knapsack problem, for which the following greedy strategy is known to be optimal: put the coordinates $[d]$ in non-increasing order of the *leverage score* θ^*_j / v_j, and then, one coordinate a time, increase the coordinate from 0 up to 1 until the constraint is satisfied; if the constraint is not satisfied, move on to the next coordinate (keeping all previous coordinates in the ordering set to 1). Namely, the following proposition holds.[3]

[3]For a permutation σ of $(1, 2, \ldots, d)$, let $\sigma(i)$ denote the i^{th}

Proposition 12. *Let $\sigma(j)$ be a permutation of $[d]$ such that*

$$\frac{\theta^*_{\sigma(j)}}{v_{\sigma(j)}} \geq \frac{\theta^*_{\sigma(k)}}{v_{\sigma(k)}} \tag{5}$$

if $j < k$. Then, there exists an optimal arm in the hyperplane $a^H \in \arg\min_{a \in H} \langle v, a \rangle$ such that, for some $l \in [d]$,

$$(a^H_{\sigma(1)}, a^H_{\sigma(2)}, \ldots, a^H_{\sigma(d)}) = (1, \ldots, 1, a^H_{\sigma(l)}, 0, \ldots, 0).$$

The third challenge is how to go from the aforementioned arm a^f to the ideal a^H. Using the greedy paradigm, we seek to find an ordering (permutation) of the coordinates σ such that for all $j, k \in [d]$, we have $\frac{\theta^*_{\sigma(j)}}{v_{\sigma(j)}} \geq \frac{\theta^*_{\sigma(k)}}{v_{\sigma(k)}}$ if $j < k$. However, since θ^* is unknown, we need a way of using 1-bit feedback to determine an optimal ordering. Because of our feedback model, if two coordinates have leverage scores that are very close, it becomes more expensive to determine their order. Our approach is therefore to recover an approximate order such that, for some tunable parameter γ', we find a permutation σ such that, for all $j, k \in [d]$ satisfying $j < k$, we have $\frac{\theta^*_{\sigma(j)}}{v_{\sigma(j)}} \geq \frac{\theta^*_{\sigma(k)}}{v_{\sigma(k)}} - \gamma'$. Formally, we call such an ordering a *γ'-insensitive ordering*. We achieve such an ordering using PAC-MergeSort (PAC for "probability approximately correct"), whose key innovation is MultiCoordinateCompare (see Algorithm 5) to approximately compare coordinates. In more detail, MultiCoordinateCompare satisfies the contract that if two coordinates differ by γ, then with high probability it correctly identifies their order; otherwise, it is allowed to output "$*$", indicating "I don't know". In this sense, MultiCoordinateCompare is an implementation of what we call a *PAC comparison oracle*. For brevity, we leave a description of PAC-MergeSort to Appendix B, but in short, it operates like standard MergeSort with a simple rule of joining elements when their comparison returns "$*$". Our (likely loose) analysis shows that given a γ-PAC comparison oracle, PAC-MergeSort produces with high probability a γ'-insensitive ordering for $\gamma' = (d-1) \cdot \gamma$.

Once we have a γ'-insensitive ordering, the final challenge is how to go from this ordering to an approximately optimal solution. Suppose that we could directly map a given ordering σ to the output a_σ of the greedy algorithm when instantiated with this ordering. We still need to ensure that a_σ has cost that is not much larger than a^H, the output of the greedy algorithm when given an optimal ordering. For this, our analysis requires an assumption that any non-zero coordinate of θ^* is bounded away from zero (see Assumption 13 and its discussion below). Next, given an approximately optimal ordering (for which greedy would output an approximately optimal solution), we need a way to actually implement the greedy algorithm using 1-bit feedback. We accomplish this using MultiGreedy (see Algorithm 6), which builds up a solution in a greedy fashion as described above, using sparring between a "smaller" arm L and a "larger" arm R in each iteration until it detects that the larger arm is positive, after which NoisyBinarySearch is used to set the value for the last non-zero coordinate in the ordering. We mention the value u in Algorithm 6 is there to ensure that we do not pull arms of very high cost relative to Δ.

We leave to the appendix a full discussion of the algorithms, along with the theoretical developments and regret analysis. Let us now consider the regret bounds for MD-Leftist.

Regret Guarantees We focus primarily on the regime where $v_{\min}$ is a positive constant. Due to the complexity of the problem, we leave consideration of very small values of $v_{\min}$ (like $v_{\min} = \Theta(T^{-1/2})$) to future work. As mentioned above, we require an assumption on θ^* to control the regret in the situation when the algorithm calls MultiGreedy.

Assumption 13 (Regularity of θ^*)**.** There exists a known positive constant $\theta_{\min} > 0$ such that, for any coordinate j for which θ^* is non-zero, we have $\theta^*_j \geq \theta_{\min}$.

This assumption has practical merit. Consider the vending example from the beginning of this paper. Each component of the vector θ^* reflects how much attention the consumer pays to a given feature. From human psychology, it is unlikely that someone would pay a very small, positive amount of attention $0 < \theta^*_j \ll 1$). Rather, they are more likely to simply ignore a feature ($\theta^*_j = 0$).

We have the following problem-dependent guarantee in the case that H is optimal and Δ is not too small.

Theorem 14. *Let Assumption 13 be satisfied and take $\varepsilon_{\text{NBS}} = \theta_{\min}/(d^3T^2)$ and $\delta = 1/T^2$. Suppose that H is optimal. If $16T^{-1/2} < \Delta \leq v_{\min}/16$, then the regret of MD-Leftist is bounded as*

$$O\left(\begin{array}{c}\log(T) \cdot \left(\frac{\Delta + \|v\|_1}{v^2_{\min}} + \frac{\max\left\{d \log(1/\varepsilon_{\text{NBS}}), d^2 \log d\right\}}{\Delta}\right) \\ + \min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v^*|}\right\}\end{array}\right),$$

where v^* is the cost of the minimum-cost positive arm.

The next result parallels Theorem 6; it is particularly useful to handle the cases that arm $\mathbf{0}$ is optimal or Δ is small.

Theorem 15. *Let Assumption 13 be satisfied and take $\varepsilon_{\text{NBS}} = \theta_{\min}/(d^3T^2)$ and $\delta = 1/T^2$. If $\Delta \leq v_{\min}/16$, then the regret of MD-Leftist is bounded as*

$$O\left(\begin{array}{c}\log(T) \cdot \frac{\Delta + \|v\|_1}{v^2_{\min}} + \max\left\{\sqrt{d} \log \frac{1}{\varepsilon_{\text{NBS}}}, d^{1.5} \log d\right\} \sqrt{T} \\ + \min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v^*|}\right\}\end{array}\right).$$

where v^* is the cost of the minimum-cost positive arm.

If we further have $\Delta \leq \frac{1}{2} \log(T) \sqrt{\frac{d}{T}}$, then

$$\mathcal{R}_T = O\left(\log(T) \cdot \left(\frac{1}{v^2_{\min}} \cdot (\Delta + \|v\|_1) + \sqrt{Td}\right)\right).$$

element in the permutation. This may be non-standard use of permutation notation, but our convention is more convenient here.

Algorithm 3: MultiDimLeftist (MD-Leftist)

$\phi = 1$ // start in Phase 1
$r \leftarrow 0$, $a_0 \leftarrow \mathbf{1}$, $\varepsilon_0 \leftarrow \frac{1}{8}$
$n_0 \leftarrow \frac{\log \frac{2}{\delta}}{2\varepsilon_0^2}$
while $\varepsilon_r \geq \log(T)\sqrt{\frac{d}{T}}$ **do**
  Make n_r pulls of arm $\mathbf{0}$ to get empirical mean $\hat{p}_{\mathbf{0}}$
  Make n_r pulls of arm a_r to get empirical mean $\hat{p}_{a_r}$
  $\hat{\Delta}_r \leftarrow \hat{p}_{a_r} - \hat{p}_{\mathbf{0}}$
  if $\hat{\Delta}_r - \varepsilon_r \geq \varepsilon_r$ **then**
    $a^f \leftarrow$ MultiEstTau(ε_r, a_r)
    Run UCB on $\{\mathbf{0}, a^f\}$ until time T
  else
    if $\phi = 2$ **then** $a_{r+1} \leftarrow \frac{1}{2} \cdot a_r$
    else if $\langle v, R(A_0)\rangle \geq d \cdot 8\varepsilon_r$ **then**
      $a_{r+1} \leftarrow R(A_1)$, $\phi = 2$ // Phase 2
    else $a_{r+1} \leftarrow a_r$
    $\varepsilon_{r+1} \leftarrow \varepsilon_r/2$, $n_{r+1} \leftarrow 4 \cdot n_r$
    $r \leftarrow r + 1$
Commit to arm $\mathbf{0}$

Algorithm 4: MultiEstTau

Input: Lower bound $\varepsilon_r \leq \Delta$, Right-most arm R
$a^c \leftarrow$ NoisyBinarySearch(ε_r, $\mathbf{0}$, R)
Use PAC-MergeSort with PAC comparison oracle MCC($\cdot, \cdot, \varepsilon_r, a^c$) to obtain ordering $\hat{\sigma}$
return MultiGreedy($\varepsilon_r, \hat{\sigma}$)

Algorithm 5: MultiCoordinateCompare (MCC)

Input: Two coordinates j, k. Lower bound ε_r satisfying $\varepsilon_r \leq \Delta$, arm a
Assume $\langle \theta^*, a\rangle \geq \tau$ and $\|a - \pi(a)\| \leq \varepsilon_{\text{NBS}}$
$\gamma \leftarrow \theta_{\min}/(d^2 T)$
$\beta \leftarrow \varepsilon_{\text{NBS}}/\gamma$, $N \leftarrow \frac{\log \frac{1}{\delta}}{\varepsilon_r^2}$
$a^{(j)} \leftarrow a + \beta \cdot \left(\frac{\mathbf{e}_j}{v_j} - \frac{\mathbf{e}_k}{v_k}\right)$
$a^{(k)} \leftarrow a - \beta \cdot \left(\frac{\mathbf{e}_j}{v_j} - \frac{\mathbf{e}_k}{v_k}\right)$
Make N pulls of each of arms $\mathbf{0}$, $a^{(j)}$, and $a^{(k)}$, giving $\hat{p}_0$, $\hat{p}_j$ and $\hat{p}_k$
if $\hat{p}_k - \hat{p}_0 < \varepsilon_r/2$ **then return** "$>$"
else if $\hat{p}_j - \hat{p}_0 < \varepsilon_r/2$ **then return** "$<$"
else return "$*$"

Algorithm 6: MultiGreedy

Input: Lower bound ε_r satisfying $\varepsilon_r \leq \Delta$, Ordering $\hat{\sigma}$
$u \leftarrow 2d \cdot 2^{-r} + 1/T$
$N \leftarrow \frac{\log \frac{1}{\delta}}{\varepsilon_r^2}$, $L \leftarrow \mathbf{0}$
for $i = 1, 2, \ldots, d$ **do**
  $R \leftarrow L + \min\left\{\frac{1}{v_{\hat{\sigma}(i)}}\left(u - \sum_{j=1}^{i-1} v_{\hat{\sigma}(j)}\right), 1\right\} \cdot \mathbf{e}_{\hat{\sigma}(i)}$
  Make N pulls of each of arms L and R
  if $\hat{p}_{\text{right}} - \hat{p}_{\text{left}} \geq \varepsilon_r/2$ **then**
    return NoisyBinarySearch(ε_r, L, R)
  else $L \leftarrow L + \mathbf{e}_{\hat{\sigma}(i)}$

Remark 16. Theorem 14 and Theorem 15 correspond to the distribution-dependent and the distribution-independent bounds, respectively. If a^H is optimal and some assumptions are satisfied, we have a logarithmic bound of Theorem 14. Note that, as discussed in Remark 7, for some parameters where arm $\mathbf{0}$ is optimal, Leftist does not have a logarithmic regret bound, but still we have a $\tilde{O}(d^{1.5}\log(d)\sqrt{T})$ bound of Theorem 15.

5 EXPERIMENTS

In this section, we evaluate Leftist's practical performance via several experiments in the one-dimensional case ($d = 1$) with $v = 1$. We use pseudo-regret to evaluate the algorithm's empirical performance. Every parameter variant was run for $m = 25$ trials to obtain the average cumulative pseudo-regret. For two experiments either τ or Δ was varied while the other remained static. These two experiments consisted of a fixed number of pulls ($T = 10^6$) and $p_0 = 0.25$. In a later experiment, we tested the performance of Leftist with varying time horizon T. In all three experiments, Leftist was compared Grid-UCB which is a modified version of UCB run on a grid of $\sqrt{T}$ arms. Grid-UCB serves as a reasonably strong baseline because, as we sketch in the appendix, we believe it is minimax (i.e., has an $\tilde{O}(\sqrt{T})$ regret bound) like Leftist. Note also that the performance of EG, which is generally outperformed by Leftist, is also shown in the appendix.

Overall, Leftist consistently outperformed Grid-UCB. When $\tau \leq \Delta$, the regret for pulling arm 0 is $\Delta - \tau$. Thus, when τ is close to zero, in Figure 3 (left), Leftist suffers a regret close to Δ for pulling arm 0. Furthermore, as the gap between Δ and τ decreases, the regret incurred for pulling arm 0 decreases, as observed when $\tau = [0.01, 0.1]$. From $\tau = [0.26, 0.40]$, arm a_r drops below $\log(T)/\sqrt{T}$, Leftist therefore correctly commits to arm 0, the optimal arm. We observed that the region from approximately $\tau = [0.10, 0.24)$ is where NBS incurs the most regret. Here, as τ increases, NBS (which is searching for τ) tends to pull larger arms, hence accumulating more regret.

In Figure 3 (middle), within the region where $\Delta = [0.005, 0.1)$ Leftist commits to arm 0 without running NBS for the same reason as mentioned earlier. Similarly, the region for $\Delta = [0.03, 0.1)$ that spikes is where NBS is called and incurs the most regret. Once $\Delta \geq \tau$, the regret from Leftist drops as Δ becomes large.

We also compared the performance of both algorithms at varying values of T and found Leftist to be significantly

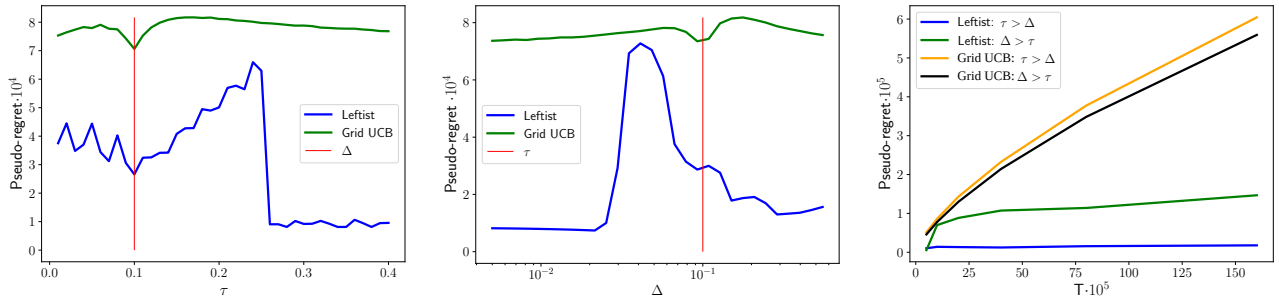


Figure 3: Average cumulative pseudo-regret ($m = 25$) of Leftist and Grid UCB per τ and Δ variant. (Right) Average pseudo-regret ($m = 25$) of Leftist and Grid UCB for a geometrically increasing number of pulls, $T = [5e5, 16e6]$.

better (see Figure 3 (right)). Though both are minimax (they behave similarly in the worst case), in many cases Leftist outperforms the Grid-UCB.

6 DISCUSSION

The thresholded linear bandits problem introduces new challenges not present in seemingly related problems like generalized linear bandits. We developed an algorithm for the one-dimensional setting, Leftist, which enjoys logarithmic regret when Δ and $|\Delta - \tau|$ are not too small as well as a minimax $\tilde{O}(\sqrt{T})$ bound up to a polylog factor. Our MD-Leftist algorithm for the multi-dimensional setting also enjoy logarithmic and worst-case $\tilde{O}(d^{1.5}\log(d)\sqrt{T})$ regret.

We close with some directions for future work. Beyond investigating whether we can obtain improved regret in the multi-dimensional setting with respect to d, we also would like to consider more advanced models of expected reward such as a union of halfspaces model. Another direction of extending model is a combination of (generalized) linear bandit and discontinuity. Such a combination is observed in many regulatory domains. For example, wage is discontinuous around the minimum wage (Blisard et al., 2004).

Acknowledgements

NM, AN, and MG were supported by a JP Morgan Faculty Research Award and NSERC Discovery Grant RGPIN-2018-03942.

Disclaimer. This paper was prepared for informational purposes by the Artificial Intelligence Research group of JPMorgan Chase & Co and its affiliates ("J.P. Morgan"), and is not a product of the Research Department of J.P. Morgan. J.P. Morgan makes no representation and warranty whatsoever and disclaims all liability, for the completeness, accuracy or reliability of the information contained herein.

References

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems 24*, pages 2312–2320, 2011.

Kenneth J Arrow, Theodore Harris, and Jacob Marschak. Optimal inventory policy. *Econometrica (pre-1986)*, 19 (3):250, 1951.

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

Noel Blisard, Hayden Stewart, and Dean Jolliffe. Low-income households' expenditures on fruits and vegetables. *USDA Economic Research Service. Agricultural Economic Report No. 833*, 02 2004.

Arnoud V den Boer and Nuri Bora Keskin. Dynamic pricing with demand learning: Emerging topics and state of the art. *The Elements of Joint Learning and Optimization in Operations Management*, pages 79–101, 2022.

Andrew Caplin, Mark Dean, and John Leahy. Rational Inattention, Optimal Consideration Sets, and Stochastic Choice. *The Review of Economic Studies*, 86(3):1061–1094, 07 2018. ISSN 0034-6527. doi: 10.1093/restud/rdy037.

Nicolo Cesa-Bianchi, Tommaso Cesari, and Vianney Perchet. Dynamic pricing with finitely many unknown valuations. In *Algorithmic Learning Theory*, pages 247–273. PMLR, 2019.

Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework, results and applications. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, page I–151–I–159. JMLR.org, 2013.

Tsan-Ming Choi. *Handbook of Newsvendor problems: Models, extensions and applications*, volume 176. Springer, 2012.

Arnoud V. den Boer and N. Bora Keskin. Discontinuous demand functions: Estimation and pricing. *Management Science*, 66(10):4516–4534, 2020. doi: 10.1287/mnsc.2019.3446.

Sarah Filippi, Olivier Cappe, Aurélien Garivier, and Csaba Szepesvári. Parametric bandits: The generalized linear case. In *NIPS*, volume 23, pages 586–594, 2010.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. *Journal of Machine Learning Research*, 17(1):1–42, jan 2016. ISSN 1532-4435.

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*, STOC '08, page 681–690. Association for Computing Machinery, 2008. ISBN 9781605580470. doi: 10.1145/1374376.1374475.

Tze Leung Lai, Herbert Robbins, et al. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

David D Lewis and William A Gale. A sequential algorithm for training text classifiers. In *SIGIR'94*, pages 3–12. Springer, 1994.

Andrew McCallum and Kamal Nigam. Employing em and pool-based active learning for text classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 350–358, 1998.

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.

Bart PG Van Parys and Negin Golrezaei. Optimal learning for structured bandits. *arXiv preprint arXiv:2007.07302*, 2020.

Jia Yuan Yu and Shie Mannor. Unimodal bandits. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 41–48, 2011.

Lijun Zhang, Jinfeng Yi, and Rong Jin. Efficient algorithms for robust one-bit compressive sensing. In *International Conference on Machine Learning*, pages 820–828. PMLR, 2014.

Contents

Overview of the appendix

In our analysis of the one-dimensional and multi-dimensional settings, we introduce several events. As we show in Appendix C, all these events hold with high probability $(1 - O(1/T))$. Hence, the regret in the situation that the events do not hold only contributes an additive constant to our regret analysis.

A Analysis for one-dimensional case

For convenience, we first re-present Leftist. This version is identical to version appearing in Algorithm 1 of the main text.

Algorithm 7: Leftist

Epoch $r \leftarrow 0$, $a_0 \leftarrow 1$, $\varepsilon_0 \leftarrow \frac{1}{8}$ `// start in Phase 1`
$n_0 \leftarrow \frac{\log \frac{2}{\delta}}{2\varepsilon_0^2}$
while $\varepsilon_r \geq \log(T) \cdot T^{-1/2}$ **do**
  Make n_r pulls of arm 0 to get empirical mean $\hat{p}_0$
  Make n_r pulls of arm a_r to get empirical mean $\hat{p}_1$
  $\hat{\Delta}_r \leftarrow \hat{p}_1 - \hat{p}_0$
  if $\hat{\Delta}_r - \varepsilon_r \geq \varepsilon_r$ **then**
    $\hat{\tau} \leftarrow$ NoisyBinarySearch(ε_r, 0, a_r)
    Run UCB on $\{0, \hat{\tau}\}$ until time T
  else
    if $v \cdot a_r \geq 8\varepsilon_r$ **then**
      $a_{r+1} \leftarrow \frac{1}{2} \cdot a_r$ `// switch to Phase 2`
    else
      $a_{r+1} \leftarrow a_r$
    $\varepsilon_{r+1} \leftarrow \varepsilon_r/2$
    $n_{r+1} \leftarrow 4 \cdot n_r$
    $r \leftarrow r + 1$
Commit to arm 0.

A.1 Regret bounds for one-dimensional case

Recall that we assume $v \geq \log(T) \cdot T^{-1/2}$.

