Adaptive Multiple-Arm Identification[*]

Jiecao Chen

Computer Science Department

Indiana University at Bloomington

jiecchen@umail.iu.edu

Xi Chen

Leonard N. Stern School of Business

New York University

xchen3@stern.nyu.edu

Qin Zhang

Computer Science Department

Indiana University at Bloomington

qzhangcs@indiana.edu

Yuan Zhou

Computer Science Department

Indiana University at Bloomington

yzhoucs@indiana.edu

June 6, 2017

Abstract

We study the problem of selecting K arms with the highest expected rewards in a stochastic n-armed bandit game. This problem has a wide range of applications, e.g., A/B testing, crowdsourcing, simulation optimization. Our goal is to develop a PAC algorithm, which, with probability at least $1 - \delta$, identifies a set of K arms with the aggregate regret at most ϵ. The notion of aggregate regret for multiple-arm identification was first introduced in Zhou et al. [2014] , which is defined as the difference of the averaged expected rewards between the selected set of arms and the best K arms. In contrast to Zhou et al. [2014] that only provides instance-independent sample complexity, we introduce a new hardness parameter for characterizing the difficulty of any given instance. We further develop two algorithms and establish the corresponding sample complexity in terms of this hardness parameter. The derived sample complexity can be significantly smaller than state-of-the-art results for a large class of instances and matches the instance-independent lower bound upto a $\log(\epsilon^{-1})$ factor in the worst case. We also prove a lower bound result showing that the extra $\log(\epsilon^{-1})$ is necessary for instance-dependent algorithms using the introduced hardness parameter.

1 Introduction

Given a set of alternatives with different quality, identifying high quality alternatives via a sequential experiment is an important problem in multi-armed bandit (MAB) literature, which is also known as the "pure-exploration" problem. This problem has a wide range of applications. For example, consider the A/B/C testing problem with multiple website designs, where each candidate design corresponds to an alternative. In order to select high-quality designs, an agent could display different designs to website visitors and measure the attractiveness of an design. The question is: how should the agent adaptively select which design to be displayed next so that the high-quality designs can be quickly and accurately identified? For another example, in crowdsourcing, it is critical to identify high-quality workers from

---

[*]Author names are listed in alphabetical order. Preliminary version to appear in ICML 2017.

1

a pool of a large number of noisy workers. An effective strategy is testing workers by gold questions, i.e., questions with the known answers provided by domain experts. Since the agent has to pay a fixed monetary reward for each answer from a worker, it is important to implement a cost-effective strategy for to select the top workers with the minimum number of tests. Other applications include simulation optimization, clinical trials, etc.

More formally, we assume that there are n alternative arms, where the i-th arm is associated with an unknown reward distribution $\mathcal{D}_i$ with mean θ_i. For the ease of illustration, we assume each $\mathcal{D}_i$ is supported on $[0, 1]$. In practice, it is easy to satisfy this assumption by a proper scaling. For example, the traffic of a website or the correctness of an answer for a crowd worker (which simply takes the value either 0 or 1), can be scaled to $[0, 1]$. The mean reward θ_i characterizes the quality of the i-th alternative. The agent sequentially pulls an arm, and upon each pulling of the i-th arm, the $i.i.d.$ reward from $\mathcal{D}_i$ is observed. The goal of "top-K arm identification" is to design an adaptive arm pulling strategy so that the top K arms with the largest mean rewards can be identified with the minimum number of trials. In practice, identifying the exact top-K arms usually requires a large number of arm pulls, which could be wasteful. In many applications (e.g., crowdsourcing), it is sufficient to find an "*approximate set*" of top-K arms. To measure the quality of the selected arms, we adopt the notion of *aggregate regret* (or regret for short) from Zhou et al. [2014]. In particular, we assume that arms are ordered by their mean $\theta_1 \geq \theta_2 \geq \cdots \geq \theta_n$ so that the set of the best K arms is $\{1, \ldots, K\}$. For the selected arm set T with the size $|T| = K$, the aggregate regret $\mathcal{R}_T$ is defined as,

$$\mathcal{R}_T = \frac{1}{K} \left(\sum_{i=1}^{K} \theta_i - \sum_{i \in T} \theta_i \right). \tag{1}$$

The set of arms T with the aggregate regret less than a pre-determined tolerance level ϵ (i.e. $\mathcal{R}_T \leq \epsilon$) is called ϵ-top-K arms. In this paper, we consider the ϵ-top-K-arm problem in the "fixed-confidence" setting: given a target confidence level $\delta > 0$, the goal is to find a set of ϵ-top-K arms with the probability at least $1 - \delta$. This is also known as the PAC (probably approximately correct) learning setting. We are interested in achieving this goal with as few arm pulls (sample complexity) as possible.

To solve this problem, Zhou et al. [2014] proposed the OptMAI algorithm and established its sample complexity $\Theta\left(\frac{n}{\epsilon^2} \left(1 + \frac{\ln \delta^{-1}}{K}\right)\right)$, which is shown to be asymptotically optimal. However, the algorithm and the corresponding sample complexity in Zhou et al. [2014] are *non-adaptive* to the underlying instance. In other words, the algorithm does not utilize the information obtained in known samples to adjust its future sampling strategy; and as a result, the sample complexity only involves the parameters K, n, δ and ϵ but is independent of $\{\theta_i\}_{i=1}^{n}$. Chen et al. [2014] developed the CLUCB-PAC algorithm and established an instance-dependent sample complexity for a more general class of problems, including the ϵ-top-K arm identification problem as one of the key examples. When applying the CLUCB-PAC algorithm to identify ϵ-top-K arms, the sample complexity becomes $O((\log H^{(0,\epsilon)} + \log \delta^{-1})H^{(0,\epsilon)})$ where $H^{(0,\epsilon)} = \sum_{i=1}^{n} \min\{(\Delta_i)^{-2}, \epsilon^{-2}\}$, $\Delta_i = \theta_i - \theta_{K+1}$ for $i \leq K$, $\Delta_i = \theta_K - \theta_i$ for $i > K$. The reason why we adopt the notation $H^{(0,\epsilon)}$ will be clear from Section 1.1. However, this bound may be improved for the following two reasons. First, intuitively, the hardness parameter $H^{(0,\epsilon)}$ is the total number of necessary pulls needed for each arm to identify whether it is among the top-K arms or the rest so that the algorithm can decide whether to accept or reject the arm (when the arm's mean is ϵ-close to the boundary between the top-K arms and the rest arms, it can be either selected or rejected). However, in many cases, even if an arm's mean is ϵ-far from the boundary, we may still be vague about the comparison between its mean and the boundary, i.e. either selecting or rejecting the arm satisfies the aggregate regret bound. This may lead to fewer number of pulls and a smaller hardness parameter for the same instance. Second, the worst-case sample complexity for CLUCB-PAC becomes $O((\log n + \log \epsilon^{-1} + \log \delta^{-1})n\epsilon^{-2})$. When δ is a constant, this bound is $\log n$ times more than the best non-adaptive algorithm in Zhou et al. [2014].

In this paper, we explore towards the above two directions and introduce new instance-sensitive algorithms for the problem of identifying ϵ-top-K arms. These algorithms significantly improve the sample complexity by CLUCB-PAC for many common instances and almost match the best non-adaptive algorithm in the worst case.

Specifically, we first introduce a new parameter H to characterize the hardness of a given instance. This new hardness parameter H could be smaller than the hardness parameter $\widetilde{H}$ used in the literature, in many natural instances. For example, we show in Lemma 1 that when $\{\theta_i\}_{i=1}^n$ are sampled from a continuous distribution with bounded probability density function (which is a common assumption in Bayesian MAB and natural for many applications), for $K = \gamma n$ with $\gamma \leq 0.5$, our hardness parameter $H = O(n/\sqrt{\epsilon})$ while $\widetilde{H} = \Omega(n/\epsilon)$.

Using this new hardness parameter H, we first propose an easy-to-implement algorithm– ADAPTIVETOPK and relate its sample complexity to H. In Theorem 1, we show that ADAPTIVETOPK uses $O\left(\left(\log\log(\epsilon^{-1}) + \log n + \log \delta^{-1}\right)H\right)$ to identify ϵ-top-K arms with probability at least $1-\delta$. Note that this bound has a similar form as the one in Chen et al. [2014], but as mentioned above, we have an $\sqrt{\epsilon}$-factor improvement in the hardness parameter for those instances where Lemma 1 applies.

We then propose the second algorithm (IMPROVEDTOPK) with even less sample complexity, which removes the $\log n$ factor in the sample complexity. In Theroem 2, we show that the algorithm uses $O\left(\left(\log \epsilon^{-1} + \log \delta^{-1}\right)H\right)$ pulls to identify ϵ-top-K arms with probability $1 - \delta$. Since H is always $\Omega(n/\epsilon^2)$ (which will be clear when the H is defined in Section 1.1), the worst-case sample complexity of IMPROVEDTOPK matches the best instance-independent shown in Zhou et al. [2014] up to an extra $\log(\epsilon^{-1})$ factor (for constant δ). We are also able to show that this extra $\log(\epsilon^{-1})$ factor is a necessary expense by being instance-adaptive (Theorem 3). It is also noteworthy that as a by-product of establishing IMPROVEDTOPK, we developed an algorithm that approximately identifies the k-th best arm, which may be of independent interest. Please see Algorithm 2 for details.

We are now ready to introduce our new hardness parameters and summarize the main results in technical details.

1.1 Summary of Main Results

Following the existing literature (see, e.g., Bubeck et al. [2013]), we first define the gap of the i-th arm

$$\Delta_i(K) = \begin{cases} \theta_i - \theta_{K+1} & \text{if } i \leq K \\ \theta_K - \theta_i & \text{if } i \geq K + 1. \end{cases} \tag{2}$$

Note that when $K = 1$, $\Delta_i(K)$ becomes $\theta_1 - \theta_i$ for all $i \geq 2$ and $\Delta_1(K) = \theta_1 - \theta_2$. When K is clear from the context, we simply use Δ_i for $\Delta_i(K)$. One commonly used hardness parameter for quantifying the sample complexity in the existing literature (see, e.g., Bubeck et al. [2013], Karnin et al. [2013]) is $\widetilde{H} \triangleq \sum_{i=1}^n \Delta_i^{-2}$. If there is an extremely small gap Δ_i, the value of $\widetilde{H}$ and thus the corresponding sample complexity can be super large. This hardness parameter is natural when the goal is to identify the exact top-K arms, where a sufficient gap between an arm and the boundary (i.e. θ_K and θ_{K+1}) is necessary. However, in many applications (e.g., finding high-quality workers in crowdsourcing), it is an overkill to select the exact top-K arms. For example, if all the top-M arms with $M > K$ have very close means, then any subset of them of size K forms an ϵ-top-K set in terms of the aggregate regret in (1). Therefore, to quantify the sample complexity when the metric is the aggregate regret, we need to construct a new hardness parameter.

Given K and an error bound ϵ, let us define $t = t(\epsilon, K)$ to be the largest $t \in \{0, 1, 2, \ldots, K - 1\}$ such

that

$$\Delta_{K-t} \cdot t \le K\epsilon \qquad \text{and} \qquad \Delta_{K+t+1} \cdot t \le K\epsilon. \tag{3}$$

Note that $\Delta_{K-t} \cdot t = (\theta_{K-t} - \theta_{K+1}) \cdot t$ upper-bounds the total gap of the t worst arms in the top K arms and $\Delta_{K+t+1} \cdot t = (\theta_K - \theta_{K+t+1}) \cdot t$ upper-bounds the total gap of the t best arms in the non-top-K arms. Intuitively, the definition in (3) means that we can tolerate exchanging at most t best arms in the non-top-K arms with the t worst arms in the top-K arms.

Given $t = t(\epsilon, K)$, we define

$$\Psi_t = \min(\Delta_{K-t}, \Delta_{K+t+1}), \tag{4}$$

and

$$\Psi_t^\epsilon = \max(\epsilon, \Psi_t). \tag{5}$$

We now introduce the following parameter to characterize the hardness of a given instance,

$$H = H^{(t,\epsilon)} = \sum_{i=1}^n \min\{(\Delta_i)^{-2}, (\Psi_t^\epsilon)^{-2}\}. \tag{6}$$

It is worthwhile to note that in this new definition of hardness parameter, no matter how small the gap Δ_i is, since $\Psi_t^\epsilon \ge \epsilon$, we always have $H^{(t,\epsilon)} \le n\epsilon^{-2}$. We also note that since Ψ_t is non-decreasing in t, $H^{(t,\epsilon)}$ is non-increasing in t.

Our first result is an easy-to-implement algorithm (see Algorithm 1) that identifies ϵ-top-K arms with sample complexity related to $H^{(t,\epsilon)}$.

Theorem 1 *There is an algorithm that computes ϵ-top-K arms with probability at least $(1 - \delta)$, and pulls the arms at most $O\left(\left(\log\log\epsilon^{-1} + \log n + \log\delta^{-1}\right) H^{(t,\epsilon)}\right)$ times.*

We also develop a more sophisticated algorithm (see Algorithm 5) with an improved sample complexity.

Theorem 2 *There is an algorithm that computes ϵ-top-K arms with probability at least $(1 - \delta)$, and pulls the arms at most $O\left(\left(\log\epsilon^{-1} + \log\delta^{-1}\right) H^{(t,\epsilon)}\right)$ times.*

Since $\Psi_t^\epsilon \ge \epsilon$ and $H^{(t,\epsilon)} \le n\epsilon^{-2}$, the worst-case sample complexity by Theorem 2 is $O\left(\frac{n}{\epsilon^2}\left(\log\epsilon^{-1} + \log\delta^{-1}\right)\right)$. While the asymptotically optimal instance-independent sample complexity is $\Theta\left(\frac{n}{\epsilon^2}\left(1 + \frac{\ln\delta^{-1}}{K}\right)\right)$, we show that the $\log\epsilon^{-1}$ factor in Theorem 2 is necessary for instance-dependent algorithms using $H^{(t,\epsilon)}$ as a hardness parameter. In particular, we prove the following lower-bound result.

Theorem 3 *For any n, K such that $n = 2K$, and any $\epsilon = \Omega(n^{-1})$, there exists an instance on n arms so that $H^{(t,\epsilon)} = \Theta(n)$ and it requires $\Omega(n\log\epsilon^{-1})$ pulls to identify a set of ϵ-top-K arms with probability at least 0.9.*

Note that since $H^{(t,\epsilon)} = \Theta(n)$ in our lower bound instances, our Theorem 3 shows that the sample complexity has to be at least $\Omega(H^{(t,\epsilon)}\log\epsilon^{-1})$ in these instances. In other words, our lower bound result shows that for any instance-dependent algorithm, and any $\epsilon = \Omega(n^{-1})$, there exists an instance where sample complexity has to be $\Omega(H^{(t,\epsilon)}\log\epsilon^{-1})$. While Theorem 3 shows the necessity of the $\log\epsilon^{-1}$ factor in Theorem 2, it is not a lower bound for *every* instance of the problem.

1.2 Review of and Comparison with Related Works

The problem of identifying the single best arm (i.e. the top-K arms with $K = 1$), has been studied extensively [Even-Dar et al., 2002, Mannor and Tsitsiklis, 2004, Audibert et al., 2010, Gabillon et al., 2011, 2012, Karnin et al., 2013, Jamieson et al., 2014, Kaufmann et al., 2016, Russo, 2016, Chen et al., 2016b]. More specifically, in the special case when $K = 1$, our problem reduces to identifying an ϵ-best arm, i.e. an arm whose expected reward is different from the best arm by an additive error of at most ϵ, with probability at least $(1-\delta)$. For this problem, Even-Dar et al. [2006] showed an algorithm with an instance-independent sample complexity $O\left(\frac{n}{\epsilon^2} \log \delta^{-1}\right)$ (and this was proved to be asymptotically optimal by Mannor and Tsitsiklis [2004]). An instance-dependent algorithm for this problem was given by Bubeck et al. [2013] and an improved algorithm was given by Karnin et al. [2013] with an instance-dependent sample complexity of $O\left(\sum_{i=2}^{n} \max\{\Delta_i, \epsilon\}^{-2}(\log \delta^{-1} + \log\log \max\{\Delta_i, \epsilon\}^{-1})\right)$. In the worst case, this bound becomes $O\left(\frac{n}{\epsilon^2}(\log \delta^{-1} + \log\log \epsilon^{-1})\right)$, almost matching the instance-independent bound in Even-Dar et al. [2006]. When $K = 1$, we have $t(\epsilon, K) = 0$ and thus $H^{(t,\epsilon)} = H^{(0,\epsilon)} = \Theta\left(\sum_{i=2}^{n} \max\{\Delta_i, \epsilon\}^{-2}\right)$. Therefore, the sample complexity in our Theorem 2 becomes $O((\log \epsilon^{-1} + \log \delta^{-1})H) = O\left(\frac{n}{\epsilon^2}(\log \epsilon^{-1} + \log \delta^{-1})\right)$ in the worst-case, almost matching the bound by Karnin et al. [2013].

For the problem of identifying top-K arms with $K > 1$, different notions of ϵ-optimal solution have been proposed. One popular metric is the misidentification probability (MISPROB), i.e. $\mathbf{Pr}(T \neq \{1, \ldots, K\})$. In the PAC setting (i.e. controlling MISPROB less than ϵ with probability at least $1 - \delta$), many algorithms have been developed recently, e.g., Bubeck et al. [2013] in the fixed budget setting and Chen et al. [2014] for both fixed confidence and fixed budget settings. Gabillon et al. [2016] further improved the sample complexity in Chen et al. [2014]; however the current implementations of their algorithm have an exponential running time. As argued in Zhou et al. [2014], the MISPROB requires to identify the exact top-K arms, which might be too stringent for some applications (e.g., crowdsourcing). The MISPROB requires a certain gap between θ_K and θ_{K+1} to identify the top-K arms, and this requirement is not unnecessary when using the aggregate regret. As shown in Zhou et al. [2014], when the gap of any consecutive pair between θ_i and θ_{i+1} among the first $2K$ arms is $o(1/n)$, the sample complexity has to be huge ($\omega(n^2)$) to make the MISPROB less than ϵ, while any K arms among the first $2K$ form a desirably set of ϵ-top-K arms in terms of aggregate regret. Therefore, we follow Zhou et al. [2014] and adopt the aggregate regret to define the approximate solution in this paper.

Kalyanakrishnan et al. [2012] proposed the so-called EXPLORE-K metric, which requires for each arm i in the selected set T to satisfy $\theta_i \geq \theta_K - \epsilon$, where θ_K is the mean of the K-th best arm. Cao et al. [2015] proposed a more restrictive notion of optimality—ELEMENTWISE-ϵ-OPTIMAL, which requires the mean reward of the i-th best arm in the selected set T be at least $\theta_i - \epsilon$ for $1 \leq i \leq K$. It is clear that the ELEMENTWISE-ϵ-OPTIMAL is a stronger guarantee than our ϵ-top-K in regret, while the latter is stronger than EXPLORE-K. Chen et al. [2016a] further extended Cao et al. [2015] to pure exploration problems under matroid constraints. Audibert et al. [2010] and Bubeck et al. [2013] considered expected aggregate regret (i.e. $\frac{1}{K}\left(\sum_{i=1}^{K} \theta_i - \mathbf{E}\left(\sum_{i \in T} \theta_i\right)\right)$), where the expectation is taken over the randomness of the algorithm. Note that this notion of expected aggregate regret is a weaker objective than the aggregate regret.

Moreover, there are some other recent works studying the problem of best-arm identification in different setups, e.g., linear contextual bandit [Soare et al., 2014], batch arm pulls [Jun et al., 2016].

For our ϵ-top-K arm problem, the state-of-the-art instance-dependent sample complexity was given by Chen et al. [2014] (see Section B.2 in Appendix of their paper). More specifically, Chen et al. [2014] proposed CLUCB-PAC algorithms that finds ϵ-top-K arms with probability at least $(1 - \delta)$ using $O\left(\left(\log \delta^{-1} + \log H^{(0,\epsilon)}\right) H^{(0,\epsilon)}\right)$ pulls. Since we always have $H^{(0,\epsilon)} \geq H^{(t,\epsilon)} \geq \Omega(n)$ and $H^{(0,\epsilon)} \geq (\Psi_t^\epsilon)^{-2}$, our Theorem 1 is not worse than the bound in Chen et al. [2014]. Indeed, in many common settings, $H^{(t,\epsilon)}$ can be much smaller than $H^{(0,\epsilon)}$ so that Theorem 1 (and therefore Theorem 2) requires

much less sample complexity. We explain this argument in more details as follows.

In many real-world applications, it is common to assume the arms θ_i are sampled from a prior distribution $\mathcal{D}$ over $[0, 1]$ with cumulative distribution function $F_{\mathcal{D}}(\theta)$. In fact, this is the most fundamental assumption in Bayesian multi-armed bandit literature (e.g., best-arm identification in Bayesian setup Russo [2016]). In crowdsourcing applications, Chen et al. [2015] and Abbasi-Yadkori et al. [2015] also made this assumption for modeling workers' accuracy, which correspond to the expected rewards. Under this assumption, it is natural to let θ_i be the $(1 - \frac{i}{n})$ quantile of the distribution $\mathcal{D}$, i.e. $F_{\mathcal{D}}^{-1}(1 - \frac{i}{n})$. If the prior distribution $\mathcal{D}$'s probability density function $f_{\mathcal{D}} = \frac{\mathrm{d}F_{\mathcal{D}}}{\mathrm{d}\theta}$ has bounded value (a few common examples include uniform distribution over $[0, 1]$, Beta distribution, or the truncated Gaussian distribution), the arms' mean rewards $\{\theta_i\}_{i=1}^n$ can be characterized by the following property with $c = O(1)$.

Definition 1 *We call a set of n arms $\theta_1 \geq \theta_2 \geq \cdots \geq \theta_n$ c-spread (for some $c \geq 1$) if for all $i, j \in [n]$ we have $|\theta_i - \theta_j| \in \left[\frac{|i-j|}{cn}, \frac{c|i-j|}{n} \right]$.*

The following lemma upper-bounds $H^{(t,\epsilon)}$ for $O(1)$-spread arms, and shows the improvement of our algorithms compared to Chen et al. [2014] on $O(1)$-spread arms.

Lemma 1 *Given a set of n c-spread arms, let $K = \gamma n \leq \frac{n}{2}$. When $c = O(1)$ and $\gamma = \Omega(1)$, we have $H^{(t,\epsilon)} = O(n/\sqrt{\epsilon})$. In contrast, $H^{(0,\epsilon)} = \Omega(n/\epsilon)$ for $O(1)$-spread arms and every $K \in [n]$.*

Proof: Given a set of n c-spread arms, we have $\frac{t+1}{cn} \leq \Delta_{K-t} \leq \frac{c(t+1)}{n}$ and $\frac{t+1}{cn} \leq \Delta_{K+t+1} \leq \frac{c(t+1)}{n}$. Therefore $t = t(\epsilon, K) \in [\sqrt{Kn\epsilon/c}, \sqrt{cKn\epsilon} - 1] = [\sqrt{\gamma\epsilon/c}n, \sqrt{c\gamma\epsilon}n - 1]$, and $\Psi_t \geq \frac{t+1}{cn} \geq \sqrt{\gamma\epsilon/c^3}$. Therefore

$$
H^{(t,\epsilon)} \leq O(1) \sum_{i=1}^n \min\left\{ \frac{ci}{n}, \Psi_t \right\}^{-2} \leq O\left(t \cdot \Psi_t^{-2} + \sum_{i=t+1}^n \left(\frac{i}{cn} \right)^{-2} \right)
$$

$$
= O\left(t \cdot \Psi_t^{-2} + c^2 n^2 / t \right) = O(\sqrt{c\gamma\epsilon}n) \cdot \frac{c^3}{\gamma\epsilon} + O\left(\frac{c^2 n}{\sqrt{\gamma\epsilon/c}} \right) = O(c^{3.5}\gamma^{-0.5}) \cdot \frac{n}{\sqrt{\epsilon}}.
$$

One the other hand, we have

$$
H^{(0,\epsilon)} \geq \sum_{i=1}^{n-K} \min\{\Delta_{i+K}^{-2}, \epsilon^{-2}\} \geq \sum_{i=1}^{n/2} \min\left\{ \frac{n^2}{c^2 i^2}, \epsilon^{-2} \right\} = \sum_{i=1}^{[\epsilon n/c]} \epsilon^{-2} + \sum_{[\epsilon n/c]+1}^{n/2} \frac{n^2}{c^2 i^2} = \Omega\left(\frac{n}{c\epsilon} \right).
$$

$\square$

2 An Instance Dependent Algorithm for ϵ-top-K Arms

In this section, we show Theorem 1 by proving the following theorem.

Theorem 4 *Algorithm 1 computes ϵ-top-K arms with probability at least $1 - \delta$, and pulls the arms at most*

$$
O\left(\left(\log\log(\Delta_t^\epsilon)^{-1} + \log n + \log \delta^{-1} \right) \sum_{i=1}^n \min\{(\Delta_i)^{-2}, (\Delta_t^\epsilon)^{-2}\} \right)
$$

times, where $t \in \{0, 1, 2, \ldots, K-1\}$ is the largest integer satisfying $\Delta_{K-t} \cdot t \leq K\epsilon$, and $\Delta_t^\epsilon = \max(\epsilon, \Delta_{K-t})$.

Note that Theorem 4 implies Theorem 1 because of the following reasons: 1) t defined in Theorem 4 is always at least $t(\epsilon, K)$ defined in (3); and 2) $\Delta_t^\epsilon \geq \Psi_t^\epsilon \geq \epsilon$.

Algorithm 1 is similar to the accept-reject types of algorithms in e.g. Bubeck et al. [2013]. The algorithm goes by rounds for $r = 1, 2, 3, \ldots$, and keeps at set of undecided arms $S_r \subseteq [n]$ at Round r.

Algorithm 1: ADAPTIVETOPK(n, ϵ, K, δ)

Input: n: number of arms; K and ϵ: parameters in ϵ-top-K arms; δ: error probability
Output: ϵ-top-K arms

1 Let r denote the current round, initialized to be 0. Let $S_r \subseteq [n]$ denote the set of candidate arms at round r. S_1 is initialized to be $[n]$. Set $A, B \leftarrow \emptyset$

2 $\Delta \leftarrow 2^{-r}$

3 while $2 \cdot \Delta \cdot (K - |A|) > \epsilon K$ **do**

4 $r \leftarrow r + 1$

5   Pull each arm in S_r by $\Delta^{-2} \ln \frac{2nr^2}{\delta}$ times, and let $\widetilde{\theta}_i^r$ be the empirical-mean

6   Define $\widetilde{\theta}_a(S_r)$ and $\widetilde{\theta}_b(S_r)$ be the $(K - |A| + 1)$th and $(K - |A|)$th largest empirical-means in S_r, and define
$$\widetilde{\Delta}_i(S_r) = \max\left(\widetilde{\theta}_i^r - \widetilde{\theta}_a(S_r), \widetilde{\theta}_b(S_r) - \widetilde{\theta}_i^r\right) \tag{7}$$

7   **while** $\max_{i \in S_r} \widetilde{\Delta}_i(S_r) > 2 \cdot \Delta$ **do**

8 $x \leftarrow \arg\max_{i \in S_r} \widetilde{\Delta}_i(S_r)$

9   **if** $\widetilde{\theta}_x^r > \widetilde{\theta}_a(S_r)$ **then**

10 $A \leftarrow A \cup \{x\}$

11   **else**

12 $B \leftarrow B \cup \{x\}$

13 $S_r \leftarrow S_r \backslash \{x\}$

14 $S_{r+1} \leftarrow S_r$

15 $\Delta \leftarrow 2^{-r}$

16 Set A' as the $(K - |A|)$ arms with the largest empirical-means in S_{r+1}

17 return $A \cup A'$

All other arms (in $[n] \setminus S_r$) are either accepted (in A) or rejected (in B). At each round, all undecided arms are pulled by equal number of times. This number is designed in a way such that the event $\mathcal{E}$, defined to be the empirical means of all arms within a small neighborhood of their true means, happens with probability $1 - \delta$ (See Definition 2 and Claim 1). Note that $\mathcal{E}$ is defined for all rounds and the length of the neighborhood becomes smaller as the algorithm proceeds. We are able to prove that when $\mathcal{E}$ happens, the algorithm returns the desired set of ϵ-top-K armsand has small query complexity.

To prove the correctness of the algorithm, we first show that when conditioning on $\mathcal{E}$, the algorithm always accepts a top-K arm in A (Lemma 3) and rejects a non-top-K arm in B (Lemma 4). The key observation here is that our algorithm never introduces any regret due to arms in A and B. We then use the key Lemma 5 to upper bound the regret that may be introduced due to the remaining arms. Once this upper bound is not more than ϵK (i.e. the total budget for regret), we can choose the remaining $(K - |A|)$ arms without further samplings. Details about this analysis can be found in Section 2.1.

We analyze of the query complexity of our algorithm in Section 2.2. We establish data-dependent bound by relating the number of pulls to each arms to both their Δ_i's and Δ_{K-t} (Lemma 6 and Lemma 7).

2.1 Correctness of Algorithm 1

We first define an event $\mathcal{E}$ which we will condition on in the rest of the analysis.

Definition 2 *Let $\mathcal{E}$ be the event that $|\widetilde{\theta}_i^r - \theta_i| < 2^{-r}$ for all $r \geq 1$ and $i \in S_r$.*

Claim 1 $\Pr[\mathcal{E}] \geq 1 - \delta$.