For the convenience of the reader, we begin by re-presenting Theorems 5 and 6, our regret bounds for the one-dimensional case. First, we present a problem-dependent regret bound in the case that Leftist stops in Phase 2.

Theorem 17 (Theorem 5 from the main text)**.** *Take $\varepsilon_{\text{NBS}} = 1/T$ and $\delta = 1/T^2$. Suppose that $\tau \leq \Delta/v$. Then the regret of Leftist is bounded as*

$$\mathcal{R}_T = O\left(\frac{\log^2 T}{\Delta} + \min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v \cdot \tau|}\right\}\right).$$

The next theorem applies more generally. It is particularly useful for handling the case that arm 0 is optimal or the case when Δ is small.

Theorem 18 (Theorem 6 from the main text)**.** *Take $\varepsilon_{\text{NBS}} = 1/T$ and $\delta = 1/T^2$. Then the regret of Leftist is bounded as*

$$\mathcal{R}_T = O\left(\log(T)\sqrt{T} + \min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v \cdot \tau|}\right\}\right).$$

If it further holds that $\Delta \leq \log(T) \cdot \frac{T^{-1/2}}{2}$, then the bound can be improved to

$$\mathcal{R}_T = O\left(\log(T)\sqrt{T}\right).$$

We present proofs of the above theorems in Section A.5.4.

A.2 Analysis for Leftist

A.2.1 Preliminaries

In the sequel, we introduce a special epoch r_{I} which is the last round of Phase 1. More precisely:

$$r_{\mathrm{I}} := \arg\min_{r \geq 0} \{v \geq 8\varepsilon_r\}.$$

In Phase 2, arm a_r will halve until Leftist believes it has a satisfactory upper confidence bound on arm τ or it commits to arm 0. In the case where $\frac{\Delta}{v} \geq \tau$, to ensure that it does not pull an arm less τ, Leftist will use a *halting condition*, $\hat{\Delta}_r - \varepsilon_r \geq \varepsilon_r$.

We define ρ to be the stopping epoch of Leftist; this is either the epoch in which NoisyBinarySearch is called or, if the former is never called, the largest possible epoch $r_{\max} := \lfloor \log_2 \frac{\sqrt{T}}{\log T} \rfloor - 3$. When τ is optimal, we can show that with high probability, Leftist's stopping epoch ρ is no greater than 3 epochs after the critical epoch r_Δ, defined as

$$r_\Delta := \arg\max_{r \geq 0} \{\varepsilon_r > \Delta\}. \tag{6}$$

For later use, it will be convenient to note the explicit value of the critical epoch:

$$r_\Delta = \left\lceil \log_2 \frac{1}{\Delta} \right\rceil - 4. \tag{7}$$

Before beginning the main analysis, it will be useful to introduce an event. Let $\mathcal{E}_{\mathrm{L}}$ be the event that both of the following are true:

- In Leftist, for all epochs $r \leq \rho$ such that arm a_r is positive, $|\hat{\Delta}_r - \Delta| \leq \varepsilon_r$;
- In Leftist, for all epochs $r \leq \rho$, it holds that $\hat{\Delta}_r - \varepsilon_r \leq \Delta$.

A.2.2 Correctness analysis

We first state a lemma that will be useful later.

Lemma 19. *If r_Δ exists, then for all epochs $r_\Delta + 1 + i$ where $i \geq 0$,*

$$\frac{\Delta}{2^i} \geq \varepsilon_{r_\Delta + 1 + i}$$

Proof. From the definition of r_Δ, at epoch $r_\Delta + 1$ we have $\Delta \geq \varepsilon_{r_\Delta + 1}$. Since ε_r is halved after every epoch, the claim follows. □

We need to ensure that the lower confidence bound used by Leftist is not too large.

Lemma 20. *Let $r = r_\Delta + 3$. If $\tau \leq \frac{\Delta}{v}$, then on event $\mathcal{E}_{\mathrm{L}}$, the lower confidence bound of Δ is lower bounded as*

$$\frac{\Delta}{2} \leq \hat{\Delta}_r - \varepsilon_r.$$

Proof. We want to know if the following inequality holds

$$\frac{\Delta}{2} \leq \hat{\Delta}_r - \varepsilon_r. \tag{8}$$

It is true that (8) is equivalent to

$$\frac{\Delta}{2} + 2\varepsilon_r \leq \hat{\Delta}_r + \varepsilon_r. \tag{9}$$

We will lower bound the right-hand side by Δ and then upper bound the left-hand side by Δ, after which the above inequality follows. Beginning with the right-hand side, we know that $\Delta < \varepsilon_{r_\Delta}$ by definition of r_Δ which gives us the following inequality,

$$\tau \leq \frac{\Delta}{v} < \frac{\varepsilon_{r_\Delta}}{v}. \tag{10}$$

Next, we will show that $a_r \geq \tau$. First, consider the case where $r \leq r_{\mathrm{I}}$. In this case, we have $a_r = 1 \geq \tau$. Next, suppose that $r > r_{\mathrm{I}}$. Then $\varepsilon_{r_\Delta} = 8\varepsilon_r \leq v \cdot a_r$, where the second inequality is where we used $r > r_{\mathrm{I}}$. Combining this with the above inequality, it must mean that

$$\tau < a_r.$$

Hence, regardless of the case, every arm a_r pulled is positive and hence provides useful information. Therefore, the fact that event $\mathcal{E}_{\mathrm{L}}$ happened implies that $\hat{\Delta}_r + \varepsilon_r$ in (9) is an upper confidence bound for Δ giving us

$$\Delta \leq \hat{\Delta}_r + \varepsilon_r. \tag{11}$$

As for the left-hand side, we have $\varepsilon_r = \varepsilon_{r_\Delta+3}$, which, by Lemma 19, is at most $\frac{\Delta}{4}$, giving us

$$\frac{\Delta}{2} + 2\varepsilon_r \leq \frac{\Delta}{2} + \frac{\Delta}{2} = \Delta. \tag{12}$$

Combining the left- and right-hand sides we get that

$$\frac{\Delta}{2} + 2\varepsilon_r \leq \Delta \leq \hat{\Delta}_r + \varepsilon_r$$

holds and therefore (8) holds. □

Lemma 21. *If $\tau \leq \frac{\Delta}{v}$, then on event $\mathcal{E}_{\mathrm{L}}$, Leftist will stop no later than epoch $r_\Delta + 3$.*

Proof. It suffices to show that if Leftist reaches epoch $r_\Delta + 3$, then the algorithm stops. Let $r = r_\Delta + 3$ and assume $\tau \leq \frac{\Delta}{v}$. Suppose Leftist reaches epoch r but does not stop. That must mean $\hat{\Delta}_r - \varepsilon_r < \varepsilon_r$ happened. As we assume event $\mathcal{E}_{\mathrm{L}}$ happened, Lemma 20 gives us a lower bound on $\hat{\Delta}_r - \varepsilon_r$, giving us

$$\frac{\Delta}{2} \leq \hat{\Delta}_r - \varepsilon_r < \varepsilon_r.$$

By Lemma 19, $\varepsilon_{r_\Delta+3} < \frac{\Delta}{4}$, and so

$$\frac{\Delta}{2} \leq \hat{\Delta}_r - \varepsilon_r < \frac{\Delta}{4},$$

which is a contradiction. □

Corollary 22. *If $\tau \leq \frac{\Delta}{v}$, then on event $\mathcal{E}_{\mathrm{L}}$, for any epoch r, either $\tau \leq a_r$ or Leftist has stopped before this epoch (i.e., $\rho < r$).*

Proof. Assume $\tau \leq \frac{\Delta}{v}$.

We first consider what happens when r_Δ does not exist. Then, for all $r \geq 0$, we have $\Delta \leq \varepsilon_r$. But then $\tau \leq \frac{\Delta}{v}$ implies that $\tau = 0$, and hence $\tau \leq a_r$ holds for all epochs from the positivity of a_r.

We now move on to the more interesting situation in which r_Δ exists. As we assume event $\mathcal{E}_{\mathrm{L}}$ happened, from Lemma 21, we know that Leftist will stop no later than epoch $r_\Delta + 3$. Since a_r is non-increasing, it suffices to show that $\tau \leq a_{r_\Delta+3}$. We consider two cases. First, suppose that $r_\Delta + 3 \leq r_{\mathrm{I}}$. Then we trivially have $\tau \leq a_{r_\Delta+3} = 1$. Next, suppose that $r_\Delta + 3 > r_{\mathrm{I}}$, so that we have $8\varepsilon_{r_\Delta+3} \leq v \cdot a_{r_\Delta+3}$. We also know that $\tau \leq \frac{\Delta}{v} < \frac{\varepsilon_{r_\Delta}}{v} = 8 \cdot \frac{\varepsilon_{r_\Delta+3}}{v} \leq \frac{v \cdot a_{r_\Delta+3}}{v} = a_{r_\Delta+3}$. Therefore, in this case, we have $\tau \leq a_r$ for all $r \leq r_\Delta + 3$. □

If the halting condition is not satisfied, then eventually a terminating condition, $\varepsilon_r < T^{-1/2}$, will happen and Leftist will commit to arm 0 thereafter.

A.3 Regret analysis for pulls from Leftist

This section bounds the regret contribution from the pulls made by Leftist.

Below, we heavily use the fact that $r^* := r_\Delta + 3 = \left\lceil \log_2 \frac{1}{\Delta} \right\rceil - 1$ and hence $r^* \leq \log_2 \frac{1}{\Delta}$. Recall that ρ is the stopping epoch. Intuitively speaking, we show that ρ is close to r^* with a high probability.

In the following, we derive the regret bound in several different cases. In particular, we consider whether $\Delta \leq v/16$ or not[4] as well as whether $\Delta > v\tau$ or not[5].

A.3.1 Case 1: $\Delta \leq v/16$, arm τ is optimal, and $\mathcal{E}_{\text{L}}$ happened

When arm τ is optimal, there are 2 regimes of interest[6]:

1. $\rho \leq r_{\max}$
2. $\rho = r_{\max}$

Due to our assumption that $v \geq \log(T) \cdot T^{-1/2}$, the last possible epoch is *after* Phase 1.

Recall that ρ is the stopping epoch for Leftist. In the below, we use the fact that from Lemma 21, we have $\rho \leq r_\Delta + 3$.

Regime 1: $\rho \leq r_{\max}$ Since $\mathcal{E}_{\text{L}}$ happened and $\Delta \leq v/16$, a reasoning that is essentially identical to that for Corollary 67 (for the multi-dimensional case) implies that $\rho \geq r_{\text{I}} + 1$ (i.e., the algorithm completes Phase 1). We begin by bounding the regret contribution from Phase 1. Leftist runs for at most $r_\Delta + 3 = \left\lceil \log_2 \frac{1}{\Delta} \right\rceil - 1$ epochs. In each epoch, we pull arms 0 and 1.

- The regret from pulling arm 0 in all rounds is at most

$$O\left(\log\left(\frac{1}{\delta}\right)\Delta \cdot \frac{1}{v^2}\right) = O\left(\log\left(\frac{1}{\delta}\right)\frac{1}{v}\right).$$

- The regret from pulling arm 1 in all rounds is of order at most

$$\log\left(\frac{1}{\delta}\right) v \cdot \frac{1}{v^2} = \log\left(\frac{1}{\delta}\right)\frac{1}{v}.$$

From the above and using $\delta = 1/T^2$, the regret is of order at most

$$\log(T) \cdot \frac{1}{v}. \tag{13}$$

To bound the regret contribution from Phase 2, we first observe that for $r \geq r_{\text{I}} + 1$, we have $a_r = 2^{r_{\text{I}} - r}$. Therefore, the

[4] The condition $\Delta \leq v/16$ implies that the halving of a_r begins (i.e., the algorithm completes Phase 1).

[5] The condition $\Delta > v\tau$ states that the arm 1 is optimal.

[6] It may seem odd that the regimes overlap; our point is that we will develop regret bounds for each regime, and it so happens that in the second regime of $\rho = r_{\max}$ we also can apply the regret bound for the first regime if desired.

regret contribution from Phase 2 can be bounded as

$$\begin{aligned}\sum_{r=r_{\mathrm{l}}+1}^{r_\Delta+3} n_r \cdot \left(\underbrace{\Delta}_{\text{arm } 0} + \underbrace{v \cdot a_r}_{\text{arm } a_r}\right) &= \sum_{r=r_{\mathrm{l}}+1}^{r_\Delta+3} n_r \cdot \left(\Delta + v \cdot 2^{r_{\mathrm{l}}-r}\right) \\ &= \sum_{r=r_{\mathrm{l}}+1}^{r_\Delta+3} n_r \cdot \left(\Delta + 2^{\log_2\lceil \frac{1}{v}\rceil} \cdot v \cdot 2^{-r}\right) \\ &\le 2 \sum_{r=r_{\mathrm{l}}+1}^{r_\Delta+3} n_r \cdot \left(\Delta + 2^{-r}\right) \\ &\le 2 \sum_{r=0}^{r_\Delta+3} n_r \cdot \left(\Delta + 2^{-r}\right) \\ &\le 2^6 (\log(1/\delta) \cdot \left(\frac{1}{\Delta} + \frac{1}{\Delta}\right),\end{aligned} \tag{14}$$

which is of order at most

$$(\log T) \cdot \frac{1}{\Delta}. \tag{15}$$

Hence, in this regime, the regret is bounded as

$$O\left(\log(T) \cdot \left(\frac{1}{v} + \frac{1}{\Delta}\right)\right) = O\left(\frac{\log T}{\Delta}\right). \tag{16}$$

We have just proved the following lemma.

Lemma 23. *Take $\delta = 1/T^2$. If $\Delta \le v/16$ and τ is optimal, then on event $\mathcal{E}_{\mathrm{L}}$, the pulls of Leftist contribute regret of order at most $\frac{\log T}{\Delta}$.*

Regime 2: $\boldsymbol{\Delta \le 16 \log(T) \cdot T^{-1/2}}$ Recall that the last possible epoch is $r_{\max} = \lfloor \log_2 \frac{\sqrt{T}}{\log T} \rfloor - 3$.

The analysis is like Regime 1, except we truncate the summation as:

$$\begin{aligned}\sum_{r=r_{\mathrm{l}}+1}^{r_{\max}} n_r \cdot \left(\underbrace{\Delta}_{\text{arm } 0} + \underbrace{v \cdot a_r}_{\text{arm } a_r}\right) &\le 2 \sum_{r=0}^{r_{\max}} n_r \cdot \left(\Delta + 2^{-r}\right) \\ &= O\left(\log(1/\delta) \cdot (\log^{-2}(T) \cdot T \cdot \Delta + \log^{-1}(T) \cdot \sqrt{T})\right) \\ &= O\left(\sqrt{T}\right),\end{aligned} \tag{17}$$

where the second equality uses $\Delta = O(\log(T) \cdot T^{-1/2})$.

Hence, we get regret at most (using $v \ge \log(T) \cdot T^{-1/2}$)

$$O\left(\log(T) \cdot \frac{1}{v} + \sqrt{T}\right) = O\left(\sqrt{T}\right). \tag{18}$$

We have just proved the following lemma.

Lemma 24. *Take $\delta = 1/T^2$. If $\Delta \le 16 \log(T) \cdot T^{-1/2}$ and τ is optimal, then on event $\mathcal{E}_{\mathrm{L}}$, the pulls of Leftist contribute regret of order at most $\sqrt{T}$.*

A.3.2 Case 2: $\Delta > v/16$, arm τ is optimal, and $\mathcal{E}_{\text{L}}$ happened

We give a direct analysis. Since many steps are similar to the previous analyses, we go more quickly. First, note that we still have from Lemma 21 that $\rho \le r_\Delta + 3$. The regret contribution is at most

$$\begin{aligned}
&\sum_{r=0}^{\min\{r_{\text{I}}, r^*\}} n_r \left(\Delta + v \cdot a_r\right) + \sum_{r=r_{\text{I}}+1}^{r^*} n_r \left(\Delta + v \cdot a_r\right) \\
&\le (\Delta + v) \sum_{r=0}^{\min\{r_{\text{I}}, r^*\}} n_r + 2 \sum_{r=r_{\text{I}}+1}^{r^*} n_r \left(\Delta + 2^{-r}\right) \\
&= O\left(\log(T) \cdot (\Delta + v) \min\left\{2^{2r_{\text{I}}}, 2^{2r^*}\right\} + \mathbf{1}\left[r_{\text{I}} < r^*\right] \cdot \log(T) \cdot \left(2^{2r^*}\Delta + 2^{r^*}\right)\right) \qquad (19) \\
&= O\left(\log(T) \cdot (\Delta + v) \min\left\{2^{2r_{\text{I}}}, 2^{2r^*}\right\} + \mathbf{1}\left[r_{\text{I}} < r^*\right] \cdot \log(T) \cdot \frac{1}{\Delta}\right). \qquad (20)
\end{aligned}$$

Next, we consider two sub-cases.

First, suppose that $r_{\text{I}} < r^*$. Then, then by unpacking the explicit value of each of r_{I} and r^*, it follows that $\Delta < \frac{v}{8}$ and hence $\frac{v}{16} < \Delta < \frac{v}{8}$. Hence, in this case, the regret is at most

$$\begin{aligned}
O\left(\log(T) \cdot \Delta \cdot 2^{2r_{\text{I}}} + \frac{\log T}{\Delta}\right) &= O\left(\frac{\Delta \log T}{v^2} + \frac{\log T}{\Delta}\right) \\
&= O\left(\frac{\log T}{\Delta}\right).
\end{aligned}$$

Suppose instead that $r_{\text{I}} \ge r^*$. Then we can drop the second term in the summation in (20) to get regret at most

$$O\left(\log(T) \cdot (\Delta + v) \cdot 2^{2r^*}\right) = O\left(\log(T) \cdot (\Delta + v) \cdot \frac{1}{\Delta^2}\right) = O\left(\frac{\log T}{\Delta}\right).$$

Hence, in this regime, the regret is bounded as

$$O\left(\frac{\log T}{\Delta}\right). \qquad (21)$$

We have just proved the following lemma.

Lemma 25. *Take $\delta = 1/T^2$. If $\Delta > v/16$ and τ is optimal, then on event $\mathcal{E}_{\text{L}}$, the pulls of Leftist contribute regret of order at most $\frac{\log T}{\Delta}$.*

A.3.3 Case 3: Arm 0 is optimal and $\mathcal{E}_{\text{L}}$ happened

The analysis in this case is simpler for two reasons. First, any pull of arm 0 gives no pseudoregret. Second, we do not provide any sort of guarantee that for any epoch $r \le \rho$ it holds that arm a_r is positive, and we therefore also do not provide any sort of guarantee that $\rho \le r^*$. Indeed, it can happen that arm τ has cost that is much greater than Δ, and in this situation the algorithm is likely to run for many epochs r for which a_r is negative, thereby preventing the algorithm for having informative estimates $\hat{\Delta}_r$ of Δ. Therefore, we only consider the all-encompassing regime that $\rho \le r_{\max}$ by bounding the regret as if the algorithm ran until epoch $r_{\max}$, which may be overcounting. The analysis is similar to Regime 2 from Case 1 above; for completeness, we describe how to modify the analysis and giving the corresponding regret bound.

We first consider the contribution from Phase 1. We only need consider the regret contribution from arm 1, but this still leads to the same order as (13), giving a regret contribution of order at most

$$\log(T) \cdot \frac{1}{v}.$$

In this regime, we get a regret bound whose order is the same as Regime 2 from Case 1 except that we can and do drop the contribution from arm 0 in the step (17), giving regret of order at most (using $v \ge \log(T) \cdot T^{-1/2}$)

$$\log(T) \cdot \frac{1}{v} + \sqrt{T} = \sqrt{T}. \qquad (22)$$

Note that being able to drop the aforementioned term is vital, as here we have no guarantee that $\Delta = O(\log(T) \cdot T^{-1/2})$.

We have just proved the following lemma.

Lemma 26. *Take $\delta = 1/T^2$. If arm 0 is optimal, then on event $\mathcal{E}_{\text{L}}$, the pulls of Leftist contribute regret of order at most $\sqrt{T}$.*

Combining the above lemmas yields the following, general regret bound for the pulls of Leftist.

Theorem 27. *Take $\delta = 1/T^2$. On event $\mathcal{E}_{\text{L}}$, the pulls of Leftist contribute regret of order at most $\sqrt{T}$.*

Proof (of Theorem 27). We begin by bounding the contribution to the regret from Leftist's pulls. We then consider the contribution from the other algorithms.

We consider several cases.

First, if arm 0 is optimal, the bound follows from Lemma 26.

Next, suppose that arm τ is optimal and $\Delta \leq v/16$. Then on the one hand, from Lemma 23, we have a bound of order at most

$$\frac{\log T}{\Delta};$$

when $\Delta \geq 16 \log(T) \cdot T^{-1/2}$, the above bound is of order at most the bound in the theorem. On the other hand, if $\Delta < 16 \log(T) \cdot T^{-1/2}$, then Lemma 24 implies the worst-case bound $\sqrt{T}$.

Finally, if arm τ is optimal and $\Delta > v/16$, then from Lemma 25, we have the bound

$$O\left(\frac{\log T}{\Delta}\right) = O\left(\frac{\log T}{v}\right),$$

which is of order at most $\sqrt{T}$ since $v \geq \log(T) \cdot T^{-1/2}$.

Hence, the result follows. □

A.4 Correctness analysis for NoisyBinarySearch

We analyze the correctness of NoisyBinarySearch, re-presented below for convenience.