Proof: By Hoeffding's inequality, we can show that for any fixed r and i, $\mathbf{Pr}\left[|\widetilde{\theta}_i^r - \theta_i| \geq 2^{-r}\right] \leq 2(\frac{\delta}{2nr^2})^2 \leq \frac{\delta}{2nr^2}$. By a union bound,

$$\mathbf{Pr}[\neg\mathcal{E}] \leq \sum_{r=1}^{\infty}\sum_{i \in S_r}\mathbf{Pr}\left[|\widetilde{\theta}_i^r - \theta_i| \geq 2^{-r}\right] \leq \sum_{r=1}^{\infty}\frac{\delta}{2r^2} \leq \delta.$$

$\square$

The following lemma will be a very useful tool for our analysis.

Lemma 2 *Given $\mu_1 \geq \ldots \geq \mu_n$ and $\Delta > 0$, assuming that $|\widetilde{\mu}_i - \mu_i| \leq \Delta$ for all $i \in [n]$, and letting $y_1 \geq \ldots \geq y_n$ be the sorted version of $\widetilde{\mu}_1, \ldots, \widetilde{\mu}_n$, we have $|y_i - \mu_i| \leq \Delta$ for all $i \in [n]$.*

Proof: Suppose $y_i > \mu_i + \Delta$ for any $i \in [n]$, we must have $y_1 \geq \ldots \geq y_i > \mu_i + \Delta$. On the other hand, there can not be more than $i - 1$ numbers among $\widetilde{\mu}_1, \ldots, \widetilde{\mu}_n$ (the only candidates are $\widetilde{\mu}_1, \ldots, \widetilde{\mu}_{i-1}$) that are larger than $\mu_i + \Delta$. A contradition. We thus have $y_i \leq \mu_i + \Delta$ for all $i \in [n]$. Similarly, we can show that $y_i \geq \mu_i - \Delta$ for all $i \in [n]$. $\square$

We now prove that conditioned on $\mathcal{E}$, the algorithm always accepts a desired arm in A.

Lemma 3 *Conditioned on $\mathcal{E}$, during the run of Algorithm 1, $A \subseteq \{1, 2, \ldots, K\}$, that is, all arms in A are among the top-K arms.*

Proof: We prove by induction on the round r. The lemma holds trivially when $r = 0$ ($A = \emptyset$). Now fix a round $r \geq 1$, and let x be the arm that is added to A at Line 10 of Algorithm 1. By the induction hypothesis, assuming that before round r all arms in A are in $[K]$, our goal is to show $x \in [K]$.

By the *inner while* condition we have

$$\widetilde{\theta}_x^r - \widetilde{\theta}_a(S_r) > 2 \cdot 2^{-r}. \tag{8}$$

For any $m \in [K - |A| + 1, |S_r|]$, let j be the arm of the m-th largest true-mean in S_r, and j' be the arm of the m-th largest empirical-mean in S_r. Since $m \geq K - |A| + 1$, we must have $j \notin [K]$ and $\widetilde{\theta}_{j'}^r \leq \widetilde{\theta}_a(S_r)$. By Lemma 2 we also have $|\widetilde{\theta}_{j'}^r - \theta_j| < 2^{-r}$. We thus have

$$\theta_x > \widetilde{\theta}_x^r - 2^{-r} \overset{\text{by (8)}}{>} \widetilde{\theta}_a(S_r) + 2^{-r} > \widetilde{\theta}_{j'}^r + 2^{-r} > \theta_j.$$

That is, *at least $|S_r| - K + |A|$ arms in S_r have true-means smaller than arm x*. On the other hand, $|S_r| - K + |A|$ arms in S_r are not in $[K]$. We therefore conclude that x must be in $[K]$. $\square$

By symmetry, we also have the following lemma, stating that when $\mathcal{E}$ happens, the algorithm always rejects a non-top-K arm in B. We omit the proof because it is almost identical to the proof of Lemma 3.

Lemma 4 *Conditioning on $\mathcal{E}$, during the run of Algorithm 1, $B \subseteq \{K + 1, K + 2, \ldots, n\}$.*

Lemma 5 *Conditioned on $\mathcal{E}$, for all rounds r and $i \in S_r$, it holds that*

$$\widetilde{\theta}_i^r - \widetilde{\theta}_a(S_r) > \theta_i - \theta_{K+1} - 2 \cdot 2^{-r} \quad and \quad \widetilde{\theta}_b(S_r) - \widetilde{\theta}_i^r > \theta_K - \theta_i - 2 \cdot 2^{-r}.$$

Consequently, we have $\widetilde{\Delta}_i(S_r) \geq \Delta_i - 2 \cdot 2^{-r}$ for all rounds r and $i \in S_r$.

Proof: We look at a particular round r. Let j be the arm with $(K - |A| + 1)$-th largest true-mean in S_r. Since by Lemma 3 we have $A \subseteq [K]$, it holds that $j \geq K + 1$. By Lemma 2, we also have $|\widetilde{\theta}_a(S_r) - \theta_j| < 2^{-r}$. We therefore have for any $i \in S_r$

$$\widetilde{\theta}_i^r - \widetilde{\theta}_a(S_r) > \theta_i - \theta_j - 2 \cdot 2^{-r} \geq \theta_i - \theta_{K+1} - 2 \cdot 2^{-r}. \tag{9}$$

With a similar argument (by symmetry and using Lemma 4), we can show that

$$\widetilde{\theta}_b(S_r) - \widetilde{\theta}_i^r > \theta_K - \theta_i - 2 \cdot 2^{-r}. \tag{10}$$

Combining (9), (10) and the definitions of $\widetilde{\Delta}_i(S_r)$ and Δ_i, the lemma follows. $\qquad\square$

Now we are ready to prove the correctness of Theorem 4. By Lemma 3, all the arms that we add into the set A at Line 10 are in $[K]$. The rest of our job is to look at the arms in the set A'.

When the algorithm exits the *outer while* loop (at round $r = r^*$) and arrives at Line 16, we have by the condition of the *outer while* loop that

$$2 \cdot 2^{-r^*} \cdot (K - |A|) \leq \epsilon K. \tag{11}$$

Let $m = K - |A|$, and $C = [K] \backslash A = \{i_1, i_2, \ldots, i_m\}$ where $i_1 < i_2 < \ldots < i_m$. Let $\widetilde{\theta}_{j_1} \geq \widetilde{\theta}_{j_2} \geq \ldots \geq \widetilde{\theta}_{j_m}$ be the $(K - |A|)$ empirical-means of the arms that we pick at Line 16. Note that it is *not* necessary that $j_1 < \ldots < j_m$. By Lemma 2 and $\mathcal{E}$, for any $s \in [K - |A|]$, we have $|\widetilde{\theta}_{j_s} - \theta_{i_s}| \leq 2^{-r^*}$ and $|\widetilde{\theta}_{j_s} - \theta_{j_s}| \leq 2^{-r^*}$. By the triangle inequality, it holds that

$$|\theta_{j_s} - \theta_{i_s}| \leq 2 \cdot 2^{-r^*}. \tag{12}$$

We thus can bound the error introduced by arms in A' by

$$\sum_{i \in [K]} \theta_i - \sum_{i \in A \cup A'} \theta_i = \sum_{i \in C} \theta_i - \sum_{i \in A'} \theta_i \overset{by\ (12)}{\leq} 2 \cdot 2^{-r^*} \cdot (K - |A|) \overset{by\ (11)}{\leq} \epsilon K.$$

2.2 Query Complexity of Algorithm 1

Recall (in the statement of Theorem 4) that $t \in \{0, 1, 2, \ldots, K - 1\}$ is the largest integer satisfying

$$\Delta_{K-t} \cdot t \leq \epsilon K. \tag{13}$$

Lemma 6 *If the algorithm exits the* outer while *loop at round $r = r^*$, then we must have*

$$8 \cdot 2^{-r^*} \geq \Delta_{K-t}. \tag{14}$$

Proof: We show that once $2^{-r} < \Delta_{K-t}/4$, the algorithm will exit the outer while loop after executing round r. So any valid round r must satisfy $2^{-r} \geq \Delta_{K-t}/8$ and the lemma holds trivially.

To this end, assume now we are in round r and $2^{-r} < \Delta_{K-t}/4$, we have that for any $i \in S_r$ and $i \leq K - t$,

$$\begin{aligned}
\widetilde{\Delta}_i(S_r) \geq \widetilde{\theta}_i^r - \widetilde{\theta}_a(S_r) &> \theta_i - \theta_{K+1} - 2 \cdot 2^{-r} \quad \text{(Lemma 5)} \\
&= \Delta_i - 2 \cdot 2^{-r} \\
&\geq \Delta_{K-t} - 2 \cdot 2^{-r} \qquad \text{(since } i \leq K - t) \\
&> 2 \cdot 2^{-r}.
\end{aligned}$$

9

Thus the condition of the *inner while* loop is satisfied, which means that all arms i with $i \leq K - t$ will be added into A. Therefore we have $|A| \geq K - t$ when the algorithm exits the inner while loop. We then have

$$2 \cdot 2^{-r} \cdot (K - |A|) \leq 2 \cdot 2^{-r} \cdot t < \frac{1}{2} \Delta_{K-t} \cdot t \overset{by\ (13)}{\leq} \epsilon K / 2 \leq \epsilon K,$$

so the algorithm exits the outter loop. $\qquad\qquad\square$

Lemma 7 *For any arm i, let r_i be the round where arm i is removed from the candidate set if this ever happens; otherwise set $r_i = r^*$. We must have*

$$8 \cdot 2^{-r_i} \geq \Delta_i. \tag{15}$$

Proof: Suppose for contradiction that $8 \cdot 2^{-r_i} < \Delta_i$. By Lemma 5, we have

$$\widetilde{\Delta}_i(S_{r_i-1}) \geq \Delta_i - 2 \cdot 2^{-(r_i-1)} > 8 \cdot 2^{-r_i} - 2 \cdot 2^{-(r_i-1)} = 2 \cdot 2^{-(r_i-1)}.$$

This means that arm i would have been added either to A or B at or before round $(r_i - 1)$, which contradicts to the fact that $i \in S_{r_i}$. $\qquad\qquad\square$

With Lemma 6 and Lemma 7, we are ready to analyze the query complexity of the algorithm in Theorem 4. We can bound the number of pulls on each arm i by at most

$$\sum_{j=1}^{r_i} 2^{2j} \cdot \log(2nj^2/\delta) \leq O\left(\log(r_i \cdot n\delta^{-1}) \cdot 2^{2r_i}\right). \tag{16}$$

Now let us upper-bound the RHS of (16). First, if $i \in A$, then by (15) we know that $r_i \leq \log_2 \Delta_i^{-1} + O(1)$. Second, by (14) we have $r_i \leq r^* \leq \log_2 \Delta_{K-t}^{-1} + O(1)$. Third, since $2^{-r^*} \geq \epsilon/2$ (otherwise the algorithm will exit the *outer while* loop), we have $r_i \leq r^* \leq \log_2 \epsilon^{-1} + O(1)$. To summarize, we have $r_i \leq \log_2 \min\{\Delta_i^{-1}, \Delta_{K-t}^{-1}, \epsilon^{-1}\} + O(1) = \log_2 \min\{\Delta_i^{-1}, (\Delta_t^\epsilon)^{-1}\}$ (recall that $\Delta_t^\epsilon = \max\{\epsilon, \Delta_{K-t}\}$). We thus can upper-bound the RHS of (16) by

$$O\left(\left(\log\log(\Delta_t^\epsilon)^{-1} + \log n + \log\delta^{-1}\right) \cdot \min\{(\Delta_i)^{-2}, (\Delta_t^\epsilon)^{-2}\}\right).$$

The total cost is a summation over all n arms.

3   An Improved Algorithm for ϵ-top-K Arms

In this section, we present the improved algorithm for identifying the ϵ-top-K arms and prove that the algorithm succeeds with probability $1 - \delta$ with query complexity $O((\log \epsilon^{-1} + \log \delta^{-1})H^{(t,\epsilon)})$ (Theorem 6). This algorithm reduces the $\log n$ factor in the query complexity of Algorithm 1 to $\log \epsilon^{-1}$ and is substantially more complex than Algorithm 1.

The main procedure of the improved algorithm is described in Algorithm 5. For this algorithm, we that assume $K \leq n/2$. For the case where $K > n/2$, we can apply the same algorithm to identify the ϵ-bottom-$(n - K)$ arms and report the rest arms to be the ϵ-top-K arms. Similarly to Algorithm 1, the improved algorithm also goes by rounds and keeps a set A of accepted arms, a set B of rejected arms, and a set S of undecided arms. However, we can no longer guarantee that all the arms accepted in A and rejected in B are correctly classified – otherwise, we need to apply a union bound over all arms and this would incur an extra $\log n$ factor. To solve this problem, we have to allow a few number of mistakes. We now illustrate the high-level idea as follows.

Algorithm 2: EstKthArm$(S, K, \tau, \phi, \delta)$

Input: S: set of arms; K: top-K; τ: an relative error; δ: error probability; ϕ: an additive error
Output: an arm whose true-mean is close to the K-th largest true-mean

1 set $R_1 \leftarrow S, r \leftarrow 1$

2 set $\tau_1 \leftarrow \frac{\tau}{4}, \phi_1 \leftarrow \frac{\phi}{4}, \delta_1 \leftarrow \frac{\delta}{8}$

3 while $|R_r| > K$ **do**

4   for each $i \in R_r$, pull $\frac{8}{\phi_r^2} \ln(\frac{1}{\tau_r \delta_r \delta})$ times; let $\widetilde{\theta}_i^r$ be its empirical-mean

5   let R_{r+1} be the set of $\max\{K, \lceil |R_r|/2 \rceil\}$ arms that have the largest empirical-means among R_r

6   set $\tau_{r+1} \leftarrow 3\tau_r/4, \phi_{r+1} \leftarrow 3\phi_r/4, \delta_{r+1} \leftarrow \delta_r/2$

7 $r \leftarrow r + 1$

8 set $r^* \leftarrow r$

9 set $\widetilde{p}_1 \geq \widetilde{p}_2 \geq \ldots \geq \widetilde{p}_{|R_{r^*}|}$ be the sorted version of $\{\widetilde{\theta}_i^{r^*} \mid i \in R_{r^*}\}$

10 uniformly sample an arm from $\{i \in R_{r^*} \mid \widetilde{\theta}_i^{r^*} \leq \widetilde{p}_{(1-\tau/2)K}\}$ and output it

---

Given a set of n arms $\{\theta_1 \geq \theta_2 \geq \cdots \geq \theta_n\}$, if we pull every arm $c \cdot \Delta_{.8n}^{-2}(\log \epsilon^{-1} + \log \delta^{-1})$ times for some large enough constant c, and discard the $.1n$ arms with the lowest empirical means, it can be shown by standard probabilistic method that at most $\epsilon^2 K$ top-K arms may be mistakenly discarded with probability $1 - \delta$. Note that the constants $.8$ and $.1$ are arbitrary as long as $K/n < .8 < 1 - .1$. This procedure is described in Algorithm 2 and analyzed in Lemma 11. Similarly, if $.2n < K$ and we pull every arm $c \cdot \Delta_{.2n}^{-2}(\log \epsilon^{-1} + \log \delta^{-1})$ times for some large enough constant c, and accept the $.1n$ arms with the highest empirical means, with probability $1 - \delta$, at most $\epsilon^2 K$ non-top-K arms may be mistakenly accepted. This procedure is described in Algorithm 2 and analyzed in Lemma 12. Algorithm 5 uses these two subroutines to repeatedly accept and reject arms, and makes sure that with high probability, the total number of mistakenly accepted or rejected arms is at most $O(\epsilon^2 K)$ (Lemma 13). These mistakes lead to $O(\epsilon^2 K)$ total regret – negligible when compared to our ϵK budget. In this way, the improved algorithm keeps accepting and rejecting arms as Algorithm 1 does, while introducing negligible regret (while Algorithm 1 introduces none). The termination condition is also similar to Algorithm 1 in Line 12 of Algorithm 5 so that the query complexity is related to $H^{(t,\epsilon)}$ rather than $H^{(0,\epsilon)}$.

However, there is an extra termination condition and many extra efforts in the improved algorithm because of the few allowed mistakes. For our adaptive algorithm, in order to estimate $\Delta_{.8n}$ and $\Delta_{.2n}$ (and other gaps as the algorithm proceeds), we need to estimate θ_K, $\theta_{.8n}$ and $\theta_{.2n}$ with $O(\phi^{-2}(\log \epsilon^{-1} + \log \delta^{-1})$ pulls, where $\phi^{-2} = \Omega(\min\{\Delta_{.8n}, \Delta_{.2n}\})$. However, using these many pulls, we can only estimate the mean of an arm that is close to the target index, rather than with the exact index. This procedure is presented in Section 3.1 and Algorithm 2. We use this subroutine to estimate $\theta_{.8n}$ as θ^+, $\theta_{.2n}$ as θ^- in Algorithm 5, and use two estimations θ_K^+ and θ_K^- to sandwich θ_K. (The precise statement can be found in Lemma 14.) When θ_K^+ and θ_K^- are close to each other, we can use $\theta^- - \theta_K^-$ and $\theta_K^+ - \theta^+$ as estimations of $\Delta_{.2n}$ and $\Delta_{.8n}$; otherwise, it means that there is a big gap in the neighborhood of the K-th arm, and we can easily separate the top-K arms from the rest using the subprocedure EpsSplit described in Lemma 9 and quit the procedure (in Line 14 of the algorithm).

We now dive into the details of the improved algorithm. We start by introducing the useful subroutines.

3.1   Estimating the K-th Largest Arm

In the subsection we present an algorithm that try to find an arm whose true-mean is close to the K-th largest true-mean, which will be used as a subroutine in our improved algorithm for ϵ-top-K arms.

Theorem 5 *For a set of arms $S = \{\theta_1 \geq \ldots \geq \theta_{|S|}\}$, there is an algorithm, denoted by EstKthArm$(S, K, \tau, \phi, \delta)$, that outputs an arm i such that $\theta_i \in [\theta_K - \phi, \theta_{(1-\tau)K} + \phi]$ with probability at least $1 - \delta$, using $O\left(\frac{|S|}{\phi^2} \cdot (\log \tau^{-1} + \log \delta^{-1})\right)$ pulls in total.*

We described the algorithm in Algorithm 2. In the high level, the algorithm works in rounds, and in each round it tries to find the top half arms in the current set, and discard the rest. We continue until there are at most K arms left, and then we choose the output arm randomly from those with the lowest empirical-means in the remaining arms. We are going to prove the following theorem.

3.1.1   Correctness of Algorithm 2

The following lemma is the key to the proof of correctness.

Lemma 8 *With probability at least $1 - \delta/4$, we have that*

$$|\{i \in R_{r^*} \mid \theta_i < \theta_K - \phi\}| \leq \tau \delta K/4.$$

Proof: We first define a few notations.

- $H_r = \{i \in S \mid \theta_i \geq \theta_K - \sum_{\ell \in [r]} \phi_\ell\}$.

- $L_r = S \backslash H_r$.

- $k_r = (1 - \sum_{\ell \in [r]} \tau_\ell \delta/4) K$.

- For any $i \in [n]$ and round r, let $X_i^r = \mathbf{1}\{|\widetilde{\theta}_i^r - \theta_i| \geq \phi_r/2\}$ where $\widetilde{\theta}_i^r$ is the empirical-mean of arm i at round r (after been pulled by $\frac{8}{\phi_r^2} \ln(\frac{1}{\tau_r \delta_r \delta})$ times at Line 4).

- $R_r' \subseteq R_r$: the top k_{r-1} arms in R_r with the largest true-means.

- $A_r = \{i \in R_r' \mid X_i = 0\}$. $A_r \subseteq R_r' \subseteq R_r$.

- $C_r = \{i \in L_r \cap R_r \mid X_i = 1\}$. $C_r \subseteq R_r$.

We define the following event. Intuitively, it tells that most of the arms we put in R_r for the next round processing fall into the set H_r of high true-means.

$$\mathcal{E}_r (r \geq 2) : \ |H_{r-1} \cap R_r| \leq k_{r-1},$$

and we define $\mathcal{E}_1$ to be an always true event. We will prove by induction the following inequality.

$$\mathbf{Pr}[\mathcal{E}_{r+1} \mid \neg \mathcal{E}_r] \leq \delta_r \ \text{ for each } r \geq 1. \tag{17}$$

We focus a particular round $r \geq 1$. Define event $\mathcal{E}_A : |A_r| \leq k_r$, and event $\mathcal{E}_C : |C_r| \leq |R_{r+1}| - k_r$.

Claim 2 $\mathbf{Pr}[\mathcal{E}_A \mid \neg \mathcal{E}_r] \leq \delta_r/4.$

Proof: By a Hoeffding's inequality, we have for each $i \in R_r$, we have $\mathbf{Pr}[X_i^r = 1] \leq \tau_r \delta_r \delta/16$. We bound the probability the $\mathcal{E}_A$ happens by a Markov's inequality.

$$\mathbf{Pr}[\mathcal{E}_A \mid \neg \mathcal{E}_r] = \mathbf{Pr}[|A_r| \leq k_r \mid \neg \mathcal{E}_r] = \mathbf{Pr}[|R_r'| - |A_r| \geq k_{r-1} - k_r \mid \neg \mathcal{E}_r]$$

$$\leq \frac{\mathbf{E}\left[\sum_{i \in R_r'} X_i \mid \neg \mathcal{E}_r\right]}{k_{r-1} - k_r} = \frac{\mathbf{E}\left[\sum_{i \in R_r'} X_i\right]}{k_{r-1} - k_r}$$

$$\leq \frac{\tau_r \delta_r \delta k_{r-1}/16}{\tau_r \delta K/4} \leq \frac{\tau_r \delta_r \delta K/16}{\tau_r \delta K/4}$$

$$= \delta_r/4.$$

12

$\square$

Claim 3 $\mathbf{Pr}[\mathcal{E}_C \mid \neg\mathcal{E}_A, \neg\mathcal{E}_r] \le 3\delta_r/4.$

Proof:

$$\mathbf{Pr}[\mathcal{E}_C \mid \neg\mathcal{E}_A, \neg\mathcal{E}_r] = \mathbf{Pr}[|C_r| \le |R_{r+1}| - k_r \mid \neg\mathcal{E}_A, \neg\mathcal{E}_r]$$

$$\le \frac{\mathbf{E}[|C_r| \mid \neg\mathcal{E}_A, \neg\mathcal{E}_r]}{|R_{r+1}| - k_r} \qquad \text{(Markov's inequality)} \tag{18}$$

$$\le \frac{(|R_r| - k_r) \cdot \tau_r \delta \delta_r/16}{|R_{r+1}| - k_r} \tag{19}$$

$$\le \frac{(2|R_{r+1}| - k_r) \cdot \tau_r \delta \delta_r/16}{|R_{r+1}| - k_r} \tag{20}$$

$$\le 3\delta_r/4, \tag{21}$$

where (18) to (19) is due to the fact that conditioned on $\neg\mathcal{E}_r$, we have

$$|C_r| \le |L_r \cap R_r| \le |L_{r-1} \cap R_r| = |R_r| - |H_{r-1} \cap R_r| \overset{\neg\mathcal{E}_r}{\le} |R_r| - k_{r-1} \le |R_r| - k_r.$$

And (20) to (21) is due to the following: If $|R_{r+1}| \ge 2K$, then since $k_r \le K$ we have $\frac{2|R_{r+1}| - k_r}{|R_{r+1}| - k_r} \le 3$. Otherwise if $|R_{r+1}| < 2K$, by the definition of k_r, we have

$$|R_{r+1}| - k_r > K - k_r = \sum_{i \in [r-1]} \tau_i \delta/4 \ge \tau_r \delta K/3.$$

On the other hand, we have $2|R_{r+1}| - k_r \le 4K$. We thus have $(21) \le 3\delta_r/4$. $\square$

Claim 4 *Conditioned on $\mathcal{E}_C, \neg\mathcal{E}_A, \neg\mathcal{E}_r$, we have $|H_r \cap R_{r+1}| \ge k_r$, or, $\mathbf{Pr}[\mathcal{E}_{r+1} \mid \mathcal{E}_C, \neg\mathcal{E}_A, \neg\mathcal{E}_r] = 0$.*

Proof: First, conditioned on $\neg\mathcal{E}_r$, we have

$$A_r \subseteq R'_r \subseteq H_{r-1}. \tag{22}$$

We prove the claim by analyzing two cases.

1. $((L_r \cap R_r)\backslash C_r) \cap R_{r+1} = \emptyset$. We thus have $(L_r \cap R_{r+1}) \subseteq C_r$, which implies

$$|H_r \cap R_{r+1}| \ge |R_{r+1}| - |L_r \cap R_{r+1}| \ge |R_{r+1}| - |C_r| \overset{\mathcal{E}_C}{\ge} k_r.$$

2. $((L_r \cap R_r)\backslash C_r) \cap R_{r+1} \ne \emptyset$. We can show that

$$A_r \subseteq R_{r+1} \tag{23}$$

Indeed, for any $j \in A_r (\subseteq H_{r-1}$ by (22)) and any $i \in (L_r \cap R_r)\backslash C_r$, we have $\widetilde{\theta}_j > \widetilde{\theta}_i$, since $\theta_j \ge \theta_K - \sum_{\ell \in [r-1]} \phi_\ell$, while $\theta_i < \theta_K - \sum_{\ell \in [r]} \phi_\ell$ (by the definition of L_r). Thus conditioned on $\neg\mathcal{E}_A$, (22) and (23) we have

$$|H_r \cap R_{r+1}| \ge |H_{r-1} \cap R_{r+1}| \ge |A_r| \ge k_r.$$

$\square$

Now we try to prove (17).

$$\mathbf{Pr}[\mathcal{E}_{r+1} \mid \neg\mathcal{E}_r] = \mathbf{Pr}[\mathcal{E}_A \mid \neg\mathcal{E}_r] + \mathbf{Pr}[\mathcal{E}_C \mid \neg\mathcal{E}_A, \neg\mathcal{E}_r] + \mathbf{Pr}[\mathcal{E}_{r+1} \mid \neg\mathcal{E}_A, \neg\mathcal{E}_r]$$

13

$$\leq \delta/4 + 3\delta/4 + 0 \qquad \text{(Claim 2, Claim 3 and Claim 4)}$$
$$= \delta.$$

Using (17) and summing over all $\ell \in [r-1]$, we have $\mathbf{Pr}[\mathcal{E}_r] \leq \sum_{\ell \in [r-1]} \delta_i < \delta/4$ for any $r \geq 2$. In other words, with probability at least $1 - \delta/4$, we have for any $r \leq r^*$, $|H_{r-1} \cap R_r| \geq k_{r-1}$, or $|L_{r-1} \cap R_r| \leq K - k_{r-1} \leq 1 - (1 - \tau\delta K/4) = \tau\delta K/4$, which gives the lemma. $\qquad \square$

Now we are ready to prove the correctness of Theorem 5. Let

- $P = \{i \in R_{r^*} \mid \widetilde{\theta}_i^{r^*} > \widetilde{p}_{(1-\tau/2)K}\}$.

- $Q = R_{r^*} \backslash P$.

- $L^* = \{i \in R_{r^*} \mid \theta_i < \theta_K - \phi\}$.

- For each $i \in R_{r^*}$, let $Y_i = \mathbf{1}\{|\widetilde{\theta}_i^{r^*} - \theta_i| > \phi/2\}$.

Claim 5 $\mathbf{Pr}[\theta_a \geq \theta_K - \phi] \geq 1 - 3\delta/4$.

Proof: At Line 10 of Algorithm 2, we randomly sampled an arm a from Q. Since $|Q| \geq \tau K$ and $\mathbf{Pr}[|L^*| < \tau\delta K/4] > 1 - \delta/4$ (by Lemma 8), we have

$$\mathbf{Pr}[a \in L^*] \leq \mathbf{Pr}[|L^*| \geq \tau\delta K/4] + \mathbf{Pr}[a \in L^* \mid |L^*| < \tau\delta K/4] \leq \delta/4 + \frac{\tau\delta K/4}{\tau K/2} = 3\delta/4. \qquad (24)$$

$\qquad \square$

Claim 6 $\mathbf{Pr}[\theta_a \leq \theta_{(1-\tau)K} + \phi] \geq 1 - \delta/4$.

Proof: We first show that if $Y_a = 0$ and $\sum_{i \in P} Y_i \leq \tau K/2$, then $\theta_a \leq \theta_{(1-\tau)K} + \phi$. Let $P_0 = \{i \in P \mid Y_i = 0\}$. By definition of P and the assumption that $\sum_{i \in P} Y_i \leq \tau K/2$, we have $|P_0| \geq |P| - \tau K/2 \geq (1-\tau)K$. Let b be the arm in P_0 that has the minimum true-mean, then we must have

$$\theta_b \leq \theta_{|P_0|} \leq \theta_{(1-\tau)K}. \qquad (25)$$

Since $a \in Q$ and $b \in P$, we have $\widetilde{\theta}_a \leq \widetilde{\theta}_b$, which, together with the facts that $Y_a = Y_b = 0$ and (25), gives

$$\theta_a \leq \theta_b + \phi \leq \theta_{(1-\tau)K} + \phi.$$

We now bound the probabilities that the two conditions hold. By a Hoeffding's inequality, we have $\mathbf{Pr}[Y_i = 1] \leq \tau\delta/16$ for any $i \in R_{r^*}$. By a Markov's inequality and the fact that $P \subseteq R_{r^*}$ (by definition), we have

$$\mathbf{Pr}\left[\sum_{i \in P} Y_i > \tau K/2\right] \leq \mathbf{Pr}\left[\sum_{i \in R_{r^*}} Y_i > \tau K/2\right] \leq \frac{\mathbf{E}\left[\sum_{i \in R_{r^*}} Y_i\right]}{\tau K/2} \leq \frac{\tau\delta K/16}{\tau K/2} = \delta/8.$$

We thus have $\mathbf{Pr}\left[(Y_a = 0) \wedge (\sum_{i \in P} Y_i \leq \tau K/2)\right] \geq 1 - \delta/16 - \delta/8 \geq 1 - \delta/4$.