Algorithm 8: NoisyBinarySearch

Input: Lower bound ε_r satisfying $\varepsilon_r \leq \Delta$, Left-most arm L, Right-most arm R

$N \leftarrow \frac{4 \log \frac{1}{\delta}}{\varepsilon_r^2}$
Make N pulls of arm L to get empirical mean $\hat{p}_{\text{left}}$
Make N pulls of arm R to get empirical mean $\hat{p}_{\text{right}}$
while $R - L \geq \varepsilon_{\text{NBS}}$ **do**
  $m \leftarrow \frac{L+R}{2}$
  Make N pulls of arm m to get empirical mean $\hat{p}_{\text{mid}}$
  if $\hat{p}_{\text{right}} - \hat{p}_{\text{mid}} \geq \varepsilon_r/2$ **then**
    $L \leftarrow m$, $\hat{p}_{\text{left}} \leftarrow \hat{p}_{\text{mid}}$
  else
    $R \leftarrow m$, $\hat{p}_{\text{right}} \leftarrow \hat{p}_{\text{mid}}$
return R

In NoisyBinarySearch (NBS), we are passed two parameters. The first is ε_r, which is a lower bound on a lower confidence bound for the probability gap Δ. The other parameter, a_r, is the right-most arm that will be used in NBS, R.

Lemma 28. *NBS runs for at most $\lfloor \log_2(2/\varepsilon_{\text{NBS}}) \rfloor$ iterations.*

Proof. Initially, $R - L \leq 1$. After iteration i, we have $R - L \leq 2^{-i}$. We seek the smallest integer i such that $2^{-i} < \varepsilon_{\text{NBS}}$, which is equivalent to $i > \log_2 \frac{1}{\varepsilon_{\text{NBS}}}$. Hence, the number of iterations is at most $\lfloor \log_2(1/\varepsilon_{\text{NBS}}) \rfloor + 1 = \lfloor \log_2(2/\varepsilon_{\text{NBS}}) \rfloor$. □

Each time the if statement on line 7 is reached, there is a possibility of making a mistake by incorrectly predicting the label of m. If this mistake does not happen, then upon the subsequent update of L or R, we have the L is negative and R is positive, as desired. The next lemma bounds the probability of such a mistake.

Lemma 29. *Consider a given iteration i of NBS. Suppose that in this iteration, arm R is positive. Then on event $\mathcal{E}_{\mathrm{L}}$, the probability that NBS incorrectly predicts the label of m on line 7 is at most δ.*

Proof. To better match the notation of the algorithm, we use $r := \rho$, where we recall that ρ is the stopping epoch for Leftist. Each time the if statement on line 7 is reached, we have the possibility of two mistakes occurring. Let $\bar{X} = \hat{p}_{\mathrm{right}} - \hat{p}_{\mathrm{mid}}$. We have the case where $\bar{X} \geq \frac{\varepsilon}{2}$, but $\mathsf{E}[\bar{X}] = 0$. In words, this means that we think that m and R have different labels, but in actuality, they have the same label. The other case is where $\bar{X} < \frac{\varepsilon_r}{2}$, but $\mathsf{E}[\bar{X}] = \Delta$. In words, this means that we think that m and R have the same label, but in actuality, they have different labels.

Using Hoeffding's inequality with appropriate conditioning on $\mathsf{E}[\bar{X}]$ (since m and R are fixed, conditional on the samples drawn from the previous iterations of NBS), we bound the probability of each type of mistake in turn.

We have[7]

$$\begin{aligned}\Pr\left(\bar{X} \geq \frac{\varepsilon_r}{2} \mid \mathsf{E}[\bar{X}] = 0\right) &= \Pr\left(\bar{X} - \mathsf{E}[\bar{X}] \geq \frac{\varepsilon_r}{2} \mid \mathsf{E}[\bar{X}] = 0\right) \\ &\leq e^{-2(2N)(\varepsilon_r/2)^2/2^2} \\ &= e^{-N\varepsilon_r^2/4}\end{aligned}$$

and

$$\begin{aligned}\Pr\left(\bar{X} < \frac{\varepsilon_r}{2} \mid \mathsf{E}[\bar{X}] = \Delta\right) &= \Pr\left(\bar{X} - \mathsf{E}[\bar{X}] < \frac{\varepsilon_r}{2} - \Delta \mid \mathsf{E}[\bar{X}] = \Delta\right) \\ &\leq \Pr\left(\bar{X} - \mathsf{E}[\bar{X}] < \frac{\varepsilon_r}{2} - \varepsilon_r \mid \mathsf{E}[\bar{X}] = \Delta\right) \\ &= \Pr\left(\bar{X} - \mathsf{E}[\bar{X}] < -\frac{\varepsilon_r}{2} \mid \mathsf{E}[\bar{X}] = \Delta\right) \\ &\leq e^{-2(2N)(-\varepsilon_r/2)^2/2^2} \\ &\leq e^{-N\varepsilon_r^2/4},\end{aligned}$$

where the first inequality uses the fact that $\varepsilon_r \leq \hat{\Delta}_r - \varepsilon \leq \Delta$ under event $\mathcal{E}_{\mathrm{L}}$.

We will only be concerned with one of these probabilities of an error occurring for each iteration, and since the two bounds are equal, the probability of a mistake occurring during a single iteration is at most $e^{-N\varepsilon_r^2} = \delta$. □

Because the lemma below will also find use in our multi-dimensional analysis, we make its presentation more general by using a projection onto the separating hyperplane (which, in the one-dimensional case, is simply $\{\tau\}$). To this end, for any arm a, let $\pi(a)$ be the Euclidean projection of a onto the hyperplane; that is,

$$\pi(a) = \operatorname*{arg\,min}_{a' \in H \cap [0,1]^d} \|a' - a\|_2.$$

Lemma 30. *On event $\mathcal{E}_{\mathrm{L}}$, with probability at least $1 - T \cdot \delta$, NBS will return a positive arm a^f satisfying $\|a^f - \pi(a^f)\|_2 < \varepsilon_{\mathrm{NBS}}$.*

Proof. First, suppose that NBS did not make a mistake in any iteration. Then since L is negative and R is positive at the end of the last iteration of NBS, the closed line segment

$$[L, R] := \{a = \alpha L + (1 - \alpha)R \colon \alpha \in [0, 1]\}$$

must contain an arm on the hyperplane. Since, at the end of its final iteration, NBS returns R and we have $\|R - L\|_2 < \varepsilon_{\mathrm{NBS}}$, it follows that arm a^f is positive and $\|a^f - \pi(a^f)\|_2 < \varepsilon_{\mathrm{NBS}}$, as desired.

It remains to control the probability that NBS did not make a mistake in any iteration. First, observe that if NBS did not make a mistake in any iteration prior to iteration i, then arm R is positive in iteration i and we hence can apply Lemma 29. Now, we use a union bound over the per-iteration failure probability upper bound given by Lemma 29, where we take a union bound over the (by grossly overcounting!) at most T iterations. □

[7] Note that $\bar{X}$ is twice the empirical average of $2N$ samples since each empirical mean $\hat{p}_{\mathrm{right}}$ and $\hat{p}_{\mathrm{mid}}$ is divided by only N. This leads to the extra division by 2^2 in the exponent below.

A.5 Complete regret analysis for one-dimensional case

The regret bounds in Section A.1 depend on several pieces: the regret incurred by the pulls from Leftist and, in case they are called, the regret incurred from the pulls of NoisyBinarySearch and from UCB. We now give a detailed analysis to show how each piece contributes to the regret.

A.5.1 Overview of total regret analysis

We use the abbreviation NBS (NoisyBinarySearch). There are two situations. Either Leftist calls NBS, or it does not. The former case is more complicated; we consider it first.

We begin with a description of the places where regret can be accumulated (in addition to regret from the pulls of Leftist):

1. Leftist calls NBS after detecting a lower bound on Δ. The result is $\hat{\tau}$.
2. Leftist calls UCB.

We begin by bounding the regret of NBS and UCB.

Recall that ρ is the stopping epoch for Leftist; this is the epoch in which Leftist calls NBS (if at all such a call happens).

A.5.2 Regret of NoisyBinarySearch

On event $\mathcal{E}_{\text{L}}$, we have $\varepsilon_r \leq \Delta$ and hence $a_r \leq \frac{16\varepsilon_r}{v} \leq \frac{16\Delta}{v}$ (we note that if r belongs to Phase 1, then the factor of 16 can be improved to 8).

Lemma 31. *On event $\mathcal{E}_{\text{L}}$, the instantaneous regret for any arm pulled in NBS is always upper bounded by* 16Δ.

Proof. Let us look at the case of line 6 of NBS. For this, we will only be looking at the instantaneous regret of m, which provides us with four different cases to analyze and bound.

- Case 1: $m \geq \tau$ and arm 0 is optimal:
$$(p_0) - (p_1 - m) = m - \Delta \leq v \cdot a_r \leq 16\Delta;$$
- Case 2: $m \geq \tau$ and arm τ is optimal:
$$(p_1 - \tau) - (p_1 - m) = m - \tau \leq v \cdot a_r \leq 16\Delta;$$
- Case 3: $m < \tau$ and arm 0 is optimal:
$$(p_0) - (p_0 - m) = m \leq v \cdot a_r \leq 16\Delta;$$
- Case 4: $m < \tau$ and arm τ is optimal:
$$(p_1 - \tau) - (p_0 - m) = \Delta - \tau + m \leq \Delta.$$

□

Although m changes each round, we can just keep using the upper bound of a_r for m. This is because of the scenario where τ is very close to R, in which case m keeps replacing L each round and moving to the right (and hence is approaching a_r).

From Lemma 31, the total regret for each arm that is pulled during NBS is upper bounded by $N \cdot (v \cdot a_r) \leq 16N\Delta$. This means that for each iteration of the while loop, we accumulate at most $16N\Delta$ regret due to line 6 of NBS.

Lemma 32. *On event $\mathcal{E}_{\text{L}}$, the regret of NBS is at most*

$$O\left(\frac{\log(1/\delta)\log(1/\varepsilon_{\text{NBS}})}{\Delta}\right).$$

Proof. Noting that r in NBS will be equal to the stopping epoch ρ, the number of pulls in each iteration of NBS is $O(\log(1/\delta) \cdot \frac{1}{\Delta^2})$ since $N = O(n_\rho) = O(\log(1/\delta) \cdot 2^{2\rho})$ and we have from Lemma 21 that $\rho \asymp r_\Delta \asymp \log_2 \frac{1}{\Delta}$. From Lemma 28, the number of iterations of NBS is at most $\log(1/\varepsilon_{\text{NBS}})$. Hence, each call to NBS results in $O\left(\frac{\log(1/\delta)\cdot\log(1/\varepsilon_{\text{NBS}})}{\Delta^2}\right)$ pulls.

From Lemma 31, it follows that on event $\mathcal{E}_{\text{L}}$, the regret contribution from NBS is at most

$$O\left(\frac{\log(1/\delta) \cdot \log(1/\varepsilon_{\text{NBS}})}{\Delta^2}\right) \cdot 16\Delta = O\left(\frac{\log(1/\delta) \cdot \log(1/\varepsilon_{\text{NBS}})}{\Delta}\right).$$

□

A.5.3 Regret of UCB on two-arm problem

Before bounding the regret of UCB in our setting, we introduce one more event.

Let event $\mathcal{E}_{\text{NBS-L}}$ be the event that NoisyBinarySearch, when called from Leftist, returns a positive arm satisfying $|\hat{\tau} - \tau| \leq \varepsilon_{\text{NBS}}$.

Lemma 33 (Regret of UCB). *On the event $\mathcal{E}_{\text{L}} \cap \mathcal{E}_{\text{NBS-L}}$, the regret of UCB is at most*

$$O\left(\min\left\{\frac{\log T}{|\Delta - \tau \cdot v|}, \sqrt{T \log T}\right\}\right).$$

Proof. UCB is run on arms 0 and $\hat{\tau}$, where $\hat{\tau}$ is the arm returned by NoisyBinarySearch. We therefore first study the expected reward of arm $\hat{a}$. Since event $\mathcal{E}_{\text{NBS-L}}$ happened, arm $\hat{\tau}$ is positive and satisfies $|\tau - \hat{\tau}| \leq \varepsilon_{\text{NBS}} = \frac{1}{T}$.

Recalling our notation, $\mu_c(0) = p_0$ is the expected reward of arm 0 and $\mu_c(\hat{\tau})$ is the expected reward of arm $\hat{\tau}$. We may now conclude that $|\mu_c(\hat{\tau}) - \mu_c(0)| \geq \min\{|\Delta - v \cdot \tau|, |\Delta - v \cdot \tau - 1/T|\}$.

We have from Auer et al. (2002) that the regret of UCB with respect to the set of arms $\{0, \hat{\tau}\}$ is

$$\begin{aligned}
&O\left(\min\left\{\frac{\log T}{|\mu_c(0) - \mu_c(\hat{\tau})|}, \sqrt{T \log T}\right\}\right) \\
&= O\left(\min\left\{\frac{\log T}{\min\{|\Delta - v \cdot \tau|, |\Delta - v \cdot \tau - 1/T|\}}, \sqrt{T \log T}\right\}\right).
\end{aligned}$$

Note that in the above bound, whenever the $\frac{1}{T}$ has a nontrivial effect on the second term in the inner minimum, the $\sqrt{T \log T}$ term must be the smaller term. Hence, we fortunately can present the simplified bound:

$$O\left(\min\left\{\frac{\log T}{|\Delta - v \cdot \tau|}, \sqrt{T \log T}\right\}\right).$$

Finally, as we instead want the regret with respect to the set of arms $\{0, \tau\}$, we account for the difference between the expected reward of τ and $\hat{\tau}$. But this is at most $\frac{1}{T}$ and hence can contribute only a constant to the regret. □

A.5.4 Proofs of Theorems 17 and 18

Proof (of Theorem 17). From Lemma 65, event $\mathcal{E}_{\text{L}}$ happens with probability at least $1 - 1/T$. In the sequel, consider the case that $\mathcal{E}_{\text{L}}$ happens (if it does not happen, we pick up a regret contribution of at most $T \cdot 1/T = 1$).

To bound the regret due to the pulls of Leftist, we use Lemma 23 in the case that $\Delta \leq v/16$ and Lemma 25 in the case that $\Delta > v/16$; both lemmas give the same bound. In addition, Lemmas 32 and 33 respectively imply that the regret is at most

$$\begin{aligned}
&[\text{Leftist}] + [\text{NBS}] + [\text{UCB}] \\
&= O\left(\frac{\log T}{\Delta}\right) + O\left(\frac{\log^2 T}{\Delta}\right) + O\left(\min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v \cdot \tau|}\right\}\right) \\
&= O\left(\frac{\log^2 T}{\Delta}\right) + O\left(\min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v \cdot \tau|}\right\}\right).
\end{aligned}$$

□

Proof (of Theorem 18). For both results, we use Theorem 27 to bound the regret due to the pulls of Leftist.

We now consider the remaining analysis for each regret bound in turn.

Beginning with the first regret bound, note that NoisyBinarySearch and UCB can be called only if $\Delta = \Omega(\log(T) \cdot T^{-1/2})$. To see this, observe from Lemma 66 that $\rho \geq r_\Delta + 1$. Hence, NoisyBinarySearch can be called only if $r_\Delta + 1 \leq r_{\max}$, a condition which implies that $\Delta = \Omega(\log(T) \cdot T^{-1/2})$. We may then take the regret bounds for other algorithms directly from the proof of Theorem 17 together with the substitution $\Delta = \Omega(\log(T) \cdot T^{-1/2})$ (except for the part for UCB, where the substitution is unnecessary) to get:

$$\begin{aligned}
&[\text{Leftist}] + [\text{NBS}] + [\text{UCB}] \\
&= O\left(\sqrt{T}\right) + O\left(\log(T)\sqrt{T}\right) + O\left(\min\left\{\sqrt{T\log T}, \frac{\log T}{|\Delta - v \cdot \tau|}\right\}\right) \\
&= O\left(\log(T)\sqrt{T}\right) + O\left(\min\left\{\sqrt{T\log T}, \frac{\log T}{|\Delta - v \cdot \tau|}\right\}\right).
\end{aligned}$$

We now turn to the second regret bound, for which we assume that $\Delta \leq \log(T) \cdot \frac{T^{-1/2}}{2}$. We claim that with probability at least $1 - 1/T$, NoisyBinarySearch will not be called and hence Leftist will commit to arm 0. From Lemma 66, it holds on event $\mathcal{E}_{\text{L}}$ that $\rho \geq r_\Delta + 1$. Moreover, the condition $\Delta \leq \log(T) \cdot \frac{T^{-1/2}}{2}$ implies that $r_\Delta + 1 > r_{\max}$. Finally, observe that from Lemma 65, event $\mathcal{E}_{\text{L}}$ happens with probability at least $1 - 1/T$.

Consequently, the regret is at most

$$\begin{aligned}
&[\text{Leftist}] + [\text{commit}] \\
&= O\left(\sqrt{T}\right) + O(\log(T) \cdot \sqrt{T}) \\
&= O\left(\log(T) \cdot \sqrt{T}\right).
\end{aligned}$$

□

B Analysis for multi-dimensional case

B.1 Regret bounds for multi-dimensional case

For the convenience of the reader, we begin by re-presenting Theorems 14 and 15, our regret bounds for the multi-dimensional case.

First, we present a problem-dependent regret bound in the case that MD-Leftist stops in Phase 2.

Theorem 34 (Theorem 14 from the main text)**.** *Let Assumption 57 be satisfied and take* $\varepsilon_{\text{NBS}} = \theta_{\min}/(d^3T^2)$ *and* $\delta = 1/T^2$*. Suppose that* H *is optimal. If* $16T^{-1/2} < \Delta \leq v_{\min}/16$*, then the regret of MD-Leftist is bounded as*

$$\mathcal{R}_T = O\left(\log(T) \cdot \left(\frac{\Delta + \|v\|_1}{v_{\min}^2} + \frac{\max\left\{d\log(1/\varepsilon_{\text{NBS}}), d^2\log d\right\}}{\Delta}\right) + \min\left\{\sqrt{T\log T}, \frac{\log T}{|\Delta - v^*|}\right\}\right),$$

where v^* *is the cost of the minimum-cost positive arm.*

Note that the condition $16T^{-1/2} \leq \Delta$ is not necessary for the proof of Theorem 14; on the other hand, for small enough Δ, it is more sensible to use the next theorem.

This next theorem applies more generally. It is particularly useful for handling the case that arm $\mathbf{0}$ is optimal or the case when Δ is small.

Theorem 35 (Theorem 15 from the main text)**.** *Let Assumption 57 be satisfied and take* $\varepsilon_{\text{NBS}} = \theta_{\min}/(d^3T^2)$ *and* $\delta = 1/T^2$*. If* $\Delta \leq v_{\min}/16$*, then the regret of MD-Leftist is bounded as*

$$\mathcal{R}_T = O\left(\log(T) \cdot \frac{\Delta + \|v\|_1}{v_{\min}^2} + \max\left\{\sqrt{d}\log(1/\varepsilon_{\text{NBS}}), d^{1.5}\log d\right\}\sqrt{T} + \min\left\{\sqrt{T\log T}, \frac{\log T}{|\Delta - v^*|}\right\}\right).$$

where v^* *is the cost of the minimum-cost positive arm.*

If it further holds that $\Delta \leq \frac{1}{2}\log(T)\sqrt{\frac{d}{T}}$*, then the bound can be improved to*

$$\mathcal{R}_T = O\left(\log(T) \cdot \left(\frac{1}{v_{\min}^2} \cdot (\Delta + \|v\|_1) + \sqrt{Td}\right)\right).$$

We present proofs of the above theorems in Section B.6.4.

B.2 Preliminaries

For an axis-parallel rectangle A, let $R(A)$ be the vertex for which all coordinates are maximized. This is the multi-dimensional analogue of the "right-most" point of a closed interval (a one-dimensional axis-parallel rectangle).

Central to Multi-dim Leftist's operation is the following type of axis-parallel hyperrectangle. Let A_r be the axis-parallel hyperrectangle whose j^{th} side is $2^{-r} \cdot v_{\min} \cdot [0, \frac{1}{v_j}]$, for $v_{\min} = \min\{\min_j v_j, 1\}$. We remark that A_0 is obtained by starting with a hyperrectangle whose sides are $\frac{1}{v_j}$ and scaling down this rectangle (if needed) until it is contained within the unit hypercube. Notice that the cost of $R(A_r)$ is equal to

$$\langle v, R(A_r)\rangle = v_{\min} \cdot d \cdot 2^{-r}, \tag{23}$$

which, in the case of $v = \mathbf{1}$, reduces to $d \cdot 2^{-r}$.

We first establish an important property.

Proposition 36. *For any* $r \geq 0$*, if* $R(A_r)$ *is negative, then any arm with cost at most* $2^{-r} \cdot v_{\min}$ *also must be negative.*

Proof. We will show that $R(A_r)$ coordinate-wise dominates any arm with cost $2^{-r} \cdot v_{\min}$, after which the claim immediately follows. Suppose for a contradiction that there is an arm a with cost $2^{-r} \cdot v_{\min}$ such that, for a dimension $j \in [d]$, we have $a_j > [R(A_r)]_j = 2^{-r} \cdot \frac{v_{\min}}{v_j}$. But this implies that $\langle v, a_j\rangle \geq v_j \cdot a_j > 2^{-r} \cdot v_{\min}$, a contradiction. □

Without loss of generality, we may assume that θ^* is a unit vector. Since $\langle \theta^*, \mathbf{1}\rangle \geq \tau$, it follows that $\|\theta^*\|_1 \geq \tau$. Therefore, $\tau \leq \sqrt{d}$. Moreover, since $\langle \theta^*, \mathbf{0}\rangle < \tau$, we have that $\tau > 0$.

B.3 Analysis for MD-Leftist

We first re-present MultiDimLeftist (MD-Leftist). Although the presentation looks different, functionally the algorithm is the same as Algorithm 3 from the main text. Also, the pseudocode contains some helpful comments.