The correctness of Theorem 5 immediately follows from Claim 6 and Claim 5. $\qquad \square$

3.1.2 Complexity Algorithm 2

We can bound the total number of pulls of Algorithm 2 by simply summing up the number of pulls at each round.

Algorithm 3: $\text{ELIM}(S, K, \gamma, \phi, \delta)$

Input: S: set of arms; K: top-K; γ: fraction of arms; δ: error probability; ϕ: an additive error

Output: Set of arms T with $|T| = \frac{|S|}{10}$ such that at most γK arms in T are in the top-K arms in S

1 For each arm i in S, pull $\frac{c}{\phi^2} \cdot (\log \gamma^{-1} + \log \delta^{-1})$ for some large enough constant c. Let $\widetilde{\theta}_i$ be the empirical-mean of the i-th arm.

2 **return** $|T|$ *arms in S with the smallest empirical-means.*

$$
O\left(\sum_{r=1}^{\log |S|/K} \frac{|R_r|}{\phi_r^2} \log \frac{1}{\tau_r \delta_r \delta} \right) = O\left(\sum_{r=1}^{\infty} \frac{2^{-r}|S|}{(3/4)^{2r}\phi^2} \cdot \left(\log \frac{1}{(3/4)^r \tau} + \log \frac{1}{(1/2)^r \delta} + \log \frac{1}{\delta} \right) \right)
$$

$$
= O\left(\frac{|S|}{\phi^2} \sum_{r=1}^{\infty} \left(\frac{8}{9} \right)^r \cdot \left(r + \log \frac{1}{\tau} + \log \frac{1}{\delta} \right) \right)
$$

$$
= O\left(\frac{|S|}{\phi^2} \cdot \left(\log \frac{1}{\tau} + \log \frac{1}{\delta} \right) \right).
$$

The last equality follows from the fact that $\sum_{r=1}^{\infty}(8/9)^r \cdot r = O(1)$.

Lemma 8 also implies the following lemma.

Lemma 9 *For a set of arms $S = \{\theta_1 \geq \ldots \geq \theta_{|S|}\}$ such that $\theta_{(1-\tau)K} - \theta_{(1+\tau)K+1} \geq \phi$, there is an algorithm, denoted by $\text{EPSSPLIT}(S, K, \tau, \phi, \delta)$, that computes (2τ)-top-K correctly with probability at least $1 - \delta$, using $O\left(\frac{|S|}{\phi^2} \cdot (\log \tau^{-1} + \log \delta^{-1}) \right)$ pulls in total.*

Proof: To prove the lemma, we just need to replace "K" in Algorithm 2 to be "$(1-\tau)K$". By Lemma 8 we have with probability $1 - \delta$ that

$$
|\{i \in R_{r^*} \mid \theta_i < \theta_{(1-\tau)K} - \phi\}| \leq \tau\delta(1-\tau)K \leq \tau\delta K.
$$

Since $\theta_{(1-\tau)K} - \theta_{(1+\tau)K+1} \geq \phi$ we have

$$
|\{i \in R_{r^*} \mid \theta_i < \theta_{(1+\tau)K+1}\}| \leq \tau\delta K.
$$

Consequently,

$$
|\{i \in R_{r^*} \mid \theta_i < \theta_K\}| \leq \tau K + \tau\delta K \leq 2\tau K. \tag{26}
$$

Therefore we can just choose all arms in R_{r^*}, together with $K - |R_{r^*}|$ arbitrary arms in the rest $n - |R_{r^*}|$ arms. By (26) the total average error is bounded by $2\tau K/K = 2\tau$. □

3.2 The Improved Algorithm

In this section, we introduce an improved algorithm that removes the $\log(n)$-factor in the sample complexity.

We first introduce a few more subroutines (Lemma 10, Lemma 11, and Lemma 12) that will be useful for our improved algorithm.

Lemma 10 *[Zhou et al., 2014] For a set of arms $S = \{\theta_1 \geq \ldots \geq \theta_{|S|}\}$, there is an algorithm, denoted by $\text{OPTMAI}(S, K, \epsilon, \delta)$, that computes ϵ-top-K arms with probability $1 - \delta$, with $O\left(\frac{|S|}{\epsilon^2} \log(1/\delta) \right)$ pulls.*

The following two lemmas show how to find a constant fraction of arms in the set of top-K arms and a constant fraction of arms outside the set of top-K arms respectively.

Algorithm 4: REVERSEELIM$(S, K, \gamma, \phi, \delta)$

Input: S: set of arms; K: top-K; γ: fraction of arms; δ: error probability; ϕ: an additive error

Output: Set of arms T with $|T| = \frac{|S|}{10}$ such that at most γK arms in T are in the top-$(|S| - K)$ arms in S

1 For each arm i in S, pull $\frac{c}{\phi^2} \cdot (\log \gamma^{-1} + \log \delta^{-1})$ for some large enough constant c. Let $\widetilde{\theta}_i$ be the empirical-mean of the i-th arm.

2 return $|T|$ *arms in S with the largest empirical-means.*

Lemma 11 *For a set of arms $S = \{\theta_1 \geq \ldots \geq \theta_{|S|}\}$ such that $\theta_K - \theta_{\lfloor \frac{|S|+K}{2} \rfloor} \geq \phi$, $K \leq \frac{2}{3}|S|$, there is an algorithm, denoted by $\mathrm{ELIM}(S, K, \gamma, \phi, \delta)$ in Algorithm 2, that computes $T \subseteq S, |T| = \frac{|S|}{10}$ successfully with probability $1 - \delta$ using $O\left(\frac{|S|}{\phi^2} \cdot (\log \gamma^{-1} + \log \delta^{-1})\right)$ pulls in total, such that at most γK arms in T are in the top-K arms in S.*

Proof: Let $\widetilde{\theta}_i$ be the empirical-mean of arm i after being pulled by $c \cdot \frac{1}{\phi^2} \log \frac{1}{\gamma\delta}$ times for a sufficiently large constant c. By a Hoeffding's inequality, we have $\mathbf{Pr}\left[|\widetilde{\theta}_i - \theta_i| \geq \phi/2\right] \leq \gamma\delta/2$. Let $X_i = \mathbf{1}\{|\widetilde{\theta}_i - \theta_i| \geq \phi/2\}$, and thus $\mathbf{E}[X_i] \leq \gamma\delta/2$. Let $X = \sum_{i \in [K]} X_i$; we have $\mathbf{E}[X] \leq \gamma\delta K/2$. By a Markov's inequality, we have that with probability at least $1 - \delta/2$, $X \leq \gamma K$. Consequently, with probability at least $1 - \delta/2$, there are at most γK arms $i \in [K]$ with $\widetilde{\theta}_i \leq \theta_i - \phi/2 \leq \theta_K - \phi/2$.

Let $L = \{\frac{|S|+K}{2} + 1, \ldots, |S|\}$. Since $K \leq \frac{2}{3}|S|$, we have $|L| \geq \frac{|S|}{6}$. Using similar argument we can show that with probability $1 - \delta/2$, there are at least $\frac{|S|}{10}$ arms $i \in L$ with $\widetilde{\theta}_i \leq \theta_i + \phi/2 \leq \theta_{\lfloor \frac{|S|+K}{2} \rfloor} + \phi/2 \leq \theta_K - \phi/2$ (since $\theta_K - \theta_{\lfloor \frac{|S|+K}{2} \rfloor} \geq \phi$).

Therefore, if we choose T to be the $\frac{|S|}{10}$ arms with the smallest empirical-means, then with probability at least $1 - \delta$, at most γK arms in T are in the top-K arms in S. □

Lemma 12 *For a set of arms $S = \{\theta_1 \geq \ldots \geq \theta_{|S|}\}$ such that $\theta_{\frac{K}{2}} - \theta_K \geq \phi$, $K \geq \frac{|S|}{3}$, there is an algorithm, denoted by $\mathrm{REVERSEELIM}(S, K, \gamma, \phi, \delta)$ in Algorithm 2, that computes $T \subseteq S, |T| = \frac{|S|}{10}$ successfully with probability $1 - \delta$ using $O\left(\frac{|S|}{\phi^2} \cdot (\log \gamma^{-1} + \log \delta^{-1})\right)$ pulls in total, such that at most γK arms in T are in the bottom-$(|S| - K)$ arms in S.*

By symmetry, the proof to Lemma 12 is basically the same as that for Lemma 11,

Now we are ready to show our main result.

Theorem 6 *Algorithm 5 computes ϵ-top-K arms with probability at least $1 - \delta$, and pulls the arms at most*

$$O\left(\sum_{j \in [n]} \min\left\{(\Delta_j)^{-2}, (\Psi_t^\epsilon)^{-2}\right\}\left(\log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\right) \tag{27}$$

times, where t and Ψ_t^ϵ are defined in (3) and (5) respectively.

It is worthwhile to note that the proposed algorithm is mainly for the theoretical interest and is rather complicated in terms of implementation. Thus, we omit the empirical study of this algorithm in the experimental section.

In the rest of this section we prove Theorem 6 by showing the correctness of Algorithm 5 and the analyzing its query complexity.

3.2.1 Correctness of Algorithm 5

Define $\mathcal{E}_1$ to be the event that all calls to the subroutine ESTKTHARM succeed.

Algorithm 5: IMPROVEDTOPK(n, K, ϵ, δ)

Input: n: number of arms; K and ϵ: see the definition of ϵ-top-K arms; δ: error probability. Assume $K \leq n/2$

Output: ϵ-top-K arms

1 set $S \leftarrow [n]$, $r \leftarrow 1$, $r_\phi \leftarrow 1$

2 set $A, B \leftarrow \emptyset$

3 set $K_L \leftarrow (1 - \epsilon^2)K$, $K_R \leftarrow (1 + \epsilon^2)K + 1$

4 while $S \neq \emptyset$ **do**

5 $r_\phi \leftarrow r_\phi - 1$

6   **repeat**

7 $r_\phi \leftarrow r_\phi + 1$; $\phi \leftarrow 2^{-r_\phi}$

8 $\theta_K^+ \leftarrow \text{ESTKTHARM}\left(S, K_R - |A|, \frac{\epsilon^2}{100r^2}, \phi, \frac{\delta}{100(r+r_\phi)^2}\right)$;

         $\theta_K^- \leftarrow \text{ESTKTHARM}\left(S, K_L - |A|, \frac{\epsilon^2}{100r^2}, \phi, \frac{\delta}{100(r+r_\phi)^2}\right)$

9 $\theta^+ \leftarrow \text{ESTKTHARM}\left(S, \frac{|S|+K-|A|}{2}, \frac{\epsilon^2}{100r^2}, \phi, \frac{\delta}{100(r+r_\phi)^2}\right)$;

         $\theta^- \leftarrow \text{ESTKTHARM}\left(S, \frac{K-|A|}{2}, \frac{\epsilon^2}{100r^2}, \phi, \frac{\delta}{100(r+r_\phi)^2}\right)$

10   **until** $(10(K - |A|)\phi < K\epsilon)$ *or* $(\theta_K^- - \theta_K^+ > 3\phi)$ *or* $\left(K - |A| \leq \frac{|S|}{2}\right) \wedge (\theta_K^+ - \theta^+ > 3\phi)$ *or*

     $\left(K - |A| > \frac{|S|}{2}\right) \wedge (\theta_K^+ - \theta^+ > 3\phi) \wedge (\theta^- - \theta_K^- > 3\phi)$

11   **if** $(10(K - |A|)\phi < K\epsilon)$ **then**

12   **return** $\text{OPTMAI}\left(S, K - |A|, \phi, \frac{\delta}{100}\right) \cup A$

13   **if** $(\theta_K^- - \theta_K^+ > 3\phi)$ **then**

14   **return** $\text{EPSSPLIT}\left(S, K - |A|, \frac{K_R - K_L}{K - |A|}, \phi, \frac{\delta}{100}\right) \cup A$

15 $U \leftarrow \text{ELIM}\left(S, K - |A|, \frac{\epsilon^2}{100r^2}, \phi, \frac{\delta}{100r^2}\right)$

16   **if** $(K - |A|) > \frac{|S|}{2}$ **then**

17 $V \leftarrow \text{REVERSEELIM}\left(S, K - |A|, \frac{\epsilon^2}{100r^2}, \phi, \frac{\delta}{100r^2}\right)$

18 $r \leftarrow r + 1$; $S \leftarrow S \backslash (U \cup V)$; $A \leftarrow A \cup V$; $B \leftarrow B \cup U$

19 return A.

Claim 7 $\mathbf{Pr}[\mathcal{E}_1] \geq 1 - \delta/10$.

Proof: Note that $(r + r_\phi)$ increases every time we call the four ESTKTHARM's at Line 8 and Line 9. Therefore by Theorem 5 we can bound the error probability of all calls to ESTKTHARM by $4 \cdot \sum_{z=1}^\infty \frac{\delta}{100z^2} \leq \frac{\delta}{10}$. $\qquad\square$

Define $\mathcal{E}_2$ to be the event that all calls to the subroutines ELIM and REVERSEELIM succeed. Since r increases every time we call the two subroutines, by similar arguments we have:

Claim 8 $\mathbf{Pr}[\mathcal{E}_2] \geq 1 - \delta/20$.

Define $\mathcal{E} = \mathcal{E}_1 \cup \mathcal{E}_2$; we thus have $\mathbf{Pr}[\mathcal{E}] \geq 1 - \delta/5$.

We next show that the misclassified arms are negligible during the run of the algorithm.

Lemma 13 *Conditioned on $\mathcal{E}$, suppose that the conditions of Lemma 11 and Lemma 12 always hold during the run of the Algorithm 5, then we always have*

1. The number of non-top-K arms in A, denoted by ι_A, is no more than $\frac{\epsilon^2 K}{40}$.

2. *The number of top-K arms in B, denoted by ι_B, is no more than $\frac{\epsilon^2 K}{40}$.*

Proof: By Lemma 11 we have $\iota_A \leq \sum_{r=1}^{\infty} \frac{\epsilon^2}{100r^2} \cdot K \leq \frac{\epsilon^2 K}{40}$. Similarly, by Lemma 12 we have $\iota_B \leq \sum_{r=1}^{\infty} \frac{\epsilon^2}{100r^2} \cdot K \leq \frac{\epsilon^2 K}{40}$. □

We now show that the conditions of Lemma 11 and Lemma 12 do hold. We first introducing a lemma showing that θ_K is sandwiched by θ_K^+ and θ_K^- during the run of Algorithm 5.