Algorithm 9: MultiDimLeftist (MD-Leftist)

$\phi = 1$ // start in Phase 1
$r \leftarrow 0$
$a_0 \leftarrow \mathbf{1}$ // ones vector, $\mathbf{1} \in \mathbb{R}^d$
$\varepsilon_0 \leftarrow \frac{1}{8}$
$n_0 \leftarrow \frac{\log \frac{2}{\delta}}{2\varepsilon_0^2}$
while $\varepsilon_r \geq \log(T)\sqrt{\frac{d}{T}}$ **do**
  Make n_r pulls of arm $\mathbf{0}$ to get empirical mean $\hat{p}_{\mathbf{0}}$ // zeros vector, $\mathbf{0} \in \mathbb{R}^d$
  Make n_r pulls of arm a_r to get empirical mean $\hat{p}_{a_r}$
  $\hat{\Delta}_r \leftarrow \hat{p}_{a_r} - \hat{p}_0$
  if $\hat{\Delta}_r - \varepsilon_r \geq \varepsilon_r$ **then**
    $a^f \leftarrow$ MultiEstTau(ε_r, a_r)
    For remaining rounds, run UCB on arms $\mathbf{0}$ and a^f
  else
    if $\phi = 2$ **then**
      $a_{r+1} \leftarrow \frac{1}{2} \cdot a_r$
    else
      if $\langle v, R(A_0)\rangle \geq d \cdot 8\varepsilon_r$ // Note: $\langle v, R(A_0)\rangle \geq d \cdot 8\varepsilon_r \Leftrightarrow v_{\min} \geq 2^{-r} \Leftrightarrow r \geq \ell$
      then
        $a_{r+1} \leftarrow R(A_1)$
        $\phi = 2$ // switch to Phase 2
      else
        $a_{r+1} \leftarrow a_r$
      $\varepsilon_{r+1} \leftarrow \varepsilon_r/2$
      $n_{r+1} \leftarrow \frac{\log \frac{2}{\delta}}{2\varepsilon_{r+1}^2}$
      $r \leftarrow r + 1$
Commit to arm $\mathbf{0}$

B.3.1 Preliminaries

Recall that by definition of $v_{\min}$, we always have $v_{\min} \leq 1$. Define the quantity $\ell := \lceil \log_2 \frac{1}{v_{\min}} \rceil$; note that epoch ℓ is the last epoch of Phase 1. For the convenience of the reader, we mention that we also have $-\ell = \lfloor \log_2 v_{\min} \rfloor$; we will use this form of ℓ in the proof of Lemma 37 below.

Recall that we say H is optimal if there exists an arm on the separating hyperplane that is optimal; if H is not optimal, then arm $\mathbf{0}$ is optimal.

We define ρ to be the stopping epoch of MultiDimLeftist (MD-Leftist); this is either the epoch in which MultiTauEst is called or, if the former is never called, the largest possible epoch $r_{\max} := \left\lfloor \log_2 \frac{\sqrt{T}}{\log(T)\sqrt{d}} \right\rfloor - 3$. When H is optimal, we can show that with high probability, ρ is no greater than 3 epochs after the critical epoch r_Δ (which we recall from Appendix A is defined as $r_\Delta = \max_{r \geq 0}\{\varepsilon_r > \Delta\}$) .

Before beginning the main analysis, it will be useful to introduce two events. First, let $\mathcal{E}_{\text{MD-L}}$ be the event that both of the following are true:

- In MD-Leftist, for all epochs $r \leq \rho$ such that arm a_r is positive, $|\hat{\Delta}_r - \Delta| \leq \varepsilon_r$;
- In MD-Leftist, for all epochs $r \leq \rho$, it holds that $\hat{\Delta}_r - \varepsilon_r \leq \Delta$.

Next, let $\mathcal{E}_{\text{stop}}$ be the event that $\ell + 1 \le \rho \le r_\Delta + 3$. Informally, event $\mathcal{E}_{\text{stop}}$ corresponds to the situation that MD-Leftist stops in Phase 2 but not much later than epoch $\lceil \log_2 \frac{1}{\Delta} \rceil$.

B.3.2 Correctness analysis

We need to ensure that the lower confidence bound used by MD-Leftist is not too large.

Lemma 37. *Let $r^* = r_\Delta + 3$ and assume H is optimal. Then for all epochs $r \le \min\{r^*, \rho\}$, arm a_r is positive. Also, on event $\mathcal{E}_{\text{MD-L}}$, the lower confidence bound of Δ is lower bounded as*

$$\frac{\Delta}{2} \le \hat{\Delta}_{r^*} - \varepsilon_{r^*}. \tag{24}$$

Proof. We first establish that in epoch $r^* = r_\Delta + 3$, arm a_{r^*} is positive. Note that establishing this claim implies that a_r is positive for any $r \le r^*$ since any arm from an epoch previous to epoch r^* coordinate-wise dominates arm a_{r^*}.

First, consider the case where $r^* \le \ell$. In this case, we have $a_{r^*} = \mathbf{1}$, which is positive by assumption. Next, suppose that $r^* \ge \ell + 1$. Observe that in epoch r_Δ, we have $\Delta < \varepsilon_{r_\Delta} = 2^{-r_\Delta - 3} = 2^{-(r_\Delta + 3)}$, which can be upper bounded as

$$v_{\min} \cdot 2^{-(r_\Delta + 3 + \log_2 v_{\min})} \le v_{\min} \cdot 2^{-(r_\Delta + 3 + \lfloor \log_2 v_{\min} \rfloor)}.$$

Since H is optimal, we therefore have that there is a positive arm with cost at most $v_{\min} \cdot 2^{-\bar{r}}$ for

$$\bar{r} := r_\Delta + 3 + \lfloor \log_2 v_{\min} \rfloor.$$

Therefore, from the contrapositive[8] of Proposition 36, $R(A_{\bar{r}})$ must be positive. But this means that if we pull arm $R(A_{\bar{r}})$ (or larger) in epoch $r_\Delta + 3$, then we are pulling a positive arm in this epoch and hence are done. Therefore, it suffices to ensure that arm a_{r^*} is equal to $R(A_{r'})$ for some $r' \le r^* + \lfloor \log_2 v_{\min} \rfloor = r^* - \ell$. But this is true by definition of the algorithm since epoch $\ell + 1$ is the first epoch of Phase 2, and in this epoch we have $a_{\ell+1} = R(A_1)$; moreover, again by definition of the algorithm, for any epoch $r^* \ge \ell + 1$, we have that $a_{r^*} = R(A_{r^* - \ell})$.

Hence, in either case, arm a_{r^*} is positive. This fact will be used in proving the second part of the lemma, which we now do. Note that (24) is equivalent to

$$\frac{\Delta}{2} + 2\epsilon_{r^*} \le \hat{\Delta}_{r^*} + \epsilon_{r^*}. \tag{25}$$

We will show that

$$\frac{\Delta}{2} + 2\epsilon_{r^*} \le \Delta \le \hat{\Delta}_{r^*} + \epsilon_{r^*}, \tag{26}$$

after which (25) follows.

We first establish the right inequality of (26) by noting that the RHS of (26) is an upper confidence bound for Δ; indeed, since we have shown that a_{r^*} is positive, then the fact that event $\mathcal{E}_{\text{MD-L}}$ happened implies that $\hat{\Delta}_{r^*} + \epsilon_{r^*}$ is an upper confidence bound for Δ, as desired.

As for the left inequality of (26), we know that $\varepsilon_{r^*} = \varepsilon_{r_\Delta + 3}$ which, by Lemma 19, is at most $\frac{\Delta}{4}$, giving us

$$\frac{\Delta}{2} + 2\epsilon_{r^*} \le \frac{\Delta}{2} + \frac{\Delta}{2} = \Delta. \tag{27}$$

We have thus proved (26). □

Lemma 38. *If H is optimal, then on event $\mathcal{E}_{\text{MD-L}}$, MD-Leftist will stop no later than epoch $r_\Delta + 3$.*

Proof. It suffices to show that if MD-Leftist reaches epoch $r_\Delta + 3$, then the algorithm stops. Let $r = r_\Delta + 3$ and assume that H is optimal. Suppose MD-Leftist reaches epoch r but does not stop. That must mean $\hat{\Delta}_r - \epsilon_r < \varepsilon_r$ happened. As we assume event $\mathcal{E}_{\text{MD-L}}$ happened, Lemma 37 gives us a lower bound on $\hat{\Delta}_r - \epsilon_r$, so that

$$\frac{\Delta}{2} \le \hat{\Delta}_r - \epsilon_r < \varepsilon_r.$$

[8] For convenience, we state the contrapositive form of Proposition 36: "Suppose that there is a positive arm with cost at most $\Delta \le 2^{-r} \cdot v_{\min}$. Then arm $R(A_r)$ is positive."

By Lemma 19, $\varepsilon_{r_\Delta+3} < \frac{\Delta}{4}$, and so

$$\frac{\Delta}{2} \le \hat{\Delta}_r - \varepsilon_r < \frac{\Delta}{4},$$

which is a contradiction. □

Corollary 39. *If H is optimal, then on event $\mathcal{E}_{\text{MD-L}}$, for any epoch r, either arm a_r is positive or MD-Leftist stopped before this epoch (i.e., $\rho < r$).*

Proof. The proof is a direct consequence of the first part of Lemma 37 together with Lemma 38. From the first part of Lemma 37, if the algorithm is in an epoch $r \le r^* := r_\Delta + 3$, then arm a_r is positive. Next, Lemma 38 implies that if the algorithm reaches epoch r^*, then on event $\mathcal{E}_{\text{MD-L}}$, it stops in that epoch. □

Finally, if r_Δ is not defined, then eventually MD-Leftist's terminating condition, $\varepsilon_r < T^{-1/2}$ will be satisfied, after which MD-Leftist will commit to arm $\mathbf{0}$.

B.4 Regret analysis for pulls from MD-Leftist

This section bounds the regret contribution from the pulls made by MD-Leftist. The regret due to the other algorithms is handled in Section B.6.

Below, we heavily use the fact that $r^* := r_\Delta + 3 = \left\lceil \log_2 \frac{1}{\Delta} \right\rceil - 1$ and hence $r^* \le \log_2 \frac{1}{\Delta}$.

Recall that ρ is the stopping epoch.

B.4.1 Case 1: $\Delta \le v_{\min}/16$, H is optimal, and $\mathcal{E}_{\text{MD-L}}$ happened

When H is optimal and event $\mathcal{E}_{\text{stop}}$ happened, there are 2 regimes of interest:

1. $\rho \le r_{\max}$
2. $\rho = r_{\max}$

We assume that $v_{\min} \ge T^{-1/2}$, so that the last possible epoch is *after* Phase 1.

Recall that ρ is the stopping epoch for MD-Leftist. In the below, we use the fact that from Lemma 38, with probability at least $1 - 1/T$, we have $\rho \le r_\Delta + 3$.

Regime 1: $\rho \le r_{\max}$ Corollary 67 implies that event $\mathcal{E}_{\text{stop}}$ occurs and hence $\rho \ge \ell + 1$ (i.e., the algorithm completes Phase 1). We begin by bounding the regret contribution from Phase 1. MD-Leftist runs for at most $r_\Delta + 3 = \left\lceil \log_2 \frac{1}{\Delta} \right\rceil - 1$ epochs. In each epoch, we pull arms $\mathbf{0}$ and $\mathbf{1}$.

- The regret from pulling arm $\mathbf{0}$ in all rounds is of order at most

$$\left(\log \frac{1}{\delta}\right) \Delta \cdot \frac{1}{v_{\min}^2}.$$

- The regret from pulling arm $\mathbf{1}$ in all rounds is of order at most

$$\left(\log \frac{1}{\delta}\right) \|v\|_1 \cdot \frac{1}{v_{\min}^2}.$$

From the above and using $\delta = 1/T^2$, the regret is of order at most

$$\log(T) \cdot \frac{\Delta + \|v\|_1}{v_{\min}^2}. \tag{28}$$

To bound the regret contribution from Phase 2, we first observe that for $r \geq \ell + 1$, we have $a_r = R(A_{r-\ell})$, and hence from (23), it follows that $\langle v, a_r \rangle = v_{\min} \cdot d \cdot 2^{\ell - r}$. Therefore, the regret contribution from Phase 2 can be bounded as

$$\begin{aligned}
\sum_{r=\ell+1}^{r_\Delta+3} n_r \cdot \left(\underbrace{\Delta}_{\text{arm } \mathbf{0}} + \underbrace{\langle v, a_r \rangle}_{\text{arm } a_r} \right) &= \sum_{r=\ell+1}^{r_\Delta+3} n_r \cdot \left(\Delta + v_{\min} \cdot d \cdot 2^{\ell - r} \right) \\
&= \sum_{r=\ell+1}^{r_\Delta+3} n_r \cdot \left(\Delta + 2^{\log_2 \left\lceil \frac{1}{v_{\min}} \right\rceil} \cdot v_{\min} \cdot d \cdot 2^{-r} \right) \\
&\leq 2 \sum_{r=\ell+1}^{r_\Delta+3} n_r \cdot \left(\Delta + d \cdot 2^{-r} \right) \qquad (29) \\
&\leq 2 \sum_{r=0}^{r_\Delta+3} n_r \cdot \left(\Delta + d \cdot 2^{-r} \right) \\
&\leq 2^6 (\log(1/\delta) \cdot \left(\frac{1}{\Delta} + \frac{d}{\Delta} \right),
\end{aligned}$$

which is of order at most

$$(\log T) \cdot \frac{d}{\Delta}. \qquad (30)$$

Hence, in this regime, we get regret of order at most

$$\log(T) \cdot \left(\frac{\Delta + \|v\|_1}{v_{\min}^2} + \frac{d}{\Delta} \right). \qquad (31)$$

We have just proved the following lemma.

Lemma 40. *Take $\delta = 1/T^2$. If $\Delta \leq v_{\min}/16$ and H is optimal, then on event $\mathcal{E}_{\text{MD-L}}$, the pulls of MD-Leftist contribute regret of order at most*

$$\log(T) \cdot \left(\frac{\Delta + \|v\|_1}{v_{\min}^2} + \frac{d}{\Delta} \right).$$

Regime 2: $\boldsymbol{\Delta \leq 16 \log(T) \cdot \sqrt{\frac{d}{T}}}$ Recall that the last possible epoch is $r_{\max} = \left\lfloor \log_2 \frac{\sqrt{T}}{\log(T)\sqrt{d}} \right\rfloor - 3$.

The analysis is like Regime 1, except we truncate the summation as:

$$\begin{aligned}
\sum_{r=\ell+1}^{r_{\max}} n_r \cdot \left(\underbrace{\Delta}_{\text{arm } \mathbf{0}} + \underbrace{\langle v, a_r \rangle}_{\text{arm } a_r} \right) &\leq 2 \sum_{r=0}^{r_{\max}} n_r \cdot \left(\Delta + d \cdot 2^{-r} \right) \\
&= O\left(\log(1/\delta) \cdot (\log^{-2}(T) \cdot (T/d) \cdot \Delta + d \log^{-1}(T) \cdot \sqrt{T/d}) \right) \qquad (32) \\
&= O\left(\log(1/\delta) \cdot (\log^{-1}(T) \cdot \sqrt{T/d} + \log^{-1}(T) \cdot \sqrt{Td}) \right) \\
&= O\left(\sqrt{Td} \right),
\end{aligned}$$

where the second equality uses $\Delta = O\left(\log(T) \cdot \sqrt{d/T} \right)$.

Hence, we get regret of order at most

$$\log(T) \cdot \frac{\Delta + \|v\|_1}{v_{\min}^2} + \sqrt{Td}.$$

We have just proved the following lemma.

Lemma 41. *Take $\delta = 1/T^2$. If $\Delta \leq 16\log(T)\sqrt{\frac{d}{T}}$ and H is optimal, then on event $\mathcal{E}_{\text{MD-L}}$, the pulls of MD-Leftist contribute regret of order at most*

$$\log(T) \cdot \frac{\Delta + \|v\|_1}{v_{\min}^2} + \sqrt{Td}.$$

B.4.2 Case 2: $\Delta \leq v_{\min}/16$, arm 0 is optimal, and $\mathcal{E}_{\text{MD-L}}$ happened

The analysis in this case is simpler for two reasons. First, any pull of arm **0** gives no pseudoregret. Second, we do not provide any sort of guarantee that for any epoch $r \leq \rho$ it holds that arm a_r is positive, and we therefore also do not provide any sort of guarantee that $\rho \leq r^*$. Indeed, it can happen that the minimum cost arm on the separating hyperplane H has cost that is much greater than Δ, and in this situation the algorithm is likely to run for many epochs r for which a_r is negative, thereby preventing the algorithm for having informative estimates $\hat{\Delta}_r$ of Δ. Therefore, we only consider the all-encompassing regime that $\rho \leq r_{\max}$ by bounding the regret as if the algorithm ran until epoch $r_{\max}$, which may be overcounting. The analysis is similar to regime 2 above; for completeness, we describe how to modify the analysis and giving the corresponding regret bound.

We first consider the contribution from Phase 1. As we only need consider the regret contribution from arm **1**, we only need the second term of the sum in (28), giving a regret contribution of order at most

$$\log(T) \cdot \frac{\|v\|_1}{v_{\min}^2}.$$

In this regime, we get a regret bound whose order is the same as Regime 2 from Case 1 except that we can and do drop the contribution from arm **0** in the step (32), giving regret of order at most

$$\log(T) \cdot \frac{\|v\|_1}{v_{\min}^2} + \sqrt{Td}. \tag{33}$$

Note that being able to drop the aforementioned term is vital, as here we have no guarantee that $\Delta = O(\log(T)\sqrt{d/T})$.

We have just proved the following lemma.

Lemma 42. *Take $\delta = 1/T^2$. If $\Delta \leq v_{\min}/16$ and arm **0** is optimal, then on event $\mathcal{E}_{\text{MD-L}}$, the pulls of MD-Leftist contribute regret of order at most*

$$\log(T) \cdot \frac{\|v\|_1}{v_{\min}^2} + \sqrt{Td}.$$

Combining the above lemmas yields the following, general regret bound for the pulls of MD-Leftist.

Theorem 43. *Take $\delta = 1/T^2$. If $\Delta \leq v_{\min}/16$, then on event $\mathcal{E}_{\text{MD-L}}$, the pulls of MD-Leftist contribute regret of order at most*

$$\log(T) \cdot \frac{\Delta + \|v\|_1}{v_{\min}^2} + \sqrt{Td}.$$

Proof (of Theorem 43). We begin by bounding the contribution to the regret from MD-Leftist's pulls. We then consider the contribution from the other algorithms.

We consider several cases.

First, if arm **0** is optimal, the bound follows from Lemma 42.

Next, suppose that H is optimal. Then from Lemma 40, we have the bound

$$\log(T) \cdot \left(\frac{\Delta + \|v\|_1}{v_{\min}^2} + \frac{d}{\Delta} \right);$$

when $\Delta \geq 16\log(T)\sqrt{\frac{d}{T}}$, the above bound is of order at most the bound in the theorem. On the other hand, if $\Delta < 16\log(T)\sqrt{\frac{d}{T}}$, then Lemma 41 implies the worst-case bound

$$\log(T) \cdot \frac{\Delta + \|v\|_1}{v_{\min}^2} + \sqrt{Td}.$$

Hence, the result follows. □

B.5 Correctness Analysis for MultiEstTau, MultiCoordinateCompare, and MultiGreedy

Preliminaries. Let H be the separating hyperplane, defined as

$$H := \{a \in \mathbb{R}^d \colon \langle \theta^*, a \rangle = \tau\}.$$

For any arm a, let $\pi(a)$ be the Euclidean projection of a onto the hyperplane; that is,

$$\pi(a) = \operatorname*{arg\,min}_{a' \in H \cap [0,1]^d} \|a' - a\|_2.$$

B.5.1 Correctness analysis for NoisyBinarySearch

For the analysis of NoisyBinarySearch, our analysis from Section A.4 carries over almost entirely without modification. We only need to adapt each of Lemmas 29 and 30 to use event $\mathcal{E}_{\text{MD-L}}$ rather than $\mathcal{E}_{\text{L}}$. To this end, we state the following two adaptations without proof; for the proof of each, one need only replace "Leftist" and $\mathcal{E}_{\text{L}}$ with "MD-Leftist" and $\mathcal{E}_{\text{MD-L}}$ respectively.

Lemma 44 (Adaptation of Lemma 29)**.** *Consider a given iteration i of NBS. Suppose that in this iteration, arm R is positive. Then on event $\mathcal{E}_{\text{MD-L}}$, the probability that NBS incorrectly predicts the label of m on line 7 is at most δ.*

Lemma 45 (Adaptation of Lemma 30)**.** *On event $\mathcal{E}_{\text{MD-L}}$, with probability at least $1 - T \cdot \delta$, NBS will return a positive arm a^f satisfying $\|a^f - \pi(a^f)\|_2 < \varepsilon_{\text{NBS}}$.*

B.5.2 Analysis of MultiCoordinateCompare

We first re-present MultiCoordinateCompare (Algorithm 5 from the main text) in a more space-abundant format, along with a brief English description.

Algorithm 10: MultiCoordinateCompare

Input: Two coordinates j, k. Lower bound ε_r satisfying $\varepsilon_r \leq \Delta$, arm a

Assume $\langle \theta^*, a \rangle \geq \tau$ and $\|a - \pi(a)\| \leq \varepsilon_{\text{NBS}}$
$\gamma \leftarrow \theta_{\min}/(d^2 T)$
$\beta \leftarrow \varepsilon_{\text{NBS}}/\gamma$
$N \leftarrow \frac{\log \frac{1}{\delta}}{\varepsilon_r^2}$
$a^{(j)} \leftarrow a + \beta \cdot \left(\frac{\mathbf{e}_j}{v_j} - \frac{\mathbf{e}_k}{v_k}\right)$
$a^{(k)} \leftarrow a - \beta \cdot \left(\frac{\mathbf{e}_j}{v_j} - \frac{\mathbf{e}_k}{v_k}\right)$
Make N pulls of each of arms $\mathbf{0}$, $a^{(j)}$, and $a^{(k)}$, giving $\hat{p}_0$, $\hat{p}_j$ and $\hat{p}_k$
if $\hat{p}_k - \hat{p}_0 < \varepsilon_r/2$ **then return** "$>$" `// arm` $a^{(k)}$ `is negative`
else if $\hat{p}_j - \hat{p}_0 < \varepsilon_r/2$ **then return** "$<$" `// arm` $a^{(j)}$ `is negative`
else return "$*$" `// arms` $a^{(j)}$ `and` $a^{(k)}$ `both are positive`

In words, MultiCoordinateCompare does the following:

For a positive constant β,[9] create the following two arms:

$$a^{(j)} \leftarrow a + \beta \cdot \left(\frac{\mathbf{e}_j}{v_j} - \frac{\mathbf{e}_k}{v_k} \right)$$
$$a^{(k)} \leftarrow a - \beta \cdot \left(\frac{\mathbf{e}_j}{v_j} - \frac{\mathbf{e}_k}{v_k} \right).$$

Make N pulls of arms $a^{(j)}$, $a^{(k)}$, and $\mathbf{0}$ to identify the label of $a^{(j)}$ and $a^{(k)}$ with high probability.

If $a^{(k)}$ is negative, output "$>$".

If $a^{(j)}$ is negative, output "$<$".

If both $a^{(j)}$ and $a^{(k)}$ are positive, output "$*$", meaning "I don't know".

We will show that MultiCoordinateCompare is an implementation of what we call a *PAC comparison oracle*.

Definition 46 (PAC comparison oracle)**.** Let $\mathcal{I}$ be a finite set and let $f : \mathcal{I} \to \mathbb{R}$. Let $\mathcal{A}$ be an algorithm that takes as input two distinct elements of $\mathcal{I}$ and outputs a symbol from the set {"$>$", "$<$", "$*$"}. We say that $\mathcal{A}$ is a (γ, δ)-PAC comparison oracle for $(\mathcal{I}, f)$ if $\mathcal{A}$ satisfies the following guarantee:

Suppose $\mathcal{A}$ is given two distinct elements $j, k \in \mathcal{I}$ as input; then with probability at least $1 - \delta$:

$$\begin{cases} \text{if } \mathcal{A} \text{ outputs ``}>\text{''} & \text{then } f(j) > f(k); \\ \text{if } \mathcal{A} \text{ outputs ``}<\text{''} & \text{then } f(j) < f(k); \\ \text{if } \mathcal{A} \text{ outputs ``}*\text{''} & \text{then } |f(j) - f(k)| \leq \gamma. \end{cases}$$

In particular, we will show that MultiCoordinateCompare is a PAC comparison oracle for $([d], s)$, where we recall that the leverage score function is defined as $s : j \mapsto \frac{\theta_j^*}{v_j}$. We first require some initial setup.

Assumptions. For the analysis of MultiCoordinateCompare, we currently require some additional assumptions. Some truly are assumptions, while others can be satisfied by appropriate parameter settings for other algorithms on which MultiCoordinateCompare depends.

MultiCoordinateCompare is called by MultiEstTau with an arm a^c as input; this arm comes from a call to NoisyBinarySearch. Within MultiCoordinateCompare, we refer to this arm as a.

Assumptions:

- $\|a - \pi(a)\|_2 \leq \varepsilon_{\max}$
- a is in the κ-interior of $[0, 1]^d$ according to the ℓ_∞-metric, for some $\kappa > 0$.

The first assumption can be satisfied for $\varepsilon_{\max} > 0$ as small as desired by decreasing the termination threshold of NoisyBinarySearch. The second assumption is truly an assumption. We will discuss κ in more detail later.

The following lemma does most of the work for showing that MultiCoordinateCompare is a PAC comparison oracle for $([d], s)$.

In the sequel, we adopt the following notation for brevity:

$$D_{jk} := s(j) - s(k) = \frac{\theta_j^*}{v_j} - \frac{\theta_k^*}{v_k}.$$

Lemma 47. *Let $\varepsilon_a = \|a - \pi(a)\|$. Assume that $\beta \leq \min\{v_j, v_k\} \cdot \kappa$. With probability at least $1 - 2\delta$, MultiCoordinateCompare behaves as follows: it outputs*

$$\begin{cases} \text{``}>\text{''} & \text{if } D_{jk} > \frac{\varepsilon_a}{\beta}; \\ \text{``}<\text{''} & \text{if } D_{jk} < -\frac{\varepsilon_a}{\beta}; \\ \text{``}*\text{''} & \text{if } -\frac{\varepsilon_a}{\beta} \leq D_{jk} \leq \frac{\varepsilon_a}{\beta}. \end{cases}$$

[9] We cannot choose β to be too large as we need both of the above arms to lie in $[0, 1]^d$, but we also need β to be large enough to be meaningful. To satisfy the former restriction, it suffices to require that $\beta \leq \min\{v_j, v_k\} \cdot \kappa$. We will discuss how large β needs to be later.

Proof. The upper bound on β guarantees that both arms are in $[0,1]^d$.

Observe that on the one hand,

$$\begin{aligned}\langle \theta^*, a^{(j)} \rangle &= \langle \theta^*, \pi(a) \rangle + \langle \theta^*, a - \pi(a) \rangle + \beta \cdot \left(\frac{\theta_j}{v_j} - \frac{\theta_k}{v_k} \right) \\ &= \tau + \varepsilon_a + \beta \cdot D_{jk},\end{aligned}$$

while on the other hand,

$$\langle \theta^*, a^{(k)} \rangle = \tau + \varepsilon_a - \beta \cdot D_{jk}.$$

We first establish whether each of arms $a^{(j)}$ and $a^{(k)}$ has a true label that is positive or negative. We then show that the algorithm correctly predicts their labels. Suppose that $D_{jk} > \frac{\varepsilon_a}{\beta}$. Then from the formula for $\langle \theta^*, a^{(k)} \rangle$ above, we see that $a^{(k)}$ is negative. Similarly, suppose that $D_{jk} < -\frac{\varepsilon_a}{\beta}$. Then from the formula for $\langle \theta^*, a^{(j)} \rangle$ above, we see that $a^{(j)}$ is negative. Finally, suppose that $-\frac{\varepsilon_a}{\beta} \leq D_{jk} \leq \frac{\varepsilon_a}{\beta}$. Then it is easy to verify from the above formulas that both $a^{(j)}$ and $a^{(k)}$ are positive.

Next, we consider the algorithm's predicted labels for arms $a^{(j)}$ and $a^{(k)}$. From Lemma 44, line 8 of MultiCoordinateCompare correctly predicts the label of $a^{(k)}$ with probability at least $1 - \delta$. Similarly, the same lemma implies that line 9 of MultiCoordinateCompare correctly predicts the label of $a^{(j)}$ with probability at least $1 - \delta$.

Consequently, with probability at least $1 - 2\delta$, the following statements hold simultaneously:

- if $D_{jk} > \frac{\varepsilon_a}{\beta}$, the algorithm correctly predicts that $a^{(k)}$ is negative and hence indeed outputs "$>$";
- if $D_{jk} < -\frac{\varepsilon_a}{\beta}$, the algorithm correctly predicts that $a^{(j)}$ is negative and hence indeed outputs "$<$";
- if $-\frac{\varepsilon_a}{\beta} \leq D_{jk} \leq \frac{\varepsilon_a}{\beta}$, the algorithm correctly predicts that both $a^{(j)}$ and $a^{(k)}$ are positive and hence outputs "$*$".

□

Next, we present a simple corollary to Lemma 47. This corollary shows that an appropriate parameterization of MultiCoordinateCompare is a (γ, δ)-PAC comparison oracle provided that $\varepsilon_a = \|a - \pi(a)\|$ can be guaranteed to be small enough.

Corollary 48. *Take the setting of Lemma 47, assume that $\varepsilon_a \leq \varepsilon_{\max}$ for some positive constant $\varepsilon_{\max} \leq \gamma \cdot \min\{v_j, v_k\} \cdot \kappa$, and set β as $\beta = \frac{\varepsilon_{\max}}{\gamma}$ for a positive constant γ.*

Then, with probability at least $1 - 2\delta$:

MultiCoordinateCompare is trustworthy *in the sense that:*

$$\begin{cases} \text{if it outputs ``}>\text{''} & \text{then } D_{jk} > 0; \\ \text{if it outputs ``}<\text{''} & \text{then } D_{jk} < 0; \\ \text{if it outputs ``}*\text{''} & \text{then } -\gamma \leq D_{jk} \leq \gamma. \end{cases}$$

MultiCoordinateCompare is γ-accurate *in the sense that:*

$$\begin{cases} \text{if } D_{jk} > \gamma & \text{then it outputs ``}>\text{''} \\ \text{if } D_{jk} < -\gamma & \text{then it outputs ``}<\text{''}. \end{cases}$$

In particular, MultiCoordinateCompare is a $(\gamma, 2\delta)$-PAC comparison oracle.

Proof. The upper bound on $\varepsilon_{\max}$ ensures that $\beta \leq \min\{v_j, v_k\} \cdot \kappa$, thereby guaranteeing that both arms are in $[0,1]^d$.

We first show that MultiCoordinateCompare is trustworthy. The first two claims are trivial, as from Lemma 47, with probability at least $1 - 2\delta$, outputting "$>$" implies that $D_{jk} > \frac{\varepsilon_a}{\beta} > 0$ while outputting "$<$" implies that $D_{jk} < -\frac{\varepsilon_a}{\beta} < 0$.

For the last claim, observe from Lemma 47 that (with probability at least $1-2\delta$) outputting "$*$" implies that $-\frac{\varepsilon_a}{\beta} \leq D_{jk} \leq \frac{\varepsilon_a}{\beta}$, after which we use $\gamma = \frac{\varepsilon_{\max}}{\beta} \geq \frac{\varepsilon_a}{\beta}$ and likewise $-\gamma \leq \frac{\varepsilon_a}{\beta}$.

The claim that MultiCoordinateCompare is γ-accurate straightforward to verify using Lemma 47 and the fact that $\gamma \geq \frac{\varepsilon_a}{\beta}$.

The fact that MultiCoordinateCompare is a (γ, δ)-PAC comparison oracle follows immediately from the algorithm being trustworthy with probability at least $1 - \delta$. □

B.5.3 PAC-MergeSort

In this section, we present PAC-MergeSort, a method for probably, approximately correctly sorting elements given access to a PAC comparison oracle. We will use PAC-MergeSort to obtain an ordering of the coordinates that is approximately optimal in a sense that we now define.

For a permutation σ of $(1, 2, \ldots, d)$, let $\sigma(i)$ denote the i^{th} element in the permutation.[10]

Definition 49 (γ-insensitivity (general version))**.** We say that a permutation σ of $\mathcal{I}$ is γ-*insensitive* with respect to $f : \mathcal{I} \to \mathbb{R}$ if, for all j and k satisfying $1 \leq j < k \leq |\mathcal{I}|$, we have

$$f(\sigma(j)) \geq f(\sigma(k)) - \gamma.$$

Definition 50 (γ-insensitivity)**.** We say that a permutation σ is γ-*insensitive* if, for all j and k satisfying $1 \leq j < k \leq d$, we have

$$\frac{\theta^*_{\sigma(j)}}{v_{\sigma(j)}} \geq \frac{\theta^*_{\sigma(k)}}{v_{\sigma(k)}} - \gamma.$$

We remark that a 0-insensitive permutation corresponds to correctly placing the elements in non-increasing order. In our setting, we wish to obtain a γ-insensitive permutation over $[d]$ with respect to $s : j \mapsto \frac{\theta^*_j}{v_j}$. As we show in Section B.5.4, such a permutation is approximately optimal and enables the greedy algorithm (when run with this permutation) to output an arm on the hyperplane that approximately minimizes the cost over all arms on the hyperplane.

We now present PAC-MergeSort, an extension of MergeSort that handles approximate comparisons. For γ-approximately correct comparisons, PAC-MergeSort also approximately places the elements in decreasing order.

Algorithm Sketch 51 (PAC-MergeSort)**.** First, recall that MergeSort only makes comparisons in the merge step. In the merge step, we have two lists, each in order, and traversing each list from left to right we perform comparisons to get a single, merged list that is in order. In PAC-MergeSort, we run MergeSort as usual with the following modification: whenever two elements are compared and we receive response "$*$", we join the elements into a single element by arbitrarily selecting one of them as the representative. Since the elements are joined, we just pick this single element for the next item in the merged list currently being constructed. The result is a list of representatives that is strictly decreasing. To get an ordering of all the elements, we traverse the list of representatives from left to right, outputting each representative followed by the elements it represents before going to the next representative.

Pictorially, PAC-MergeSort can be thought of as initially placing each element in its own bucket. When we receive a "$*$" from comparing two buckets (by comparing their representatives), we union the two buckets into a single bucket whose representative is arbitrarily selected to be one of the representatives of the two buckets. In the end, the algorithm returns an ordering that begins with the elements of the first bucket (in arbitrary order), then the elements of the second bucket (in arbitrary order), and so on.

How bad an ordering σ' could PAC-MergeSort produce? As we now show, given comparison tolerance γ, PAC-MergeSort delivers γ'-sensitivity for γ' at most a $(|\mathcal{I}| - 1)$-amplification of γ.

Theorem 52. *Given a (γ, δ)-PAC comparison oracle for $(\mathcal{I}, f)$, PAC-MergeSort returns an ordering that, with probability at least $1 - (2|\mathcal{I}| \log_2 |\mathcal{I}|) \cdot \delta$, is $(|\mathcal{I}| - 1)\gamma$-insensitive.*

Proof. Define $n := |\mathcal{I}|$. First, we use the fact that standard MergeSort makes at most (overcounting a bit) $2n \log_2 n$ comparisons. Hence, taking a union bound over all the comparisons, we have with probability at least $1 - (2n \log_2 n) \cdot \delta$ that the PAC comparison oracle's guarantees hold for every comparison made.

[10] We recognize that this is non-standard use of permutation notation, but our convention is more convenient here.

With the "high probability" part out of the way, we proceed with the rest of the proof. Let distinct positions $i_j, i_k \in [n]$ be arbitrary except that $i_j > i_k$. We adopt the notation $j := \sigma(i_j)$ and $k := \sigma(i_k)$; note that i_j and i_k will be used to indicate positions of permutation σ', while j and k are the corresponding elements in those positions.

How bad an ordering σ' could PAC-MergeSort produce? If $f(j) \leq f(k)$, then the coordinates were sorted correctly. The out of order case is when $f(j) > f(k)$, but fortunately, we must have $f(j) - f(k) \leq (n-1)\gamma$. To see this, we consider two cases.

In the first case, elements j and k are out of order and belong to the same bucket $B \subseteq \mathcal{I}$ at the end of the execution of PAC-MergeSort. Therefore, the diameter of a bucket $\mathrm{diam}(B) := \max_{m,m' \in [B]} f(m) - f(m')$ is an upper bound on $f(j) - f(k)$. But in PAC-MergeSort, each bucket initially has diameter zero, and the diameter of a bucket increases only when it is unioned with another bucket; specifically, if bucket B_1 is unioned with bucket B_2, then the diameter of the resulting bucket is at most $\mathrm{diam}(B_1) + \mathrm{diam}(B_2) + \gamma$. The diameter of B is maximized when B was formed by successively unioning with $|B| - 1$ singleton buckets, giving diameter at most $(|B| - 1)\gamma \leq (n-1)\gamma$.

In the second case, elements j and k are out of order but belong to different buckets at the end of the execution of PAC-MergeSort. Before analyzing this case, we need some preliminary setup:

- We define the *range* of a bucket A to be $[\min A, \max A]$.
- For a bucket A, let $q(A)$ be its representative.

Let A and B be distinct buckets with $q(A) > q(B) + \gamma$. Suppose A and B overlap (i.e., $\mathrm{range}(A) \cap \mathrm{range}(B) \neq \emptyset$). Then we must have $q(A) - q(B) \leq \mathrm{diam}(A) + \mathrm{diam}(B)$.

Suppose two elements $k \in A$ and $j \in B$ are out of order in σ' (so $f(j) > f(k)$). Then we must have $f(k) \in \mathrm{range}(B)$ and $f(j) \in \mathrm{range}(A)$. Now, we claim that $f(j) - f(k) \leq \min\{\mathrm{diam}(A), \mathrm{diam}(B)\}$. Indeed, if instead $f(j) - f(k) > \mathrm{diam}(A)$, then $f(j)$ cannot belong to $\mathrm{range}(A)$. Similarly, if $f(j) - f(k) > \mathrm{diam}(B)$, then $f(k)$ cannot belong to $\mathrm{range}(B)$.

Hence, the final ordering satisfies $(n-1) \cdot \gamma$-sensitivity. □

The following corollary is immediate.

Corollary 53. *Given a (γ, δ)-PAC comparison oracle for $([d], s)$, PAC-MergeSort returns an ordering that, with probability at least $1 - (2d \log_2 d) \cdot \delta$, is $(d-1)\gamma$-insensitive.*

B.5.4 Analysis of MultiGreedy

We first re-present MultiGreedy for the convenience of the reader.

Algorithm 11: MultiGreedy

Input: Lower bound ε_r satisfying $\varepsilon_r \leq \Delta$, Ordering $\hat{\sigma}$

$u \leftarrow 2d \cdot 2^{-r} + 1/T$

$N \leftarrow \frac{\log \frac{1}{\delta}}{\varepsilon_r^2}$

$L \leftarrow \mathbf{0}$

for $i = 1, 2, \ldots, d$ **do**

  $R \leftarrow L + \min\left\{\frac{1}{v_{\hat{\sigma}(i)}}\left(u - \sum_{j=1}^{i-1} v_{\hat{\sigma}(j)}\right), 1\right\} \cdot \mathbf{e}_{\hat{\sigma}(i)}$

  Make N pulls of each of arms L and R

  if $\hat{p}_{\text{right}} - \hat{p}_{\text{left}} \geq \varepsilon_r/2$ **then**

    return NoisyBinarySearch(ε_r, L, R)

  else $L \leftarrow L + \mathbf{e}_{\hat{\sigma}(i)}$

In this section, we show that MultiGreedy returns an arm in the positive halfspace that is approximately of minimum cost among all arms in the positive halfspace.

Prior to analyzing MultiGreedy, we set up some notation and introduce two events. Let a^H be the minimum cost arm on the separating hyperplane. In addition, let a_σ be the minimum cost arm on the hyperplane that abides[11] by the ordering σ, and let $\hat{a}_\sigma$ be the arm returned by MultiGreedy when given ordering σ. Finally, define $v_{\max} := \max_{j \in [d]} v_j$.

The idea of MultiGreedy relies on the following permutation-based characterization of a minimum cost arm in H.

Proposition 54. *Let $\sigma(j)$ be a permutation of $[d]$ such that*

$$\frac{\theta^*_{\sigma(j)}}{v_{\sigma(j)}} \geq \frac{\theta^*_{\sigma(k)}}{v_{\sigma(k)}} \tag{34}$$

if $j < k$. Then, there exists an optimal arm in the hyperplane $a^H \in \arg\min_{a \in H} \langle v, a \rangle$ such that, for some $l \in [d]$,

$$(a^H_{\sigma(1)}, a^H_{\sigma(2)}, \ldots, a^H_{\sigma(d)}) = (1, \ldots, 1, a^H_{\sigma(l)}, 0, \ldots, 0).$$

Proof of Proposition 54. For the ease of discussion, we assume the coordinates are re-indexed as

$$\frac{\theta^*_1}{v_1} \geq \frac{\theta^*_2}{v_2} \geq \ldots \geq \frac{\theta^*_d}{v_d}. \tag{35}$$

We show that

$$a^{\text{grdy}} := (1, 1, \ldots, a_l, 0, \ldots, 0) : \sum_{m<l} \theta^*_m + a_l \theta^*_l = \tau$$

is an optimal solution of the optimization problem

$$\begin{aligned} \arg\min_{a \in [0,1]^d} & \sum_{m=1}^{d} v_m a_m \\ \text{s.t. } & \theta^*_m a_m = \tau. \end{aligned} \tag{36}$$

Suppose that there exists an optimal solution a' of problem (36) such that $a'_i > 0$ for some $i > l$. Then, by definition there exists $j \leq l$ such that $a^{\text{grdy}}_j > a'_j$. In this case, we can transform a' as follows: Namely, we reduce a'_i by $\Delta' = \min\{a'_i, (a^{\text{grdy}}_j - a'_j)(\theta^*_j/\theta^*_i)\}$ and increase a'_j by $\Delta'(\theta^*_i/\theta^*_j)$. This operation (a) does not modify $\sum_{m=1}^d \theta^*_m a'_m$ (i.e., the resultant arm stays on the separating hyperplane) and (b) does not increase $\sum_{m=1}^d v_m a'_m$ by (35). Moreover, this operation (c) increases $\sum_{m=1}^d \mathbf{1}\left[a^{\text{grdy}}_m = a'_m\right]$ by at least one. Repeating this procedure at most d times transforms a' into a^{grdy} (by (c)) without increasing the objective value (by (a) and (b)). □

[11] By this, we mean that greedy fills the coordinates in the order given by σ.