Lemma 14 *Conditioned on $\mathcal{E}$, at any point of the run of Algorithm 5, we have*

$$\theta_K^- + \phi \geq \theta_{K - \frac{\epsilon^2 K}{10}} \geq \theta_K \geq \theta_{K+1} \geq \theta_{K+1+\frac{\epsilon^2 K}{10}} \geq \theta_K^+ - \phi.$$

Proof: We consider a particular around r. The difference between θ_K^+ and the fixed value θ_{K_R} is generated by calling of the subroutine EstKthArm at Line 8, which can be bounded by the the error introduced when selecting the $(K_R - |A|)$-th largest arm in S plus $\max\{\iota_A, \iota_B\}$. By Theorem 5, Lemma 13 and $\mathcal{E}$, we have $\theta_K^+ \leq \theta_{K_R - \tau_R} + \epsilon_R$, where $\epsilon_R \leq \phi$, and

$$\tau_R \leq \tau \cdot (K_R - |A|) + \max\{\iota_A, \iota_B\} \leq \frac{\epsilon^2}{100r^2} \cdot (1 + \epsilon^2) K + \frac{\epsilon^2 K}{40} \leq \frac{\epsilon^2 K}{20} \leq K_R - K - \frac{\epsilon^2 K}{10}.$$

We thus have $\theta_K^+ \leq \theta_{K+1+\frac{\epsilon^2}{10K}} + \phi$.

Similarly, by Theorem 5, Lemma 13 and $\mathcal{E}$, we have $\theta_K^- \geq \theta_{K_L + \tau_L} - \epsilon_L$, where $\epsilon_L \leq \phi$, and $\tau_L \leq \max\{\iota_A, \iota_B\} \leq \frac{\epsilon^2 K}{40} \leq K - K_L - \frac{\epsilon^2 K}{10}$. We thus have $\theta_K^- \geq \theta_{K - \frac{\epsilon^2 K}{10}} - \phi$. □

We have the following immediate corollary.

Corollary 1 *If $\theta_K^- - \theta_K^+ \leq 3\phi$, then $\theta_K, \theta_{K+1} \in [\theta_K^+ - \phi, \theta_K^+ + 4\phi]$ and $\theta_K, \theta_{K+1} \in [\theta_K^- - 4\phi, \theta_K^- + \phi]$.*

In the following, for convenience, we always use $S[i]$ to denote the true-mean of the i-th arm (sorted decreasingly) in the current set S during the run of Algorithm 5, and use $S[i..j]$ to denote the set of true-means of the $(i, i+1, \ldots, j)$-th arms in S. Let $\widetilde{K} \triangleq K - |A|$. We call $S[1..\widetilde{K}]$ the *head* of S, and $S[\widetilde{K}+1..|S|]$ the *tail* of S. The following claim follows directly from Lemma 13 and Lemma 14.

Claim 9 *At any point during the run of Algorithm 5, it holds that $\theta_{K+\frac{\epsilon^2 K}{40}} \leq S[\widetilde{K}] \leq \theta_{K-\frac{\epsilon^2 K}{40}}$, and consequently $\theta_K^- + \phi \geq S[\widetilde{K}] \geq \theta_K^+ - \phi$.*

Lemma 15 *Conditioned on $\mathcal{E}$, the conditions of Lemma 11 always hold during the run of Algorithm 5, that is, we have*

$$S[\widetilde{K}] - S\left[\frac{|S|+\widetilde{K}}{2}\right] \geq \phi \quad and \quad \widetilde{K} \leq \frac{2}{3}|S|. \tag{28}$$

Proof: For the first item of (28), by Theorem 5 we have

$$S\left[\frac{|S|+\widetilde{K}}{2}\right] - \phi \leq \theta^+, \tag{29}$$

which together with $\theta_K^+ - \theta^+ > 3\phi$ (testing condition at Line 10) and $S[\widetilde{K}] \geq \theta_K^+ - \phi$ (Claim 9) give $S[\widetilde{K}] - S\left[\frac{|S|+\widetilde{K}}{2}\right] \geq \phi$ (by two triangle inequalities).

For the second item of (28), note that if $\widetilde{K} \leq \frac{|S|}{2}$, then $\widetilde{K} \leq \frac{2}{3}|S|$ holds directly. We thus consider the case $\widetilde{K} > \frac{|S|}{2}$. The observation is that at the beginning, before the first call to ReverseElim, we must have called Elim a number of times; each time we remove $\frac{|S|}{10}$ arms, most of which are from the tail of S. After the first time when $\widetilde{K} > \frac{|S|}{2}$, we call both Elim and ReverseElim, with the intention of removing $\frac{|S|}{10}$ arms from the tail and the head respectively. It may happen that after calling both Elim and ReverseElim a few times, we again have $\widetilde{K} \leq \frac{|S|}{2}$, at which point we will again only call Elim until

18

the point that we are back to the case that $\widetilde{K} > \frac{|S|}{2}$ and then we will call both ELIM and REVERSEELIM. Basically, the two patterns 'call ELIM only' and 'call both ELIM and REVERSEELIM' interleave, and we only need to consider one run of this interleaved sequence.

By Lemma 11 and Lemma 12 we know that at most $\frac{\epsilon^2}{100r^2}\widetilde{K}$ arms in the head of S will be removed when calling ELIM, and at most $\frac{\epsilon^2}{100r^2}\widetilde{K}$ arms in the tail of S will be removed when calling REVERSEELIM. Therefore, the worst case for causing the imbalance between $\widetilde{K}$ and $(|S| - \widetilde{K})$ is that each call of ELIM removes $\frac{|S|}{10}$ from the tail of S, and each call of REVERSEELIM removes $\left(\frac{|S|}{10} - \frac{\epsilon^2}{100r^2}\widetilde{K}\right)$ arms from the head of S and $\frac{\epsilon^2}{100r^2}\widetilde{K}$ arms from the tail of S. Note that the number of calls of ELIM and REVERSEELIM is bounded by $O(\log \frac{1}{\epsilon})$ since when reaching Line 15 we always have $|S| \geq \Omega(\phi) \geq \Omega(\epsilon)$. We thus have

$$\frac{|S| - \widetilde{K}}{|S|} \geq \left(\frac{1}{2} - 0.1\right) \cdot \prod_{r=1}^{O(\log 1/\epsilon)} \left(1 - \frac{\epsilon^2}{100r^2}\right) \geq \frac{1}{3},$$

which implies $\widetilde{K} \leq \frac{2}{3}|S|$. $\qquad\square$

Lemma 16 *Conditioned on $\mathcal{E}$, the conditions of Lemma 12 always hold during the run of Algorithm 5, that is, we have*

$$S\left[\frac{\tilde{K}}{2}\right] - S[\widetilde{K}] \geq \phi \quad and \quad \widetilde{K} \geq \frac{|S|}{3}. \tag{30}$$

Proof: In Algorithm 5, when calling REVERSEELIM, we always have

$$\theta^- - \theta_K^- > 3\phi \quad and \quad \widetilde{K} > \frac{|S|}{2}. \tag{31}$$

Thus $\widetilde{K} \geq \frac{|S|}{3}$ follows directly. By Theorem 5 we have

$$S\left[\frac{\tilde{K}}{2}\right] + \phi \geq \theta^-, \tag{32}$$

which together with $\theta^- - \theta_K^- > 3\phi$ (first item of (31)) and $S[\widetilde{K}] \leq \theta_K^- + \phi$ (Claim 9), gives $S\left[\frac{\tilde{K}}{2}\right] - S[\widetilde{K}] \geq \phi$ (by two triangle inequalities). $\qquad\square$

Lemma 15, Lemma 16 and Lemma 13 give the following corollary.

Corollary 2 *Conditioned on $\mathcal{E}$, during the run of Algorithm 5 we always have (1) the number of non-top-K arms in A is no more than $\iota_A = \frac{\epsilon^2 K}{40}$, and (2) the number of top-K arms in B is no more than $\iota_B = \frac{\epsilon^2 K}{40}$.*

We now consider the boundary cases. At Line 11 when the condition is met, we have $\phi < \frac{\epsilon K}{10(K-|A|)}$, and thus with probability $(1 - \frac{\delta}{100})$ the total error introduced by subroutine OPTMAI at Line 12 is bounded by $(K - |A|)\phi \leq \frac{\epsilon K}{10}$ (Lemma 10). At Line 14, with probability $(1 - \frac{\delta}{100})$ the error introduced by subroutine EPSSPLIT is bounded by $2 \cdot \frac{K_R - K_L}{K - |A|} \cdot (K - |A|) \leq 2 \cdot 2\epsilon^2 K = 4\epsilon^2 K$. (Lemma 9).

By $\mathcal{E}$, Corollary 2, and the errors introduced by boundary cases, we have that with probability $1 - \frac{\delta}{5} - 2 \cdot \frac{\delta}{100} \geq 1 - \delta$, the total error introduced in our top-K estimation is at most $\iota_A + \iota_B + \frac{\epsilon K}{10} + 4\epsilon^2 K \leq \epsilon K$.

3.2.2 Complexity of Algorithm 5

In the whole analysis we assume that $\mathcal{E}$ holds. Recall that by definition $\Delta_i = \max(\theta_i - \theta_{K+1}, \theta_K - \theta_i)$, and $t \in [K]$ is the largest integer such that $\Delta_{K-t} \cdot t \leq K\epsilon$ and $\Delta_{K+1+t} \cdot t \leq K\epsilon$. Recall that $\Psi_t = \min\{\Delta_{K-t}, \Delta_{K+1+t}\}$, and $\Psi_t^\epsilon = \max(\epsilon, \Psi_t)$.

By Theorem 5, Lemma 11, Lemma 12, Lemma 10 and Lemma 9 we have: every call to ESTKTHARM costs $O\left(\frac{|S|}{\phi^2}\log\frac{rr_\phi}{\epsilon\delta}\right)$ pulls; every call to ELIM and REVERSEELIM costs $O\left(\frac{|S|}{\phi^2}\log\frac{r}{\epsilon\delta}\right)$; the call to OPTMAI

costs $O\left(\frac{|S|}{\phi^2}\log\frac{1}{\delta}\right)$; and the call to EPsSPLIT costs $O\left(\frac{|S|}{\phi^2}\log\frac{\widetilde{K}}{\epsilon\delta K}\right) = O\left(\frac{|S|}{\phi^2}\log\frac{1}{\epsilon\delta}\right)$. So our task is to lower bound the value of ϕ when these subroutines are called, and the maximum values of r_ϕ and r.

Lemma 17 *Conditioned on $\mathcal{E}$, at any point of the run of Algorithm 5, we have $\phi = \Omega(\Psi_t^\epsilon)$, and $r_\phi = O(\log\frac{1}{\epsilon}), r = O(\log\frac{n}{|S|})$.*

Proof: First, by the testing condition $10(K-|A|)\phi < K\epsilon$ at Line 10, together with the boundary cases at Line 11-12 and the fact that $\phi \leftarrow \phi/2$ at every update, it holds that

$$10(K-|A|)\cdot 2\phi \geq K\epsilon, \tag{33}$$

which implies $\phi = \Omega(\epsilon)$ when we call all the subroutines.

We now show $\phi = \Omega(\Psi_t)$. From the proof of Lemma 15 we know that during the run of the Algorithm we always have $\widetilde{K} \leq \frac{2|S|}{3}$. We focus on an arbitrary but fixed point during the run of the algorithm. By Theorem 5 we have

$$S\left[\frac{|S|+\widetilde{K}}{2}\right] - \phi \leq \theta^+ \leq S\left[\left(1-\frac{\epsilon^2}{100r^2}\right)\frac{|S|+\widetilde{K}}{2}\right] + \phi, \tag{34}$$

which, combined with the fact that $\widetilde{K} \leq \frac{2|S|}{3}$, gives

$$S\left[\frac{|S|+\widetilde{K}}{2}\right] - \phi \leq \theta^+ \leq S\left[\frac{|S|+\widetilde{K}-\epsilon|S|}{2}\right] + \phi. \tag{35}$$

By (33) we have

$$\frac{\epsilon^2 K}{40} \leq \frac{\epsilon}{40}\cdot 10\widetilde{K}\cdot 2\phi \leq \frac{\epsilon\widetilde{K}}{2}. \tag{36}$$

Applying Claim 9 on both sides of (34), together with (36) and $\widetilde{K} \leq \frac{2|S|}{3}$ we have

$$\theta_{K+\frac{|S|-\widetilde{K}}{2}+\frac{\epsilon^2 K}{40}} - \phi \leq \theta^+ \leq \theta_{K+\frac{|S|-\widetilde{K}-\epsilon|S|}{2}-\frac{\epsilon^2 K}{40}} + \phi$$

$$\Rightarrow \quad \theta_{K+\frac{|S|-\widetilde{K}}{2}+\frac{\epsilon\widetilde{K}}{2}} - \phi \leq \theta^+ \leq \theta_{K+\frac{|S|-\widetilde{K}-\epsilon|S|}{2}-\frac{\epsilon\widetilde{K}}{2}} + \phi$$

$$\Rightarrow \quad \theta^+ \in \theta_{K+\eta}\pm\phi \quad \text{for an} \quad \eta \geq \frac{|S|-\widetilde{K}}{2} - \epsilon|S| \geq 0.16|S|. \tag{37}$$

By symmetry, using a similar argument we can show that for a sufficiently small constant $c_{\eta'}$, we have

$$\theta^- \in \theta_{K-\eta'}\pm\phi \quad \text{for an} \quad \eta' \geq c_{\eta'}|S|, \tag{38}$$

We first consider the case where $K-|A| \leq \frac{|S|}{2}$. We analyze the following two sub-cases.

1a) The case when $\eta > t$. We have:

$$\phi \geq \frac{\theta_K^+ - \theta^+}{6} \quad \text{(by the testing condition at Line 10 and } \phi \leftarrow \phi/2 \text{ at each update)}$$

$$\geq \frac{(\theta_K - 4\phi) - (\theta_{K+\eta}+\phi)}{6} \quad \text{(by Corollary 1 and (37))}$$

$$\geq \frac{\theta_K - 4\phi - (\theta_{K+t+1}+\phi)}{6}$$

$$\geq \frac{\Psi_t - 5\phi}{6} \quad \text{(by the definition of } \Psi_t\text{)}.$$

1b) The case when $\eta \leq t$. We prove by contradiction. Suppose that $\phi \leq \Psi_t/c_t$ for a sufficiently large constant c_t, then

$$\Psi_t\cdot t \geq c_t\phi\cdot\eta \overset{(37)}{\geq} c_t\phi\cdot 0.16|S| \overset{\widetilde{K}\leq\frac{2|S|}{3}}{\geq} c_t\phi\cdot 0.16\cdot 1.5\widetilde{K} \overset{(33)}{>} K\epsilon,$$

A contradition to the definition of t.