Let $\mathcal{E}_{\text{MG}}$ be the event that MultiGreedy, over all of the at most d iterations of its for **loop**, correctly predicts the label of arm R on line 7. Lemma 69 in Appendix C shows that this event holds with suitably high probability. In addition, we define two events related to NoisyBinarySearch as they will be useful in this section's analysis. Let event $\mathcal{E}_{\text{NBS-MET}}$ (event $\mathcal{E}_{\text{NBS-MG}}$) be the event that NoisyBinarySearch, when called from MultiEstTau (when called from MultiGreedy), returns a positive arm within distance ε_{NBS} of $H \cap [0,1]^d$.

Lemma 55. *Given a γ-insensitive permutation σ, on event $\mathcal{E}_{\text{stop}} \cap \mathcal{E}_{\text{MG}} \cap \mathcal{E}_{\text{NBS-MG}}$, MultiGreedy returns an arm $\hat{a}_\sigma$ in the positive halfspace that satisfies*

$$\langle v, \hat{a}_\sigma - a^H \rangle \leq v_{\max} \cdot \varepsilon_{\text{NBS}} + \varepsilon_\gamma,$$

where $\varepsilon_\gamma = \binom{d}{2} \cdot \gamma \cdot \frac{v_{\max}^2}{\theta_{\min}}$.

The proof of this lemma follows easily from Lemmas 56 and 59 below.

Lemma 56. *For any γ-insensitive permutation σ, we have*

$$\langle v, a_\sigma \rangle \leq \langle v, a^H \rangle + \varepsilon_\gamma$$

where $\varepsilon_\gamma = \binom{d}{2} \cdot \gamma \cdot \frac{v_{\max}^2}{\theta_{\min}}$.

The next lemma relies on the following regularity assumption on θ^*.

Assumption 57 (Regularity of θ^*)**.** There exists a known, positive constant $\theta_{\min}$ such that, for any coordinate j for which θ^* is non-zero, we have $\theta_j^* \geq \theta_{\min}$.

Lemma 58. *Let σ be a permutation of $(1, 2, \ldots, d)$, and let σ' be equal to σ except that an adjacent pair of indices $(j, k) := (\sigma(i), \sigma(i+1))$ is swapped, so that $\sigma'(i) = \sigma(i+1)$ and $\sigma'(i+1) = \sigma(i)$.*

If coordinates j and k satisfy $D_{jk} > 0$ and are γ-close, i.e.,

$$\frac{\theta_k^*}{v_k} < \frac{\theta_j^*}{v_j} \leq \frac{\theta_k^*}{v_k} + \gamma,$$

then the cost of $a_{\sigma'}$ can be upper bounded as

$$\langle v, a_{\sigma'} \rangle \leq \langle v, a_\sigma \rangle + \gamma \cdot \frac{v_{\max}^2}{\theta_{\min}}.$$

Proof. For convenience, we first state some facts related to γ-insensitivity. The condition

$$\frac{\theta_j^*}{v_j} \leq \frac{\theta_k^*}{v_k} + \gamma \tag{37}$$

is equivalent to each of the following

$$\frac{\theta_j^*}{\theta_k^*} \cdot v_k - v_j \leq \gamma \cdot \frac{v_j v_k}{\theta_k^*}; \tag{38}$$

and

$$v_k - \frac{\theta_k^*}{\theta_j^*} \cdot v_j \leq \gamma \cdot \frac{v_j v_k}{\theta_j^*}. \tag{39}$$

From the premise of the lemma, we assume that coordinates j and k satisfy $D_{jk} > 0$ and are γ-close in the sense of (37). We consider the impact of running the greedy algorithm with the ordering σ' instead of the ordering σ. Let $a := a_\sigma$ be the arm greedy returns when using ordering σ, and let $a' := a_{\sigma'}$ be the arm greedy returns when using ordering σ'.

We consider a few cases to establish that the cost of arm a' is not much larger than the cost of arm a.

Case 0: Both coordinates j and k are unused under σ ($a_j = a_k = 0$); then swapping the coordinates clearly has no effect.

Case 1: Both coordinates are saturated under σ ($a_j = a_k = 1$)); then swapping the coordinates clearly has no effect.

Case 2: The arm a returned under σ satisfies $0 < a_j \leq 1$ and $0 \leq a_k < 1$.

We split this case into two sub-cases.

Case 2A: $a'_k = 1$. We need

$$a_j\theta^*_j + a_k\theta^*_k = a'_k\theta^*_k + a'_j\theta^*_j,$$

and so under our current assumptions

$$\begin{aligned} a'_j &= \frac{a_j\theta^*_j + a_k\theta^*_k - \theta^*_k}{\theta^*_j} \\ &= a_j - (1 - a_k) \cdot \frac{\theta^*_k}{\theta^*_j}. \end{aligned}$$

The difference in cost is

$$\begin{aligned} & a'_k v_k + a'_j v_j - a_j v_j - a_k v_k \\ &= v_k + \left(a_j - (1 - a_k) \cdot \frac{\theta^*_k}{\theta^*_j} \right) \cdot v_j - a_j v_j - a_k v_k \\ &= (1 - a_k) \cdot v_k - (1 - a_k) \cdot \frac{\theta^*_k}{\theta^*_j} \cdot v_j \\ &= (1 - a_k) \cdot \left(v_k - \frac{\theta^*_k}{\theta^*_j} \cdot v_j \right) \\ &\leq (1 - a_k) \cdot \gamma \cdot \frac{v_j v_k}{\theta^*_j} \\ &\leq (1 - a_k) \cdot \gamma \cdot \frac{v_j v_k}{\theta_{\min}}, \end{aligned}$$

where the first inequality is from (39) and the last inequality uses the fact that $D_{jk} > 0$ implies that $\theta^*_j > 0$, allowing us to invoke Assumption 57.

Case 2B: $a'_k < 1$ (so $a'_j = 0$). We need

$$a_j\theta^*_j + a_k\theta^*_k = a'_k\theta^*_k + a'_j\theta^*_j,$$

and so under our current assumptions

$$a'_k = \frac{a_j\theta^*_j + a_k\theta^*_k}{\theta^*_k} = a_k + a_j \cdot \frac{\theta^*_j}{\theta^*_k}.$$

The difference in cost is then

$$\begin{aligned} & a'_k v_k - a_j v_j - a_k v_k \\ &= \left(a_k + a_j \cdot \frac{\theta^*_j}{\theta^*_k} \right) \cdot v_k - a_j v_j - a_k v_k \\ &= a_j \cdot \left(\frac{\theta^*_j}{\theta^*_k} \cdot v_k - v_j \right) \\ &\leq a_j \cdot \gamma \cdot \frac{v_j v_k}{\theta^*_k} \\ &\leq a_j \cdot \gamma \cdot \frac{v_j v_k}{\theta^*_{\min}}, \end{aligned}$$

where the first inequality is from (38) and the last inequality uses the fact that $a'_k < 1$ implies that $\theta^*_k > 0$ and hence $\theta^*_k \geq \theta_{\min}$ from Assumption 57. □

Lemma 56 is a simple consequence of Lemma 58, as we now show.

Proof (of Lemma 56). Recall that σ is a γ-insensitive permutation. We will show how to convert σ into an optimal permutation σ^*, i.e., a permutation for which $D_{\sigma(i)\sigma(i+1)} \geq 0$ for all $i \in [d]$, without decreasing the cost of the corresponding greedy arm by much.

Let us first recall the notion of an *inversion*. We say that the pair of positions (i, i') is an inversion in σ if $i > i'$ but $D_{\sigma(i)\sigma(i')} > 0$; this may seem counterintuitive, so we remind the reader that we wish to sort in decreasing order, meaning that if $D_{\sigma(i)\sigma(i')} > 0$, then we wish to have $i < i'$.

If there is an inversion in σ, then there is an adjacent pair $(i+1, i)$ that is an inversion in σ. We can swap such a pair using Lemma 58, giving a new permutation which has one less inversion and whose cost has been reduced by at most $\gamma \cdot \frac{v_{\max}^2}{\theta_{\min}}$. By successively applying this exchange argument, each time removing one inversion, we finally arrive at an optimal permutation σ^*. As there can be at most $\binom{d}{2}$ inversions, the cost of $\langle v, a_\sigma \rangle$ could have larger than the cost of a_{σ^*} by at most $\binom{d}{2} \cdot \gamma \cdot \frac{v_{\max}^2}{\theta^{\min}}$, as desired. □

Lemma 59. *Given a γ-insensitive permutation σ, on event $\mathcal{E}_{\text{stop}} \cap \mathcal{E}_{\text{MG}} \cap \mathcal{E}_{\text{NBS-MG}}$, MultiGreedy returns an arm $\hat{a}_\sigma$ in the positive halfspace satisfying*

$$\langle v, \hat{a}_\sigma \rangle \leq \langle v, a_\sigma \rangle + \varepsilon_{\text{NBS}} \cdot v_{\min}.$$

Proof. We begin by studying a_σ. Recall that $a_\sigma \in H$.

For an arm a and a permutation σ', we adopt the notation $[a]_{\sigma'[d]} = (a_{\sigma'(1)}, a_{\sigma'(2)}, \ldots, a_{\sigma'(d)})$. Let $[a_\sigma]_{\sigma([d])}$ be of the form $(1, \ldots, 1, \underbrace{b}_{\text{position } j}, 0, \ldots, 0)$. It is without loss of generality that we take $b > 0$, as $\mathbf{0}$ is negative and a_σ is positive.

Let $a_\sigma^{(-)} = a_\sigma - b \cdot e_{\sigma(j)}$, and let $a_\sigma^{(+)} = a_\sigma$ except at position j, where $[a_\sigma^{(+)}]_{\sigma(j)} = u'$ for

$$u' := \min \left\{ \frac{1}{v_j} \left(2d \cdot 2^{-r} + \varepsilon_\gamma - \sum_{i=1}^{j-1} v_i \right), 1 \right\};$$

here, $r = \rho$ is the epoch in which Multi-dim Leftist called MultiEstTau. Therefore, we have:

$$\begin{aligned} [a_\sigma^{(-)}]_{\sigma([d])} &= (1, \ldots, 1, 0, 0, \ldots, 0) \\ [a_\sigma]_{\sigma([d])} &= (1, \ldots, 1, b, 0, \ldots, 0) \\ [a_\sigma^{(+)}]_{\sigma([d])} &= (1, \ldots, 1, u', 0, \ldots, 0) \end{aligned}$$

Since a_σ is on the hyperplane, it is positive. Also, $a_\sigma^{(-)}$ is negative. To see this, suppose for a contradiction that $a_\sigma^{(-)}$ is positive. Then since $v_j > 0$ by assumption, $a_\sigma^{(-)}$ is a lower cost positive point than a_σ.

In addition, as we now show, we must have $u' \geq b$, implying that $a_\sigma^{(+)}$ also is positive. It trivially holds that $u' \geq b$ when $u' = 1$, so we consider the case that $u' < 1$; to see why $u' \geq b$ in this case, we first note that it is equivalent to show that $\langle v, a_\sigma \rangle \leq \langle v, a_\sigma^{(+)} \rangle$. To see why the latter inequality is true, observe that:

$$\begin{aligned} \langle v, a_\sigma \rangle &\leq \langle v, a^H \rangle + \varepsilon_\gamma \\ &\leq \langle v, a^c \rangle + \varepsilon_\gamma \\ &\leq \langle v, R(A_{r-\ell}) \rangle + \varepsilon_\gamma \\ &= d \cdot 2^{\ell - r} \cdot v_{\min} + \varepsilon_\gamma \\ &= d \cdot 2^{-r} \cdot v_{\min} \cdot \left\lceil \log_2 \frac{1}{v_{\min}} \right\rceil + \varepsilon_\gamma \\ &\leq 2d \cdot 2^{-r} + \varepsilon_\gamma \\ &= \langle v, a_\sigma^{(+)} \rangle, \end{aligned}$$

where the first inequality is from Lemma 56; the second inequality is from the optimality of a^H among arms in the positive halfspace; the third inequality is because by virtue of NoisyBinarySearch and the fact (from $\mathcal{E}_{\text{stop}}$) that $r = \rho \geq \ell + 1$, it

follows that a_c is entrywise upper bounded by $R(A_{r-\ell})$; and the final equality is from direct verification since u is equal to the first term of the minimum in its definition.

Now, for any iteration $i < j$, both arms considered by MultiGreedy are negative. We first show that the algorithm makes it to iteration j. Indeed, in iteration 1, arm $L = \mathbf{0}$ is known to be negative, and so the algorithm need only predict the label of arm R (which it does, on line 7 of MultiGreedy). Since we assume event $\mathcal{E}_{\text{MG}}$ happened, the algorithm correctly predicts the label. In each successive iteration up to and including iteration $j-1$, from event $\mathcal{E}_{\text{MG}}$ the algorithm again correctly predicts the label of arm R to be negative. Therefore, the algorithm makes it to iteration j. In iteration j, again from event $\mathcal{E}_{\text{MG}}$, the algorithm will detect a difference between arms $a_\sigma^{(-)}$ and $a_\sigma^{(+)}$, triggering NoisyBinarySearch between these two arms.

The result of NoisyBinarySearch, which is correct as we assume event $\mathcal{E}_{\text{NBS-MG}}$, will be a positive arm $\hat{a}_\sigma$ (this is by virtue of NoisyBinarySearch returning a "rightmost" arm along the line segment it searches). Moreover, along the line segment searched, the returned arm will be within some distance ε_{NBS} of the hyperplane when measured along coordinate direction e_j. Therefore, the additional cost of $\hat{a}_\sigma$ compared to a_σ is at most $v_j \cdot \varepsilon_{\text{NBS}}$. □

For completeness, we give a brief proof of Lemma 55.

Proof (of Lemma 55). That $\hat{a}_\sigma$ is in the positive halfspace is immediate from Lemma 59. Also, from Lemmas 56 and 59,

$$\begin{aligned}\langle v, \hat{a}_\sigma - a^H \rangle &= \langle v, \hat{a}_\sigma - a_\sigma \rangle + \langle v, a_\sigma - a^H \rangle \\ &\leq v_{\max} \cdot \varepsilon_{\text{NBS}} + \varepsilon_\gamma.\end{aligned}$$

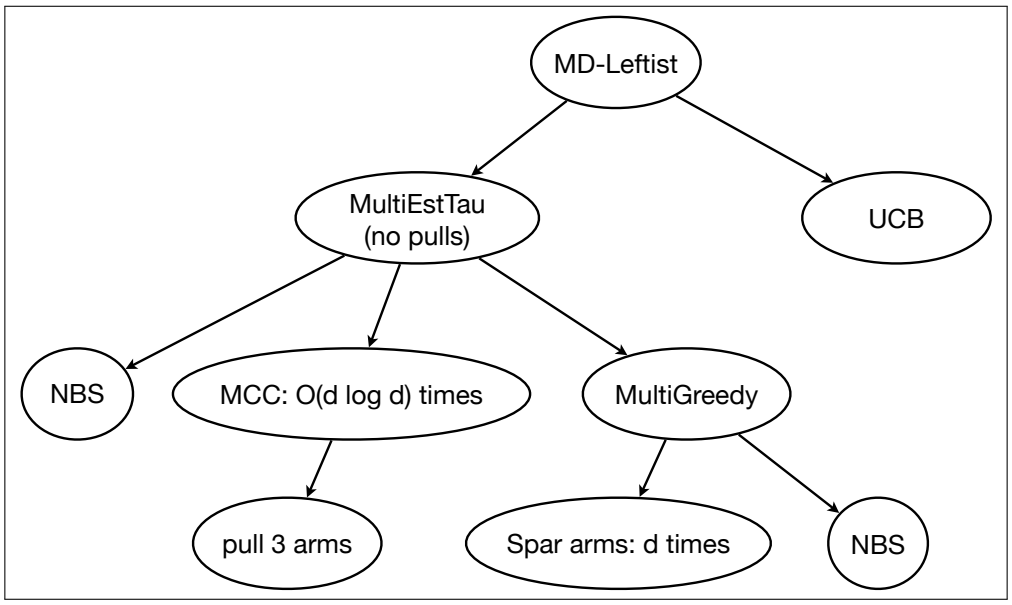
□

B.6 Complete regret analysis for multi-dimensional case

We now give a detailed analysis that considers each of the pieces that contributes to the regret.

B.6.1 Overview of total regret analysis

We use the abbreviations MD-Leftist (MultiDimLeftist), MCC (MultiCoordinateCompare), and NBS (NoisyBinarySearch). There are two situations. Either MD-Leftist calls MultiEstTau, or it does not. The former case is much more complicated; we consider it first. The diagram below shows the algorithms that are called. To interpret the diagram, we start at the root note (MD-Leftist) and do an in-order traversal. With the exception of the node for MD-Leftist, pulls (and hence regret contribution) happen only in leaf nodes.



We begin with a description of the places where regret can be accumulated (in addition to regret from the pulls of MD-Leftist):

1. MultiEstTau calls NBS once after MD-Leftist detects a lower bound on Δ. The result is a^c.
2. MultiEstTau then calls MultiCoordinateCompare $O(d \log d)$ times. For each call, we do a batch pull of arms $\mathbf{0}$, $a^{(j)}$, and $a^{(k)}$. The latter two arms are guaranteed to be very close to a^c, which is why we should be able to use the same regret contribution from a^c when we consider pulls of $a^{(j)}$ and $a^{(k)}$. The regret contribution of arm $\mathbf{0}$ should be even lower.
3. MultiGreedy has at most d iterations in which it spars two extreme points (arms) of $[0,1]^d$. By design, the cost of the arms considered should be no more than the cost of a^c.
4. MultiGreedy calls NBS once after identifying two extreme points of $[0,1]^d$ to interpolate along one dimension.
5. MD-Leftist calls UCB.

We begin by bounding the regret of the first four pieces in the order of piece 1, piece 4, piece 2, and piece 3. We only sketch the analysis as the complete details are similar to the regret analysis already done for MD-Leftist in Section B.4 and NBS in Section A.4. We then bound the regret of UCB.

Recall that ρ is the stopping epoch for MD-Leftist; this is the epoch in which MD-Leftist calls MultiEstTau (if at all such a call happens).

B.6.2 Regret analysis for first four pieces

Lemma 60 (Regret of NBS from MultiEstTau (Piece 1))**.** *On the event $\mathcal{E}_{\text{stop}} \cap \mathcal{E}_{\text{NBS-MET}}$, the pulls from MultiEstTau's call to NBS contribute regret of order at most*

$$\log(T) \cdot \log(1/\varepsilon_{\text{NBS}}) \cdot \frac{d}{\Delta}.$$

Proof. The number of pulls in each round of NBS will be of order $O(\log(T) \cdot \frac{1}{\Delta^2})$ since $n_\rho = O(\log(1/\delta) \cdot 2^{2\rho})$, we have $\rho \asymp r_\Delta \asymp \log_2 \frac{1}{\Delta}$, and we take $\delta = 1/T^2$. The number of rounds of NBS is of order $O(\log(1/\varepsilon_{\text{NBS}}))$. Hence, each call to NBS results in $O(\log(T) \cdot \log(1/\varepsilon_{\text{NBS}})/\Delta^2)$ pulls.

Upon being called, NBS's initial setting for arms L and R are arm $\mathbf{0}$ and a_ρ respectively. The instantaneous regret per pull is at most Δ for arm $\mathbf{0}$ and at most $d \cdot \Delta$ for arm a_ρ (see (29)), and the instantaneous regret of intermediate arms can be bounded by summing these two terms. Hence, the total regret in this case is of order at most

$$\log(T) \cdot \log(1/\varepsilon_{\text{NBS}}) \cdot \frac{d}{\Delta}.$$

□

Lemma 61 (Regret of NBS from MultiGreedy (Piece 4))**.** *On the event $\mathcal{E}_{\text{stop}} \cap \mathcal{E}_{\text{NBS-MET}} \cap \mathcal{E}_{\text{MG}} \cap \mathcal{E}_{\text{NBS-MG}}$, the pulls from MultiGreedy's call to NBS contribute regret of order at most*

$$\log(T) \cdot \log(1/\varepsilon_{\text{NBS}}) \cdot \frac{d}{\Delta}.$$

Proof. By the design of MultiGreedy, the cost of the "right-most" arm R in any iteration is at most $2d \cdot 2^{-\rho} + 1/T$. The regret can now be bounded by the same amount as in the case of NBS from MultiEstTau as the $1/T$ term contributes a negligible amount to the regret when using big-O notation. □

Lemma 62 (Regret of MultiCoordinateCompare (Piece 2))**.** *On the event $\mathcal{E}_{\text{stop}} \cap \mathcal{E}_{\text{NBS-MET}}$, the contribution to the regret from all pulls from all calls to MultiCoordinateCompare is of order at most*

$$\log(T) \cdot \frac{d^2 \log d}{\Delta}.$$

Proof. For each of the $\Theta(d \log d)$ calls to MultiCoordinateCompare, we pull 3 arms: arm $a^{(j)}$, arm $a^{(k)}$, and arm $\mathbf{0}$. The former two arms are guaranteed to be within distance $\beta = \frac{1}{T}$ of a^c. Hence, the order of the contribution to the regret is the same as the order of the regret from MD-Leftist's pulls while it is in epoch ρ; from (30), this latter amount is of order at most $(\log T) \cdot \frac{d}{\Delta}$. Considering the amplification by $d \log d$, the result follows. □

Lemma 63 (Regret of MultiGreedy's sparring (Piece 3)). *On the event $\mathcal{E}_{\text{stop}} \cap \mathcal{E}_{\text{NBS-MET}}$, the contribution to the regret from all pulls from MultiGreedy is of order at most*

$$\log(T) \cdot \frac{d^2}{\Delta}.$$

Proof. There can be at most d sparring rounds. By the design of MultiGreedy, the cost of the larger arm in each sparring is at most $2d \cdot 2^{-\rho} + \varepsilon_\gamma$, where we recall that $\varepsilon_\gamma \leq \frac{1}{T}$ and hence contributes a negligible amount to the regret when using big-O notation. Since $n_\rho = O(2^{2\rho})$ and $\rho \asymp r_\Delta \asymp \log_2 \frac{1}{\Delta}$, the result follows. □

B.6.3 Regret of UCB on two-arm problem

Before bounding the regret of UCB in our setting, we introduce one more event. Next, let $\mathcal{E}_{\text{sort}}$ be the event that PAC-MergeSort produces a γ-insensitive ordering for $\gamma = \frac{\theta_{\min}}{d^2 T}$. Lemma 68 in Appendix C shows that this event holds with suitably high probability.