20

We then consider the case where $K - |A| > \frac{|S|}{2}$. Now by the testing condition at Line 10 and $\phi \leftarrow \phi/2$ at each update, we know that at least one of the following inequality holds: $\phi \geq \frac{\theta_K^+ - \theta^+}{6}$; or $\phi \geq \frac{\theta^- - \theta_K^-}{6}$. If the first inequality holds, the case-analysis above suffices. Otherwise, we know that the second inequality holds, we analyze the following two sub-cases in a similar fashion.

2a) The case when $\eta' > t$.

$$
\begin{aligned}
\phi &\geq \frac{\theta^- - \theta_K^-}{6} \\
&\geq \frac{(\theta_{K-\eta'} - \phi) - (\theta_{K+1} + 4\phi)}{6} \quad \text{(by Corollary 1 and (38))} \\
&\geq \frac{(\theta_{K-t} - \phi) - (\theta_{K+1} + 4\phi)}{6} \\
&\geq \frac{\Psi_t - 5\phi}{6} \quad \text{(by the definition of } \Psi_t\text{)}.
\end{aligned}
$$

2b) The case when $\eta' \leq t$. This case is symmetric to Case 1b), and we omit here.

Since $\phi = 2^{-r_\phi}$, we immediately have $r_\phi = O(\log \frac{1}{\Psi_t^\epsilon}) = O(\log \frac{1}{\epsilon})$. By the testing condition at Line 13, and the fact that every time we call ELIM and REVERSEELIM we remove a constant fraction of arms from S, we thus have $r = O(\log \frac{n}{|S|})$. $\qquad\square$

We now look at a particular call to ELIM which removes $\frac{1}{10}$-fraction of arms in S, and the tail of S. From the testing condition at Line 10 we know that

$$
\theta_K^+ - \theta^+ \leq 3 \cdot 2\phi. \tag{39}
$$

From Corollary 1 we have that

$$
\theta_K \leq \theta_K^+ + 4\phi. \tag{40}
$$

From (39), (40) and the second inequality of (35), by applying two triangle inequalities we have

$$
\theta_K - S\left[\frac{|S| + \widetilde{K} - \epsilon|S|}{2}\right] \leq 11\phi, \tag{41}
$$

which implies that for all

$$
j \in Q = \left[1.01\widetilde{K}, \frac{|S| + \widetilde{K} - \epsilon|S|}{2}\right] \ (|Q| \geq \frac{|S|}{10} \ \text{since} \ \widetilde{K} \leq \frac{2|S|}{3}), \tag{42}
$$

letting $m(j) \in [n]$ such that $\theta_{m(j)} = S[j]$, we have

$$
\Delta_{m(j)} = \theta_K - S[j] \leq 11\phi. \tag{43}
$$

We thus can charge all the previous cost spent on the $\frac{1}{10}$-fraction of arms in S that are removed by ELIM, which is bounded by $O\left(\frac{|S|}{\phi^2}\left(\log\lceil\log\frac{n}{|S|}\rceil + \log\frac{1}{\epsilon} + \log\frac{1}{\delta}\right)\right)$, to

$$
O\left(\sum_{j \in Q} \frac{1}{\Delta_{m(j)}^2}\left(\log\left\lceil\log\frac{n}{m(j) - K}\right\rceil + \log\frac{1}{\epsilon} + \log\frac{1}{\delta}\right)\right), \tag{44}
$$

where we have used the fact that

$$
\begin{aligned}
|S| &= \Omega(j - \widetilde{K}) \quad \text{(by (42))} \\
&\geq \Omega(m(j) - K). \quad \text{(by (42), Corollary 2 and (36))}
\end{aligned}
$$

Note that it is possible that in multiple calls to ELIM with parameters $(S_1, \cdot, \cdot, \phi_1, \cdot), \ldots, (S_\kappa, \cdot, \cdot, \phi_\kappa, \cdot)$ where $\phi_1 \geq \ldots \geq \phi_\kappa$, we charge the same item $j \in Q_1 \cap \ldots \cap Q_\kappa$ multiple times. However, since

$\phi_{i+1} \le \phi_i/2$ for all $i \in [\kappa - 1]$, the total charge on j is at most twice of that of the last charge (i.e., the one with parameter ϕ_κ).

By symmetry, we can use the same arguments for REVERSEELIM and the head of S, and get a same bound as (44) except that we need to replace $m(j) - K$ with $K - m(j)$. We thus conclude that the total number of pulls can be bounded by

$$O\left(\sum_{j \in [n]} \frac{1}{\Delta_j^2} \left(\log \left\lceil \log \frac{n}{|j - K + \frac{1}{2}|} \right\rceil + \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\right). \tag{45}$$

We know from Lemma 17 that we always have $\phi = \Omega(\Psi_t^\epsilon)$, we can thus "truncate" Expression (45) and bound the total cost by

$$O\left(\sum_{j \in [n]} \frac{1}{\max\{\Delta_j^2, (\Psi_t^\epsilon)^2\}} \left(\log \left\lceil \log \frac{n}{|j - K + \frac{1}{2}|} \right\rceil + \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\right). \tag{46}$$

Now we introduce the following lemma (the proof of which is deferred to the Appendix).

Lemma 18 *If $M > a_1 \ge \ldots \ge a_n \ge 1$, then $\sum_{i \in [n]} a_i \log(n/i) \le O(\lceil \log M \rceil) \sum_{i \in [n]} a_i$.*

With Lemma 18 we can further simplify (46) to

$$O\left(\sum_{j \in [n]} \frac{1}{\max\{\Delta_j^2, (\Psi_t^\epsilon)^2\}} \left(\log \max \left\{ \frac{1}{\max\{\Delta_j^2, (\Psi_t^\epsilon)^2\}} \mid j \in [n] \right\} + \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\right)$$

$$= O\left(\sum_{j \in [n]} \frac{1}{\max\{\Delta_j^2, (\Psi_t^\epsilon)^2\}} \left(\log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\right).$$

4 A Lower Bound

In this section we prove Theorem 3. In Section 4.1, we introduce a lower-bound to a coin-tossing problem. In Section 4.2, we reduce the proof of Theorem 3 to the coin-tossing problem.

4.1 The Coin-Tossing Problem

We say a coin is *p-biased* if the probability that a toss turns *head* is p, and we call p is the *value* of the coin. Set $\eta = 10^{-4}$.

Definition 3 (Coin-Tossing) *In this problem, given a coin that may be $(0.5 + \eta)$-biased or $(0.5 - \eta)$-biased, we want to know its exact value by tosses, and we are allowed to give up and output 'unknown' with probability at most 0.9.*

We have the following theorem.

Theorem 7 *Any algorithm that solves the coin-tossing problem correctly with probability $(1 - \epsilon)$ needs $\Omega(\log 1/\epsilon)$ tosses.*

Proof: Since the input is distributional we only need to focus on deterministic algorithms. Let m be the total number of tosses of the coin, and let $\mathbf{B} = (B_1, \ldots, B_m) \in \{0, 1\}^m$ be the sequence of outcomes. Let $\mathcal{D}_\beta$ be the distribution of $\mathbf{B}$ where each B_i is the outcome of tossing a β-biased coin. For $\mathbf{v} \in \{0, 1\}^m$, let $|\mathbf{v}|$ be the number of 1-coordinates in $\mathbf{v}$.

Our first observation is that for any $\mathbf{b}_1, \mathbf{b}_2 \in \{0,1\}^m$, if $|\mathbf{b}_1| = |\mathbf{b}_2|$, then $\mathbf{Pr}[\mathbf{B} = \mathbf{b}_1] = \mathbf{Pr}[\mathbf{B} = \mathbf{b}_2]$. Therefore, the final output should only depend on the value $|\mathbf{B}|$ but *not* the ordering of the 0/1 sequence. In other words, we can view the output of the algorithm as a function

$$f : \{0, 1, \ldots, m\} \to \{0.5 - \eta, 0.5 + \eta, \perp\},$$

where $\{0, 1, \ldots, m\}$ stand for possible values of $|\mathbf{B}|$, and '$\perp$' represents 'unknown'. Recall that the algorithm can give up and output 'unknown' with probability at most 0.9. By observing that $\mathbf{B} \sim \mathcal{D}_{0.5+\eta}$ and $\mathbf{B} \sim \mathcal{D}_{0.5-\eta}$ are symmetric, the best strategy must set $f(x) = \perp$ for $x \in [0.5m - t, 0.5m + t]$, where $t \in \mathbb{N}$ is the maximum value such that

$$\mathbf{Pr}_{\mathbf{B} \sim \mathcal{D}_{0.5-\eta}}[0.5m - t \leq |\mathbf{B}| \leq 0.5m + t] \leq 0.9. \tag{47}$$

Intuitively, $[0.5m - t, 0.5m + t]$ is the range where conditioned on $|\mathbf{B}| \in [0.5m - t, 0.5m + t]$ the value of the coin is the most uncertain (so that the algorithm simply outputs '$\perp$'). We set $f(x) = 0.5 - \eta$ if $x \in [0, 0.5m - t)$, and $f(x) = 0.5 + \eta$ if $x \in (0.5m + t, m]$. The error probability of this strategy is

$$\mathbf{Pr}_{\mathbf{B} \sim \mathcal{D}_{0.5-\eta}}[|\mathbf{B}| > 0.5m + t]. \tag{48}$$

We now try to upper bound t. First, it is easy to see that $0.5m - t \leq (0.5 - \eta)m$, or $t \geq \eta m$, since otherwise LHS of (47) is at most $1/2$, violating the choice of t. By a Hoeffding's inequality we have

$$\mathbf{Pr}_{\mathbf{B} \sim \mathcal{D}_{0.5-\eta}}[|\mathbf{B}| \leq 0.5m - t] = \mathbf{Pr}_{\mathbf{B} \sim \mathcal{D}_{0.5-\eta}}[|\mathbf{B}| \leq \mathbf{E}[|\mathbf{B}|] - (t - \eta m)] \leq e^{-m(t/m-\eta)^2/2}.$$

We thus have $e^{-m(t/m-\eta)^2/2} \geq (1 - 0.9)/2$, and consequently

$$t \leq \eta m + c_t \sqrt{m} \tag{49}$$

for some large enough constant c_t.

We now lower bound the expression (48). We will need the following anti-concentration result which is an easy consequence of Feller Feller [1943] (cf. Matousek and Vondrák. [2008]).

Fact 1 *(Matousek and Vondrák. [2008]) Let Y be a sum of independent random variables, each attaining values in $[0,1]$, and let $\sigma = \sqrt{\mathbf{Var}[Y]} \geq 200$. Then for all $t \in [0, \sigma^2/100]$, we have*

$$\mathbf{Pr}[Y \geq \mathbf{E}[Y] + t] \geq c \cdot e^{-t^2/(3\sigma^2)}$$

for a universal constant $c > 0$.

In our case, since $|\mathbf{B}|$ can be seen as a sum of Bernoulli variables with $p = 0.5 - \eta$, $\mathbf{Var}|\mathbf{B}| = m \cdot (0.5 - \eta) \cdot (0.5 + \eta) \geq 0.24m$. By (49) we have $\eta m + t \leq 2\eta m + c_t\sqrt{m} \leq \mathbf{Var}|\mathbf{B}|/100$ by our choice of η. Thus by applying Fact 1 we have

$$
\begin{aligned}
\mathbf{Pr}_{\mathbf{B} \sim \mathcal{D}_{0.5-\eta}}[|\mathbf{B}| > 0.5m + t] & = & \mathbf{Pr}_{\mathbf{B} \sim \mathcal{D}_{0.5-\eta}}[|\mathbf{B}| > \mathbf{E}[|\mathbf{B}|] + (\eta m + t)] \\
& \geq & c \cdot e^{-(\eta m+t)^2/(3 \cdot 0.24m)} \\
& \geq & e^{-\Omega(m)}.
\end{aligned}
$$

To make the best strategy succeeds with probability at least $1 - \epsilon$, we have to make $e^{-\Omega(m)} \leq \epsilon$, which means we have to set $m \geq \Omega(\log 1/\epsilon)$. $\qquad\square$

4.2 The Reduction

We show a reduction from the coin-tossing problem to the ϵ-top-K arms problem. For technical convenience we set $K = n/2$, and assume that $\epsilon K \geq c_K$ for a large enough constant c_K.

Lemma 19 *If there is an algorithm for ϵ-top-K arms that succeeds with probability 0.9 using $C \leq f(n, K)/poly(\epsilon)$ tosses, then there is an algorithm for coin-tossing that succeeds with $(1-\epsilon)$ using $O(C/n)$ tosses. Moreover, the instances fed into the ϵ-top-K arms algorithm have the property that $H^{(t,\epsilon)} = \Theta(n\eta^{-2}) = \Theta(n)$ for $\epsilon \geq c_K/K$.*

We prove Lemma 19 in two steps. We first perform an input reduction, and then show that we can construct an efficient algorithm for coin-tossing using an algorithm for ϵ-top-K arms.

Input reduction. Given an input X for coin-tossing, we construct an input $Y = (X_1, \ldots, X_n)$ for ϵ-top-K arms as follows: we randomly pick a set $S \subseteq [n]$ with $|S| = K$, and set X_i ($i \in S$) to be $(0.5+\eta)$-biased coins (denoted by $X_i = 0.5 + \eta$ for convenience), and X_i ($i \neq [n]\backslash S$) to be $(0.5-\eta)$-biased coins. We then pick a random index $I \in [n]$, and reset $X_I = X$. Since in our input Y, the number of $(0.5 + \eta)$-biased coins is either $K - 1$, K, or $K + 1$, while the rest are $(0.5 - \eta)$-biased coins, it can be checked that $H^{(t,\epsilon)} = \Theta(n\eta^{-2})$ for $\epsilon \in [c_K/K, \eta]$.