Denote by $\mathcal{E}$ the event $\mathcal{E} := \mathcal{E}_{\text{MD-L}} \cap \mathcal{E}_{\text{stop}} \cap \mathcal{E}_{\text{NBS-MET}} \cap \mathcal{E}_{\text{sort}} \cap \mathcal{E}_{\text{MG}} \cap \mathcal{E}_{\text{NBS-MG}}$.

Lemma 64 (Regret of UCB). *Let $v^* = \langle v, a^H \rangle$. On the event $\mathcal{E}$, the regret of UCB is at most*

$$O\left(\min\left\{\frac{\log T}{|\Delta - v^*|}, \sqrt{T \log T}\right\}\right).$$

Proof. UCB is run on arms $\mathbf{0}$ and $\hat{a}$, where $\hat{a}$ is the arm returned by MultiGreedy. We therefore first study the expected reward of arm $\hat{a}$. Lemma 55 implies that arm $\hat{a}_\sigma$ is positive and satisfies

$$\begin{aligned}
\langle v, \hat{a} \rangle &\leq \langle v, a^H \rangle + v_{\max} \cdot \varepsilon_{\text{NBS}} + \varepsilon_\gamma \\
&= v^* + v_{\max} \cdot \varepsilon_{\text{NBS}} + \binom{d}{2} \cdot \gamma \cdot \frac{v_{\max}^2}{\theta_{\min}} \\
&\leq v^* + \varepsilon_{\text{NBS}} + \binom{d}{2} \cdot \gamma \cdot \frac{1}{\theta_{\min}}.
\end{aligned}$$

Since we assume event $\mathcal{E}_{\text{sort}}$ happened, we have that MultiGreedy was given a γ-insensitive ordering for $\gamma \leq \frac{\theta_{\min}}{d^2 T}$. Therefore,

$$\begin{aligned}
\langle v, \hat{a} \rangle &\leq v^* + \varepsilon_{\text{NBS}} + \frac{1}{T} \\
&\leq v^* + \frac{2}{T},
\end{aligned}$$

where we use the very loose bound $\varepsilon_{\text{NBS}} \leq \frac{1}{T}$.

Recalling our notation, $\mu_c(\mathbf{0}) = p_0$ is the expected reward of arm $\mathbf{0}$ and $\mu_c(\hat{a})$ is the expected reward of arm $\hat{a}$. We may now conclude that $|\mu_c(\hat{a}) - \mu_c(\mathbf{0})| \geq \min\{|\Delta - v^*|, |\Delta - v^* - 2/T|\}$.

We have from Auer et al. (2002) that the regret of UCB with respect to the set of arms $\{\mathbf{0}, \hat{a}\}$ is

$$\begin{aligned}
&O\left(\min\left\{\frac{\log T}{|\mu_c(\mathbf{0}) - \mu_c(\hat{a})|}, \sqrt{T \log T}\right\}\right) \\
&= O\left(\min\left\{\frac{\log T}{\min\{|\Delta - v^*|, |\Delta - v^* - 2/T|\}}, \sqrt{T \log T}\right\}\right).
\end{aligned}$$

Note that in the above bound, whenever the $\frac{2}{T}$ has a nontrivial effect on the second term in the inner minimum, the $\sqrt{T \log T}$ term must be the smaller term. Hence, we fortunately can present the simplified bound:

$$O\left(\min\left\{\frac{\log T}{|\Delta - v^*|}, \sqrt{T \log T}\right\}\right).$$

Finally, as we instead want the regret with respect to the set of arms $\{\mathbf{0}, a^H\}$, we account for the difference between the expected reward of a^H and $\hat{a}$. But this is at most $\frac{2}{T}$ and hence can contribute only a constant to the regret. □

B.6.4 Proofs of Theorems 34 and 35

Proof (of Theorem 34). From Lemma 65, event $\mathcal{E}_{\text{MD-L}}$ happens with probability at least $1 - 1/T$. In the sequel, consider the case that $\mathcal{E}_{\text{MD-L}}$ happens (if it does not happen, we pick up a regret contribution of at most $T \cdot 1/T = 1$). Since H is optimal and we assume $\Delta \leq \frac{v_{\min}}{16}$, Corollary 67 implies that event $\mathcal{E}_{\text{stop}}$ happens.

The regret due to the pulls of MD-Leftist is bounded by Lemma 40. In addition, since $\mathcal{E}_{\text{stop}}$ happens, Lemmas 60, 61, 62, 63, and 64 respectively imply that the regret is at most

$$
\begin{aligned}
&[\text{MD-Leftist}] + [\text{NBS from MultiEstTau and MultiGreedy}] \\
&\quad + [\text{all calls of MCC}] + [\text{MultiGreedy sparring pulls}] + [\text{UCB}] \\
&= O\left(\log(T) \cdot \frac{(\Delta + \|v\|_1)}{v_{\min}^2} + \frac{d \log T}{\Delta}\right) + O\left(\log(T) \cdot \log(1/\varepsilon_{\text{NBS}}) \cdot \frac{d}{\Delta}\right) \\
&\quad + O\left(\log(T) \cdot \frac{d^2 \log d}{\Delta}\right) + O\left(\log(T) \cdot \frac{d^2}{\Delta}\right) + O\left(\min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v^*|}\right\}\right) \\
&= O\left(\log(T) \cdot \frac{(\Delta + \|v\|_1)}{v_{\min}^2} + \log(T) \cdot \log(1/\varepsilon_{\text{NBS}}) \cdot \frac{d}{\Delta} + \log(T) \cdot \frac{d^2 \log d}{\Delta}\right) + O\left(\min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v^*|}\right\}\right) \\
&= O\left(\log(T) \cdot \left(\frac{\Delta + \|v\|_1}{v_{\min}^2} + \frac{\max\left\{d \log(1/\varepsilon_{\text{NBS}}), d^2 \log d\right\}}{\Delta}\right)\right) + O\left(\min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v^*|}\right\}\right).
\end{aligned}
$$

□

Proof (of Theorem 35). For both results, we use Theorem 43 to bound the regret due to the pulls of MD-Leftist.

We now consider the remaining analysis for each regret bound in turn.

We begin with the first regret bound. For the other algorithms, note that they can be called only if $\Delta = O(\log(T)\sqrt{d/T})$. To see this, observe from Lemma 66 that $\rho \geq r_\Delta + 1$. Hence, MultiEstTau can be called only if $r_\Delta + 1 \leq r_{\max}$, a condition which implies that $\Delta = O(\log(T)\sqrt{d/T})$. We may then take the regret bounds for other algorithms directly from the proof of Theorem 34 together with the substitution $\Delta = O(\log(T)\sqrt{d/T})$ (except for the part for UCB, where the substitution is unnecessary) to get:

$$
\begin{aligned}
&[\text{MD-Leftist}] + [\text{NBS from MultiEstTau and MultiGreedy}] \\
&\quad + [\text{all calls of MCC}] + [\text{MultiGreedy sparring pulls}] + [\text{UCB}] \\
&= O\left(\log(T) \cdot \frac{\Delta + \|v\|_1}{v_{\min}^2} + \sqrt{Td}\right) + O\left(\log(T) \cdot \log(1/\varepsilon_{\text{NBS}}) \cdot \log(T)^{-1}\sqrt{Td}\right) \\
&\quad + O\left(\log(T) \cdot d^2 \log(d) \log^{-1}(T)\sqrt{T/d}\right) + O\left(\log(T) \cdot d^2 \log^{-1}(T)\sqrt{T/d}\right) + O\left(\min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v^*|}\right\}\right) \\
&= O\left(\log(T) \cdot \frac{\Delta + \|v\|_1}{v_{\min}^2} + \log(1/\varepsilon_{\text{NBS}}) \cdot \sqrt{Td} + d^{1.5} \log(d)\sqrt{T}\right) + O\left(\min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v^*|}\right\}\right) \\
&= O\left(\log(T) \cdot \frac{\Delta + \|v\|_1}{v_{\min}^2} + \max\left\{\sqrt{d} \log(1/\varepsilon_{\text{NBS}}), d^{1.5} \log d\right\}\sqrt{T}\right) + O\left(\min\left\{\sqrt{T \log T}, \frac{\log T}{|\Delta - v^*|}\right\}\right).
\end{aligned}
$$

We now turn to the second regret bound, for which we assume that $\Delta \leq \frac{1}{2}\log(T)\sqrt{\frac{d}{T}}$. We claim that with probability at least $1 - 1/T$, MultiEstTau will not be called and hence MD-Leftist will commit to arm $\mathbf{0}$. From Lemma 66, it holds on event $\mathcal{E}_{\text{MD-L}}$ that $\rho \geq r_\Delta + 1$. Moreover, the condition $\Delta \leq \frac{1}{2}\log(T)\sqrt{\frac{d}{T}}$ implies that $r_\Delta + 1 > r_{\max}$. Finally, observe that from Lemma 65, event $\mathcal{E}_{\text{MD-L}}$ happens with probability at least $1 - 1/T$.

Consequently, the regret is at most

$$\begin{aligned}
&[\text{MD-Leftist}] + [\text{commit}] \\
&= O\left(\log(T) \cdot \frac{1}{v_{\min}^2} \cdot (\Delta + \|v\|_1) + \sqrt{Td}\right) + O(\log(T)\sqrt{Td}) \\
&= O\left(\log(T) \cdot \left(\frac{1}{v_{\min}^2} \cdot (\Delta + \|v\|_1) + \sqrt{Td}\right)\right).
\end{aligned}$$

□

C Each event holds with high probability

In this section, we show that the following events happen with suitably high probability:

- one-dimensional setting: $\mathcal{E}_{\text{L}}$ and $\mathcal{E}_{\text{NBS-L}}$;
- multi-dimensional setting: $\mathcal{E}_{\text{MD-L}}$, $\mathcal{E}_{\text{stop}}$, $\mathcal{E}_{\text{NBS-MET}}$, $\mathcal{E}_{\text{NBS-MG}}$, $\mathcal{E}_{\text{sort}}$, and $\mathcal{E}_{\text{MG}}$.

Events $\mathcal{E}_{\text{L}}$ and $\mathcal{E}_{\text{MD-L}}$

Lemma 65. *For each event among $\mathcal{E}_{\text{L}}$ and $\mathcal{E}_{\text{MD-L}}$ separately, the event holds with probability at least $1 - 1/T$.*

Proof. The analysis is the same for either event, and so let us arbitrarily pick $\mathcal{E}_{\text{L}}$. For any epoch $r \le \rho$ for which arm a_r is positive, direct verification via Hoeffding's inequality gives that $|\hat{\Delta}_r - \Delta| \le \varepsilon_r$ with probability at least $1 - \delta = 1 - 1/T^2$. More trivially, if arm a_r is not positive, then both arms are negative and hence the mean of $\hat{\Delta}_r$ is equal to zero; hence, Hoeffding's inequality implies that $\hat{\Delta}_r - \varepsilon_r \le 0 \le \Delta$. As the number of epochs is loosely upper bounded by T, a union bound over all such epochs implies the result. □

Event $\mathcal{E}_{\text{stop}}$

We next show that if Δ is not too large, then event $\mathcal{E}_{\text{stop}}$ holds with high probability. The main step to show this is the following lemma.

Lemma 66. *If $\Delta \le \frac{v_{\min}}{16}$, then on event $\mathcal{E}_{\text{MD-L}}$, we have $\rho \ge \max\{\ell + 1, r_\Delta + 1\}$.*

Proof. We first establish that

$$\rho \ge r_\Delta + 1. \tag{40}$$

We then show that the assumption $\Delta \le \frac{v_{\min}}{16}$ further implies that

$$r_\Delta + 1 \ge \ell + 1, \tag{41}$$

which gives the lemma.

Let us establish (40). For any epoch r, observe that if $\varepsilon_r > \Delta$ is satisfied, then the stopping condition inequality $\hat{\Delta}_r - \varepsilon_r \ge \varepsilon_r$ must be false since this latter inequality would imply that

$$\Delta \ge \hat{\Delta}_r - \varepsilon_r \ge \varepsilon_r > \Delta;$$

here, the first inequality holds on the event $\mathcal{E}_{\text{MD-L}}$. Therefore, let us look for the largest r such that $\varepsilon_r > \Delta$. But by definition this r is equal to r_Δ which, from (7), is equal to $\left\lceil \log_2 \frac{1}{\Delta} \right\rceil - 4$. Inequality (40) now follows.

Next, we establish (41). Observe that the inequality $\Delta \le \frac{v_{\min}}{16}$ is equivalent to

$$\frac{1}{v_{\min}} \le \frac{1}{16\Delta},$$

which implies from the monotonicity of the binary logarithm and the ceiling function that

$$\begin{aligned}\left\lceil \log_2 \frac{1}{v_{\min}} \right\rceil &\le \left\lceil \log_2 \frac{1}{16\Delta} \right\rceil \\ &= \left\lceil \log_2 \frac{1}{\Delta} \right\rceil - 4.\end{aligned}$$

Inequality (41) now follows since the LHS of the above inequality is ℓ and, from (7) the RHS is r_Δ. □

The following corollary is immediate from Lemmas 66 and 38.

Corollary 67. *Suppose that H is optimal and $\Delta \le \frac{v_{\min}}{16}$. If event $\mathcal{E}_{\text{MD-L}}$ holds, then event $\mathcal{E}_{\text{stop}}$ also holds, i.e.,*

$$\ell + 1 \le \rho \le r_\Delta + 3.$$

Events $\mathcal{E}_{\text{NBS-L}}$, $\mathcal{E}_{\text{NBS-MET}}$ and $\mathcal{E}_{\text{NBS-MG}}$

We have from Lemma 30 that event $\mathcal{E}_{\text{NBS-L}}$ happens with probability at least $1 - 1/T$ (since we take $\delta = 1/T^2$). Also, from Lemma 45, for each of the events $\mathcal{E}_{\text{NBS-MET}}$ and $\mathcal{E}_{\text{NBS-MG}}$ separately, the event holds with probability at least $1 - 1/T$.

Event $\mathcal{E}_{\text{sort}}$

Lemma 68. *Event $\mathcal{E}_{\text{sort}}$ holds with probability at least $1 - (4d \log_2 d) \cdot \delta$.*

Proof. Since we set $\varepsilon_{\text{NBS}} = \frac{\theta_{\min}}{d^3 T^2}$ in NBS and $\beta = \frac{1}{T}$ in MultiCoordinateCompare, Corollary 48 implies that MultiCoordinateCompare is a $(\gamma', 2\delta)$-PAC comparison oracle for

$$\gamma' = \frac{\varepsilon_{\text{NBS}}}{\beta} = \frac{\theta_{\min}}{d^3 T}.$$

Consequently, from Corollary 53, PAC-MergeSort produces, with probability at least $1 - (4d \log_2 d) \cdot \delta$, a γ-insensitive ordering for

$$\gamma = (d-1)\gamma' \leq \frac{\theta_{\min}}{d^2 T}.$$

□

Event $\mathcal{E}_{\text{MG}}$

Lemma 69. *Event $\mathcal{E}_{\text{MG}}$ holds with probability at least $1 - d \cdot \delta$.*

Proof. Line 7 of MultiGreedy is used to predict the label of arm R. For each such execution, Lemma 44 implies that, with probability at least $1 - \delta$, the label is correctly predicted. As there can be at most d such executions, the result follows from a union bound. □

D Lower bounds

D.1 Minimax lower bound

In this section, we use the word model to represent the set of parameter $\theta = (p_0, p_1, \tau)$. Let $d(p, q) = p \log(p/q) + (1 - p) \log((1 - p)/(1 - q))$ be the KL divergence between two Bernoulli distributions with parameters p, q.

The following lemma is in parallel to Lemma 19 in Kaufmann et al. (2016).

Lemma 70. *Let θ_0, θ_1 be two models. Let $\mathsf{E}_{\theta_0}, \mathsf{E}_{\theta_1}$ be the corresponding expectations and $\Pr_{\theta_0}, \Pr_{\theta_1}$ be the corresponding probabilities, respectively. Likewise, let $\mu_{\theta_0}, \mu_{\theta_1}$ be the revenue functions under these models. Then, the following inequality holds for any event $\mathcal{E}$.*

$$\mathsf{E}_{\theta_0}\left[\sum_{t=1}^{T} d(\mu_{\theta_0}(a_t), \mu_{\theta_1}(a_t))\right] \geq d(\Pr_{\theta_0}(\mathcal{E}), \Pr_{\theta_1}(\mathcal{E})). \tag{42}$$

We omit the proof of Lemma 70 because it is very similar to Lemma 19[12] in Kaufmann et al. (2016). The following results uses Lemma 70.

In the following, we derive an example in which we have an $\Omega(\sqrt{T})$ regret bound.[13]

Proof of Theorem 3. Assume that $v = 1$. Consider the following two models.

- Model θ_0: $p_0 = 1/2, p_1 = 1/2 + T^{-1/2}, \tau = 2T^{-1/2}$. Arm 0 is the optimal arm in this model.
- Model θ_1: $p_0 = 1/2, p_1 = 1/2 + 3T^{-1/2}, \tau = 2T^{-1/2}$. Arm τ is the optimal arm in this model.

We have

$$d(\mu_{\theta_0}(a_t), \mu_{\theta_1}(a_t)) \leq d\left(\frac{1}{2} + T^{-\frac{1}{2}}, \frac{1}{2} + 3T^{-\frac{1}{2}}\right) = \Theta\left(T^{-1}\right) \tag{43}$$

for any a_t. Therefore, Lemma 70 applied for these two models states that:

$$C = T \cdot \frac{C}{T} \geq \mathsf{E}_{\theta_0}\left[\sum_{t=1}^{T} d(\mu_{\theta_0}(a_t), \mu_{\theta_1}(a_t))\right] \geq d(\Pr_{\theta_0}(\mathcal{E}), \Pr_{\theta_1}(\mathcal{E})) \tag{44}$$

for all $T \geq T_0$ for some constants T_0 and $C > 0$.

Eq. (44) essentially states that the two models are not identifiable. Let $\mathcal{E} = \left\{\sum_{t=1}^{T} \mathbf{1}\left[a_t \leq T^{-1/2}\right] \geq T/2\right\}$ be the event that the algorithm spends at least half of the rounds pulling arms near arm 0. If $\Pr_{\theta_0}(\mathcal{E}) < 1/2$, then the algorithm suffers regret of $T \cdot T^{-1/2} = T^{1/2}$ on model θ_0. Otherwise, (44) implies that $\Pr_{\theta_1}(\mathcal{E}) = \Omega(1)$, which implies that the algorithm suffers regret of $T \cdot T^{-1/2} \cdot \Omega(1) = \Omega(T^{1/2})$ on model θ_1. In summary, the algorithm suffers regret of $\Omega(T^{-1/2})$ either on model θ_0 or θ_1. □

D.2 Trade-off between minimax regret and identifiability

Proof of Theorem 8. Assume that $v = 1$ and $\eta \in (0, 1/2)$. Consider the following pair of models.

- Model θ_0: $p_0 = 1/2, p_1 = 1/2 + T^{-(1-\eta)/2}, \tau = 1$. Arm 0 is the optimal arm in this model.
- Model θ_1: $p_0 = 1/2, p_1 = 1/2 + 2T^{-1/2}, \tau = T^{-1/2}$. Arm τ is the optimal arm in this model.

Let $N_1(t)$ be the number of draws on arm 1. We have

$$d(\mu_{\theta_0}(a_t), \mu_{\theta_1}(a_t)) \leq d\left(\frac{1}{2}, \frac{1}{2} + 2T^{-\frac{1}{2}}\right) = O\left(T^{-1}\right) \tag{45}$$

[12] Essentially, the lemma utilizes the convexity of KL divergence and careful argument of adaptive sequences, which is not specific to K-armed bandit structure.

[13] Note that the instance of Theorem 8 also implies the $\Omega(\sqrt{T})$ regret bound. However, the example here is much simpler.

for $a_t \neq 1$ and

$$d(\mu_{\theta_0}(a_t), \mu_{\theta_1}(a_t)) = d\left(\frac{1}{2} + T^{-\frac{1-\eta}{2}}, \frac{1}{2} + 2T^{-1/2}\right) = \Theta\left(T^{-(1-\eta)}\right) \tag{46}$$

for $a_t = 1$. Therefore, Lemma 70 applied for these two models states that:

$$\mathsf{E}_{\theta_0}\left[\sum_{t=1}^{T} d(\mu_{\theta_0}(a_t), \mu_{\theta_1}(a_t))\right] = C(\mathsf{E}_{\theta_0}[N_1(T)]T^{-(1-\eta)} + 1) \geq d(\Pr_{\theta_0}(\mathcal{E}), \Pr_{\theta_1}(\mathcal{E})) \tag{47}$$

for some $C > 0$. We consider the case $\mathsf{E}_{\theta_0}[N_1(T) \geq T^{1-\eta}]$ and $\mathsf{E}_{\theta_0}[N_1(T) < T^{1-\eta}]$ separately.

Case 1: $\mathsf{E}_{\theta_0}[N_1(T)] \geq T^{1-\eta}$**.** This implies the algorithm has $\Omega(T^{1-\eta})$ regret in model θ_0.

Case 2: $\mathsf{E}_{\theta_0}[N_1(T)] < T^{1-\eta}$**.** In this case, we have

$$d(\Pr_{\theta_0}(\mathcal{E}), \Pr_{\theta_1}(\mathcal{E})) < 2C. \tag{48}$$

Let an event

$$\mathcal{E} = \left\{\sum_t \mathbf{1}\left[a_t < T^{-1/2}\right] \geq T/2\right\}.$$

If $\Pr_{\theta_0}(\mathcal{E}) < 1/2$, then the algorithm suffers regret of $O(\sqrt{T})$ on model θ_0. Otherwise, (48) implies that $\Pr_{\theta_1}(\mathcal{E}) = \Omega(1)$, which implies that the algorithm suffers regret of $\Omega(\sqrt{T})$ on model θ_1. In summary, the algorithm suffers the regret of $\Omega(T^{1/2})$ either on model θ_0 or θ_1, and it draws suboptimal arm for $O(T)$ rounds. □

E Proofs on Explore-the-Gap

We first present the Explore-the-Gap algorithm in detail.

Algorithm 12: Explore-the-Gap Algorithm

Epoch $r \leftarrow 0$, $\varepsilon_0 \leftarrow \frac{1}{8}$
$n_0 \leftarrow \frac{\log \frac{2}{\delta}}{2\varepsilon_0^2}$
while *true* **do**
  Make n_r pulls of arm 0 to get empirical mean $\hat{p}_0$
  Make n_r pulls of arm 1 to get empirical mean $\hat{p}_1$
  $\hat{\Delta}_r \leftarrow \hat{p}_1 - \hat{p}_0$
  if $\hat{\Delta}_r - \varepsilon_r \geq \varepsilon_r$ **then**
    $\hat{\tau} \leftarrow$ NoisyBinarySearch(ε_r, 0, a_r)
    Run UCB on $\{0, \hat{\tau}\}$ until time T
  else
    $\varepsilon_{r+1} \leftarrow \varepsilon_r/2$, $n_{r+1} \leftarrow 4 \cdot n_r$
    $r \leftarrow r + 1$

Let $\mathcal{E}_{\text{R}}$ be the event that for all r, $|\hat{\Delta}_r - \Delta| \leq \varepsilon_r$.

Lemma 71. *Event $\mathcal{E}_{\text{R}}$ holds with probability at least $1 - 1/T$.*

Proof. For any epoch r, Hoeffding's inequality gives that $|\hat{\Delta}_r - \Delta| \leq \varepsilon_r$ with probability at least $1 - \delta = 1 - 1/T^2$, and the union bound over all applications yields $\mathcal{E}_{\text{R}}$. □

E.1 Proof of Theorem 4

This section derives a (poly)-logarithmic distribution-dependent regret bound of EG.

Proof of Theorem 4. We decompose the regret into three components, namely: the regret generated by EG, NoisyBinarySearch (NBS), and UCB.

The following assumes event $\mathcal{E}_{\text{L}}$ that holds with probability at least $1 - 1/T$ by Lemma 71.

Lemma 72. *Let $r_\Delta := \arg\max_{r \geq 0} \{\varepsilon_r > \Delta\}$. Assuming event $\mathcal{E}_{\text{R}}$, EG will stop no later than epoch $r_\Delta + 3$ and invokes NBS.*

Proof of Lemma 72. It suffices to show that if EG reaches epoch $r_\Delta + 3$, then the algorithm stops. Let $r = r_\Delta + 3$. Suppose Leftist reaches epoch r but does not stop. Under event $\mathcal{E}_{\text{R}}$, we have

$$\hat{\Delta}_r - \varepsilon_r \geq \Delta - 2\varepsilon_r \geq 2\varepsilon_r$$

which implies EG stops. □

Since EG stops in epoch $r_\Delta + 3$ or before, the regret incurred by EG is

$$2 \sum_{r=1}^{r_\Delta + 3} n_r \leq O\left(\frac{\log T}{(\varepsilon_{r_\Delta})^2}\right) = O\left(\frac{\log T}{\Delta^2}\right). \tag{49}$$

We next consider NBS. We define ρ to be the stopping epoch of EG. NBS runs for at most $\lfloor \log_2(2/\varepsilon_{\text{NBS}}) \rfloor$ iterations, and at each iteration it compares two arms L and R for N_ρ times. Since $\rho \leq r_\Delta + 3$, the regret due to this component is

$$\lfloor \log_2(2/\varepsilon_{\text{NBS}}) \rfloor \cdot O\left(\frac{\log T}{\Delta^2}\right) = O(\log T) \cdot O\left(\frac{\log T}{\Delta^2}\right) = O\left(\frac{\log^2 T}{\Delta^2}\right). \tag{50}$$

Before analysing UCB, we use the guarantee of the arm that NBS outputs. Note that under $\mathcal{E}_{\text{R}}$, the assumption of NBS (i.e., $\varepsilon_r \leq \Delta$) is satisfied.

Lemma 73. *On event $\mathcal{E}_{\mathrm{R}}$, with probability at least $1 - T \cdot \delta$, NBS will return a positive arm $\hat{\tau}$ satisfying $\hat{\tau} - \tau < \varepsilon_{\mathrm{NBS}}$.*

Proof. First, suppose that NBS did not make a mistake in any iteration. Then since L is negative and R is positive at the end of the last iteration of NBS, NBS returns R and we have $\|R - L\|_2 < \varepsilon_{\mathrm{NBS}}$, it follows that arm a^f is positive and $\hat{\tau} - \tau < \varepsilon_{\mathrm{NBS}}$, as desired.

It remains to control the probability that NBS did not make a mistake in any iteration. First, observe that if NBS did not make a mistake in any iteration prior to iteration i, then arm L is negative and R is positive in iteration i. In this case, $\Delta \geq \varepsilon_r$ and Hoeffding's inequality implies that $R - L < \varepsilon_r/2$ with probability at most δ. Taking a union bound over at most T iterations of binary search bounds the mistake probability by δT. □

Since $\delta T \leq 1/T$ is negligible, we assume $\hat{\tau} - \tau < \varepsilon_{\mathrm{NBS}}$ in the analysis of UCB. Let $\Delta_{\hat{\tau}} = p_1 - p_0 - v\hat{\tau}$. We have $\Delta_{\hat{\tau}} \leq \Delta_\tau - \varepsilon_{\mathrm{NBS}}$. it is well-known that the regret of UCB is

$$O\left(\frac{\log(T)}{\Delta_{\hat{\tau}}}\right) = O\left(\frac{\log(T)}{\Delta_\tau - \varepsilon_{\mathrm{NBS}}}\right) = O\left(\frac{\log(T)}{|\Delta - v\tau - 1/T|}\right) = O\left(\frac{\log(T)}{|\Delta - v\tau|}\right), \tag{51}$$

where we assume $|\Delta - v\tau|$ to be a positive constant in the last transformation. □

E.2 Proof that EG can get linear regret

This section shows the $\Omega(T)$ regret of EG in the worst-case.

Theorem 74. *The worst-case regret of EG is $\mathcal{R}_T = \Omega(T)$.*

Proof of Theorem 74. Consider the model where $p_0 = 1/2, p_1 = 1/2 + 1/T, \tau = 1$, and $v = 1$. In this model, Δ is extremely small. As a result, with a high probability, EG spends all T rounds comparing arm 0 and arm 1 alternately.

For any r we have $\varepsilon_r^2 n_r = \varepsilon_1^2 n_1 = \log(2/\delta)/2 \geq \log(2)/2$. Since n_r is at most T, $\varepsilon_r > (\log(2)/2)T^{-1/2} \geq 1/T$ holds and for $T \geq 2$. With probability $1 - 1/T$, $\mathcal{E}_{\mathrm{R}}$ holds, and thus

$$\hat{\Delta}_r - \varepsilon_r < 1/T - \varepsilon_r + \varepsilon_r \tag{52}$$
$$= 1/T \tag{53}$$
$$\leq \varepsilon_r \tag{54}$$

holds, and thus EG never invokes NBS. Since drawing arm 1 incurs a regret of $1 - 1/T = \Theta(1)$, this implies a $\Theta(T)$ regret. □

F Additional motivating examples

In the main paper, we have discussed the example of credit card offer, where some portion of the customers make a decision based on the comparison with other offers can show a discontinuous behavior. Although we agree that the thresholded linear bandits is a stylized model, we can discuss several works that demonstrate the discontinuity of customer behaviors.

- The utility of an item for a person is dependent on their wages Blisard et al. (2004), and wage is usually discontinuous around the minimum wage. This induces the point mass on the utility. Since a user buys an item when their utility exceeds the price, the demand is discontinuous.
- In many marketplaces, a typical consumer defines the consideration set from which the consumer chooses Caplin et al. (2018). Being the most attractive item among the consideration set is a significant sell to the consumer, which induces a discrete demand. This effect is even more pronounced under recommendation effects (ex: the consideration set is defined by recommender systems).
- In marketing on social media, a slight change in the configuration can result in a drastic difference. Sales of an item can be amplified when the attention of consumers reaches some thresholds. One can create a graph-based bandit model of influence maximization Chen et al. (2013) where a slight change is magnified through cascading on a graph. If we consider such a social graph as a black-box, then that boils down to our thresholded linear bandits.

We can find several other examples in several articles, such as den Boer and Keskin (2020).

G Experiments

G.1 Experimental details and results with EG

In this section, we give more details about how we ran the experiments appearing in the main text, and we also include experimental results for EG.

To evaluate Leftist, we ran three different experiments. For each, we obtained the average cumulative pseudo-regret for $m = 25$ repetitions and compared the algorithm's relative performance against EG and Grid UCB as benchmarks.

We first describe some modifications made to Leftist (and EG) for our experimental results. First, knowing that NBS was the major contributor to the regret, we attempted to reduce the regret incurred by relaxing NBS's precision for finding $\hat{\tau}$ by a factor of $\frac{2\log^2 T}{\varepsilon_r}$, giving $\frac{2\log^2 T}{\varepsilon_r} \cdot \frac{1}{T}$. This changed is motivated by the fact that the induced additional cumulative regret picked up by Leftist and NBS is at most $O\left(\frac{\log^2 T}{\Delta}\right)$, which is not greater than the first term in the regret bound of Theorem 5; we note that we did initially experiment with using $\frac{\log^2 T}{\varepsilon_r} \cdot \frac{1}{T}$ and found that the additional factor of 2 led to a slight improvement in performance. Second, we relaxed δ to $\frac{1}{T \log T}$ for all three experiments because there can be at most $O(\log T)$ applications of Hoeffding's inequality throughout the course of Leftist (or EG) and NBS. The final algorithmic modification was to use $\log T$ to update the upper confidence bound for both Grid UCB and Leftist/EG's UCB as opposed to $\log t$, the overall number of pulls done so far. Although $\log t$ and $\log T$ give the same regret bound, the latter would not require us to update every index in each round, thereby simplifying and speeding up the algorithm.

In the first experiment (Figure 4 top left), linear increments of 0.01 were selected for τ from $[0.01, 0.4]$ while keeping $\Delta = 0.01$ and $T = 1 \times 10^6$. This gave us 40 different settings. The second experiment, shown in Figure 4 top right, tested 30 values of Δ within $[5 \times 10^{-3}, 0.55]$ where Δ geometrically increases by a factor of 1.176. Here, $\tau = 0.1$ and $T = 1 \times 10^6$. Given that δ and the number of pulls, N, for Leftist was dependent on the time horizon T, we decided to

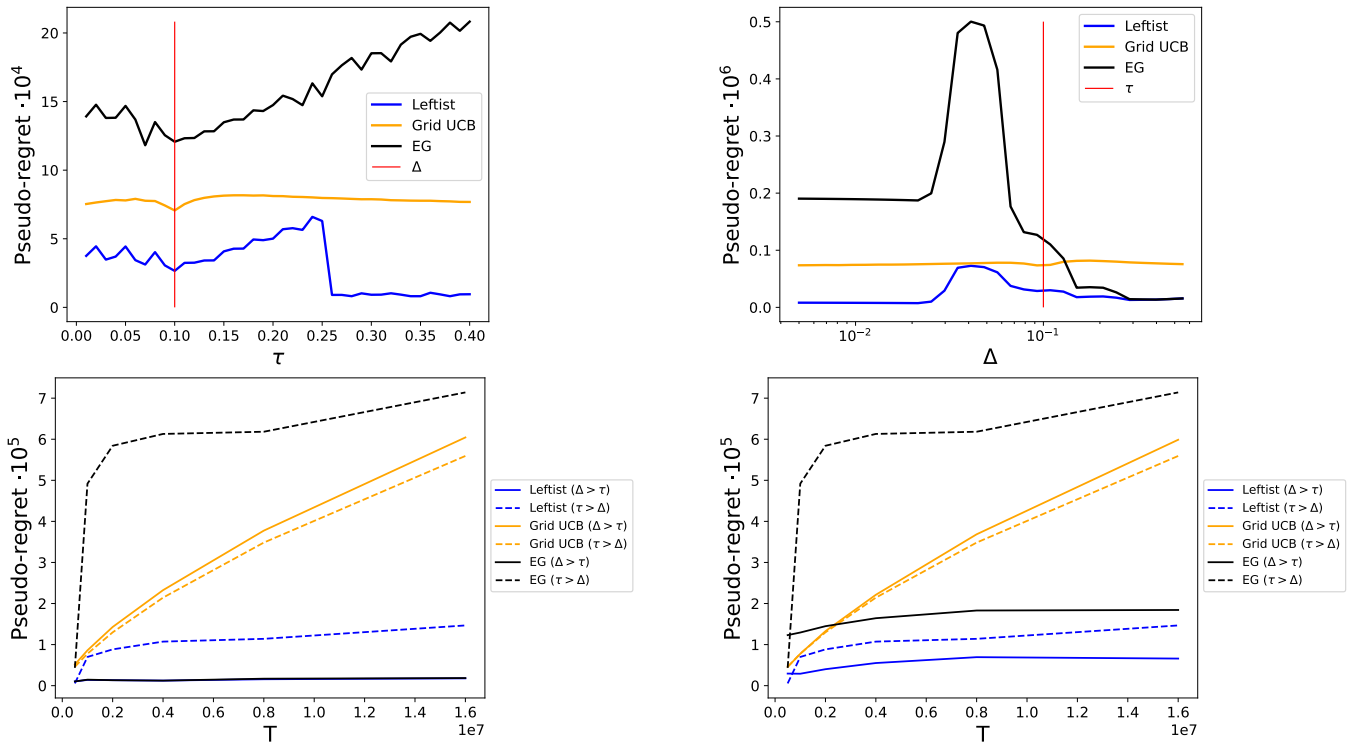


Figure 4: Top: Average cumulative pseudo-regret ($m = 25$) of Leftist, Grid UCB, and EG in experiments varying τ (top left) and varying Δ (top right). Bottom: (Left) Average cumulative pseudo-regret ($m = 25$) for varying time horizon, T. The repeat of experiment 3 with a smaller $\Delta = 0.1$ and $\tau = 0.075$ for the $\Delta > \tau$ is shown in the bottom right figure.

evaluate the algorithm's performance on 6 values of T, where T was increasing geometrically within $[5 \times 10^5, 1.6 \times 10^7]$ by a factor of 2. For this experiment, the algorithms were tested for both cases $\Delta > \tau$ with $\Delta = 0.1$ and $\tau = 0.075$, and $\tau > \Delta$ with $\tau = 0.1$ and $\Delta = 0.05$, see Figure 4 bottom left. To clearly demonstrate the difference between Leftist and EG, we repeated experiment 3 but with a smaller $\Delta = 0.1$ and $\tau = 0.075$ for the $\Delta > \tau$, shown in Figure 4 bottom right. In all experiments, we observed that EG was generally outperformed by Leftist.

G.2 On the worst-case regret of Grid-UCB

For completeness, we also sketch our argument for why Grid-UCB (run with grid resolution $\varepsilon = 1/\sqrt{T}$) should have worst-case $\tilde{O}(\sqrt{T})$ regret. We do not give a complete proof of this fact for two reasons:

1. Grid-UCB is not a contribution of our work.

2. Our purpose is merely to show that on an instance that intentionally is constructed to be hard for Grid-UCB — one in which we try to maximize the number of arms (in the grid) for which the gap is close to $1/\sqrt{T}$ — Grid-UCB still obtains $\tilde{O}(\sqrt{T})$ regret. Since it's worst-case regret appears to be good, Grid-UCB seems a worthy competitor for Leftist.

Lemma 75 (Regret bound of UCB, Theorem 1 in Auer et al. (2002))**.** *The regret UCB1 when we run it on a Bernoulli K-armed bandit problem with parameters $(\mu_1, \mu_2, \ldots, \mu_K)$ is bounded as*

$$\sum_{i \geq 2} \frac{8 \log T}{\Delta_i} + \left(1 + \frac{\pi^2}{3}\right) \sum_{i \geq 2} \Delta_i,$$

where we assume $\mu_1 = \max_i \mu_i$ and $\Delta_i = \mu_1 - \mu_i > 0$.

We give a construction which intuitively captures the worst case for Grid-UCB. We take τ to be optimal, as in this case Grid-UCB (which naturally always includes arm 0) is likely to suffer some approximation error due to not exactly including arm τ. Specifically, we take[14] $\tau = 1/2$ and $\Delta = \tau + T^{-1/2}$.

Proposition 76. *On the problem instance above, Grid-UCB with $\varepsilon = 1/\sqrt{T}$ has regret at most*

$$\mathcal{R}_T = O\left(\log^2(T)\sqrt{T}\right).$$

Proof. To bound the regret of Grid-UCB, it suffices to apply the regret bound in Lemma 75 and to further add an approximation error term to account for the closest grid point greater than or equal to τ; the contribution of the latter is at most $T \cdot \varepsilon$ in the worst case.

The structure of the thresholded linear bandits problem means the instantaneous pseudo-regret (gap) of the arms, when going from left to right (from arm 0 to arm 1) linearly increases up to and excluding arm τ, drops to zero at arm τ, and then linearly increases again up to arm 1. Hence, we split the above regret bound into the summation over Grid-UCB's arms that are strictly less than τ and its arms that are strictly greater than τ; we refer to these two sets as $\mathcal{A}_0$ and $\mathcal{A}_1$ respectively. To be clear, for fixed ε, we have the definitions

$$\mathcal{A}_0 := \{0, \varepsilon, 2\varepsilon, \ldots, k_0\varepsilon\} \qquad \mathcal{A}_1 := \{(k_0+1)\varepsilon, (k_0+2)\varepsilon, \ldots, k_1\varepsilon\},$$

where k_0 is the largest integer such that $k_0\varepsilon < \tau$ and k_1 is the largest integer such that $k_1\varepsilon \leq 1$.

[14] This particular choice of $1/2$ is unimportant: any suitably small positive constant (e.g. $\tau = 0.1$) would be fine as well.

Bounding regret from arms in $\mathcal{A}_0$.

The logarithmic term (the one involving $\log T$) in Lemma 75 can be bounded as

$$\begin{aligned}\sum_{i\in\mathcal{A}_0}\frac{\log T}{\Delta_i} &\le (\log T)\sum_{j=0}^{\tau/\varepsilon}\frac{1}{T^{-1/2}+j\cdot\varepsilon}\\ &\le (\log T)\left(\sqrt{T}+\int_0^{\tau/\varepsilon}\frac{1}{T^{-1/2}+\varepsilon x}dx\right)\\ &= (\log T)\left(\sqrt{T}+\log(T^{-1/2}+\tau)-\log(T^{-1/2}\right)\\ &= O\left((\log T)\sqrt{T}\right).\end{aligned}$$

The constant term (which is the other term) can be bounded as

$$\begin{aligned}\sum_{i\in\mathcal{A}_0}\Delta_i &\le \sum_{j=0}^{\tau/\varepsilon}\left(T^{-1/2}+j\cdot\varepsilon\right)\\ &= O\left(\frac{\tau}{\varepsilon}\cdot\left(T^{-1/2}+\frac{\tau}{\varepsilon}\cdot\varepsilon\right)\right)=O\left(\frac{1}{\varepsilon}\right).\end{aligned}$$

Bounding regret from arms in $\mathcal{A}_1$.

The logarithmic term can be bounded as

$$\begin{aligned}\sum_{i\in\mathcal{A}_1}\frac{\log T}{\Delta_i} &\le (\log T)\sum_{j=1}^{1/\varepsilon}\frac{1}{j\cdot\varepsilon}\\ &\le (\log T)\left(\frac{1}{\varepsilon}+\int_1^{1/\varepsilon}\frac{1}{\varepsilon x}dx\right)\\ &= (\log T)\left(\frac{1}{\varepsilon}+\frac{1}{\varepsilon}\log\frac{1}{\varepsilon}\right)\\ &= \frac{\log T}{\varepsilon}\cdot\left(1+\log\frac{1}{\varepsilon}\right).\end{aligned}$$

The constant term can be bounded as

$$\begin{aligned}\sum_{i\in\mathcal{A}_1}\Delta_i &\le \sum_{j=1}^{1/\varepsilon} j\cdot\varepsilon\\ &\le \frac{1}{\varepsilon}\cdot\left(\frac{1}{\varepsilon}\cdot\varepsilon\right)=\frac{1}{\varepsilon}.\end{aligned}$$

Total regret bound for Grid-UCB.

Considering the above four terms, and adding in the approximation error price of $T\varepsilon$, gives that the total regret of Grid-UCB on this problem instance is of order at most

$$(\log T)\sqrt{T}+\frac{\log T}{\varepsilon}\cdot\left(1+\log\frac{1}{\varepsilon}\right)+\frac{1}{\varepsilon}+T\cdot\varepsilon.$$

Setting $\varepsilon=1/\sqrt{T}$ yields $O\left(\log^2(T)\sqrt{T}\right)$ regret. □