Claim 10 *If S' is a set of γ-top-K arms ($\gamma \geq 2\eta/K$) on Y, then with probability at least $1 - 2\gamma/\eta$ we can correctly determine the value of X by checking whether $I \in S'$.*

Proof: If $X = X_I = 0.5 + \eta$, then to compute ϵ-top-K arms correctly we need to output a set S' such that

$$\frac{1}{K} \cdot \sum_{i \in S'} X_i \geq \frac{1}{K} \cdot \sum_{i \in S} X_i - \gamma \geq (0.5 + \eta) - \gamma.$$

By simple calculation we must have $|\{i \in S' \mid X_i = 0.5 + \eta\}| \geq \left(1 - \frac{\gamma}{2\eta}\right) K$. Since all $(0.5 + \eta)$-biased coins are *symmetric*, the probability that $I \in S'$ is at least

$$\frac{\left(1 - \frac{\gamma}{2\eta}\right)}{K + 1} \geq \left(1 - \frac{\gamma}{\eta}\right). \tag{50}$$

Otherwise if $X = X_I = 0.5 - \eta$, then to compute ϵ-top-K arms correctly we need to output a set S' such that

$$\frac{1}{K} \cdot \sum_{i \in S'} X_i \geq \frac{1}{K} \cdot \sum_{i \in S} X_i - \gamma - \frac{2\eta}{K} \geq 0.5 + \eta - \gamma - \frac{2\eta}{K},$$

By simple calculation we must have $|\{i \in S' \mid X_i = 0.5 + \eta\}| \geq \left(1 - \frac{\gamma}{\eta}\right) K$, or, $|\{i \in S' \mid X_i = 0.5 - \eta\}| \leq \frac{\gamma}{\eta} K$. Again since all $(0.5 - \eta)$-biased coins are symmetric, the probability that $I \in S'$ is at most

$$\frac{\frac{\gamma}{\eta} K}{n - (K - 1)} \leq \frac{2\gamma}{\eta}. \tag{51}$$

By (50) and (51), we conclude that by observing whether $I \in S'$ or not we can determine whether $X = X_I = 0.5 + \eta$ or $0.5 - \eta$ correctly with probability at least $1 - 2\gamma/\eta$. $\qquad\square$

An algorithm for coin-tossing. Let $\epsilon' = \eta/4 \cdot \epsilon$. We now construct an algorithm $\mathcal{A}'$ for coin-tossing using an algorithm $\mathcal{A}$ for ϵ'-top-K arms. Given an input X for coin-tossing, we first perform the input reduction as described above, getting an input Y. We then run $\mathcal{A}$ on Y. We give up and output 'unknown' during the run of $\mathcal{A}$ if the number of tosses on X_I is more than $20C/n$. If $\mathcal{A}$ finishes then let S' be the outputted set of top-K coins. We then perform a verification step to test whether S' is indeed a set of

ϵ'-top-K arms, and output 'unknown' if the verification fails. The verification is done as follows: we first compute $\rho = \frac{1}{|S' \setminus I|} \cdot \sum_{i \in S' \setminus I} X_i$, and then verify whether $\rho \geq (0.5 + \eta) - (\epsilon' + 2\eta/K)$. Finally, if we have not outputted 'unknown', we output $X = 0.5 + \eta$ if $I \in S'$, and $X = 0.5 - \eta$ if $I \notin S'$.

We now try to bound the probability that $\mathcal{A}'$ outputs 'unknown'.

Claim 11 *The probability that we give up during the run of $\mathcal{A}$ is at most 0.1.*

Proof: We prove for the case $X = 0.5 + \eta$; same arguments hold for the case $X = 0.5 - \eta$ since we have set $K = n/2$. Note that if $I \in S$, then we have K coins (including X_I) in Y that has value $(0.5 + \eta)$; otherwise if $I \notin S$ then we have $K + 1$ such coins. By symmetry, the expected tosses on X_I is at most $C/K = 2C/n$. By a Markov inequality the probability that X_I has been tossed by at least $20C/n$ is at most 0.1. $\qquad\square$

Claim 12 *Suppose we do not give up when running $\mathcal{A}$, the verification step fails with probability at most 0.1.*

Proof: Note that $\mathcal{A}'$ knows the value of all other coins except X_I, simply because $\{X_i \mid i \in [n] \setminus I\}$ are all constructed by $\mathcal{A}'$. The $2\eta/K$ factor in the test $\rho \geq (0.5 + \eta) - (\epsilon' + 2\eta/K)$ comes from the fact that we do not know the exact value of X_I, which will affect the estimation of the $\frac{1}{K} \sum_{i \in S'} X_i$ by at most an additive factor $2\eta/K$. Therefore the failure probability of the verification is at most the failure probability of $\mathcal{A}$, which is upper bounded by 0.1. $\qquad\square$

Now we are ready to prove the lemma.

Proof:(of Lemma 19) First, note that if there is an algorithm for ϵ-top-K arms that succeeds with probability 0.9 using $C \leq f(n, K) \cdot \text{poly}(\epsilon)$ tosses, then there is an algorithm for ϵ'-top-K arms ($\epsilon' = 4/\eta \cdot \epsilon$ for a constant η) that succeeds with probability 0.9 using $O(C)$.

We now show that Algorithm $\mathcal{A}'$ constructed above for coin-tossing has the following properties, which conclude the lemma.

1. It tosses X at most $O(C/n)$ times.

2. It outputs 'unknown' with probability at most 0.9.

3. When it does not output 'unknown', it successfully computes X with probability at least $1 - \epsilon$.

The first item holds according to the construction of $\mathcal{A}'$. For the second item, the probability that $\mathcal{A}'$ outputs 'unknown' is upper bounded by the sum of the probability that we give up when running $\mathcal{A}$ and the failure probability of $\mathcal{A}$, which is at most $0.1 + 0.1 < 0.9$ by Claim 11 and Claim 12. For the third item, note that any S' that passes the verification step in $\mathcal{A}'$ is a set of $(\epsilon' + 2\eta/K)$-top-K arms. The item holds by applying Claim 10 (setting $\gamma = \epsilon' + 2\eta/K$). Note that $2\gamma/\eta = (2/\eta) \cdot (\epsilon' + 2\eta/K) \leq \epsilon$ since $\epsilon' = \eta/4 \cdot \epsilon$ and $\epsilon K \geq c_K$ for a sufficiently large constant c_K. $\qquad\square$

By Theorem 7 and Lemma 19 we have the following theorem.

Theorem 8 *Any algorithm that computes that ϵ-top-K arms correctly with probability 0.9 needs $\Omega(n \log \epsilon^{-1})$ tosses.*

Together with the bound for $H^{(t,\epsilon)}$ in Lemma 19, we prove Theorem 3.

5 Experiments

In this section we present the experimental results. While our theorems are presented in the PAC form, it is in general difficult to verify them directly because the parameter ϵ is merely an upper bound and the actual aggregate regret may deviate from it. In our experiment, we convert our Algorithm 1 to the fixed-budget version (that is, fix the budget of the number of pulls and calculate the aggregate regret). We compare our Algorithm 1 (AdaptiveTopK) with two state-of-the-art methods – OptMAI in Zhou et al. [2014] and CLUCB-PAC in Chen et al. [2014]. The comparison between OptMAI /CLUCB-PAC and previous methods (e.g., the methods in Bubeck et al. [2013] and Kalyanakrishnan et al. [2012]) have already been demonstrated in Zhou et al. [2014] and Chen et al. [2014], and thus omitted here for the clarity of the presentation. To convert our algorithm to the fixed-budget version, we remove the outer while loop of Algorithm 1. As a replacement, we keep track of the total number of pulls, and stop pulling the arms once the budget is exhausted.

We test our algorithm on both synthetic and real datasets as described as follows. For simulated datasets, we set the total number of arms $n = 1,000$ and vary the parameter K. We set the tolerance parameter $\epsilon = 0.01$. In AdaptiveTopK and CLUCB-PAC, another parameter δ (i.e., the failure probability) is required and we set $\delta = 0.01$.

- TwoGroup: the mean reward for the top K arms is set to 0.7 and that for the rest of the arms is set to 0.3.

- Uniform: we set $\theta_i = 1 - \frac{i}{n}$ for $1 \leq i \leq n$.

- Synthetic-p: we set $\theta_i = (1 - \frac{K}{n}) + \frac{K}{n} \cdot (1 - \frac{i}{K})^p$ for each $i \leq K$ and $\theta_i = (1 - \frac{K}{n}) - \frac{n-K}{n} \cdot (\frac{i-K}{n-K})^p$ for each $i > K$. Note that Synthetic-1 is identical to Uniform. When p is larger than 1, arms are made closer to the boundary that separates the top-K from the rest (i.e. $1 - \frac{K}{n}$). When p is smaller than 1, arms are made farther to the boundary. We normalize all the arms such that the mean values of the arms still span the whole interval $[0, 1]$. We consider $p = .5, 1, 6$.

- RTE: We generate θ from a real recognizing textual entailment (RTE) dataset Snow et al. [2008]. There are $n = 164$ workers and we set each θ_i be the true labeling accuracy of the i-th worker. Note that the true label for each instance is provided in this dataset.

For each dataset, we first fix the budget (total number of pulls allowed) and run each algorithm 200 times. For each algorithm, we calculate the empirical probability (over 200 runs) that the aggregate regret of the selected arms is above the tolerance threshold $\epsilon = 0.01$, which is called *failure probability*. A smaller failure probability means better performance. For each dataset and different K, we plot the curve of failure probability by varying the number of pulls. The results are shown in Figure 1-5.
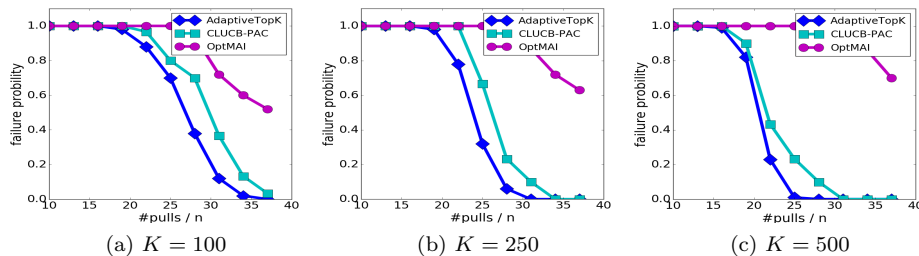

Figure 1: TwoGroup dataset

It can be observed from the experimental results that AdaptiveTopK (Algorithm 1) outperforms CLUCB-PAC in almost all the datasets. When K is relatively small, OptMAI has the best performance
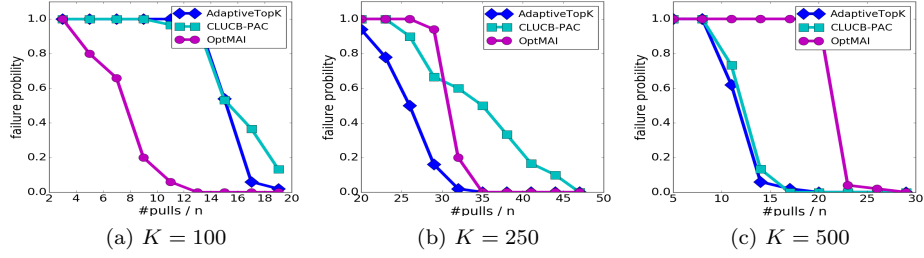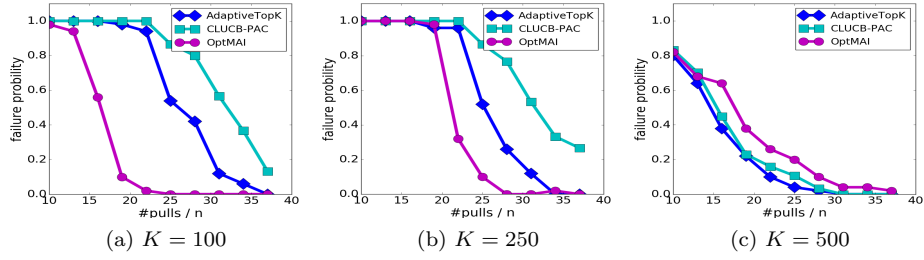
(a) $K = 100$ (b) $K = 250$ (c) $K = 500$

Figure 2: Synthetic-.5 dataset

(a) $K = 100$ (b) $K = 250$ (c) $K = 500$

Figure 3: Uniform dataset

(a) $K = 100$ (b) $K = 250$ (c) $K = 500$

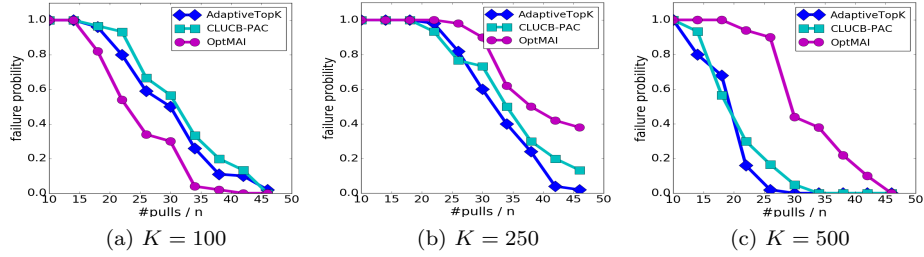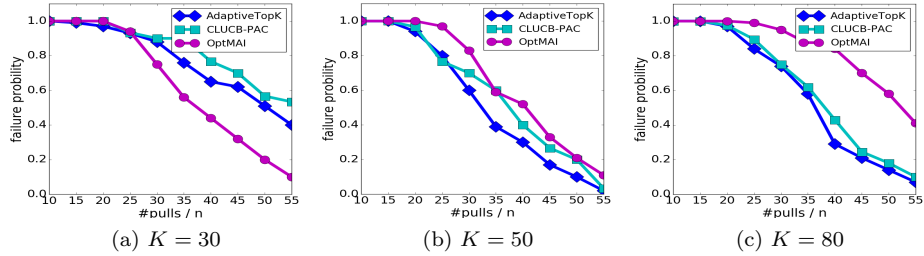Figure 4: Synthetic-6 dataset

(a) $K = 30$ (b) $K = 50$ (c) $K = 80$

Figure 5: Rte dataset

in most datasets. When K is large, AdaptiveTopK outperforms OptMAI . The details of the experimental results are elaborated as follows.

- For TwoGroup dataset (see Figure 1), AdaptiveTopK outperforms other algorithms significantly for all values of K. The advantage comes from the adaptivity of our algorithm. In the TwoGroup dataset,

27

top-K arms are very well separated from the rest. Once our algorithm identifies this situation, it need only a few pulls to classify the arms. In details, the inner while loop (Line 7) of Algorithm 1 make it possible to accept/reject a large number of arms in one round as long as the algorithm is confident.

- As K increases, the advantage of AdaptiveTopK over other algorithms (OptMAI in particular) becomes more significant. This can be explained by the definition of $H^{(t,\epsilon)}$: $t = t(\epsilon, K)$ usually becomes bigger as K grows, leading to a smaller hardness parameter $H^{(t,\epsilon)}$.

- A comparison between Synthetic-.5, Uniform, Synthetic-6 reveals that the advantage of AdaptiveTopK over other algorithms (OptMAI in particular) becomes significant in both extreme scenarios, i.e., when arms are very well separated ($p \ll 1$) and when arms are very close to the separation boundary ($p \gg 1$).

6   Conclusion and Future Work

In this paper, we proposed two algorithms for a PAC version of the multiple-arm identification problem in a stochastic multi-armed bandit (MAB) game. We introduced a new hardness parameter for characterizing the difficulty of an instance when using the aggregate regret as the evaluation metric, and established the instance-dependent sample complexity based on this hardness parameter. We also established lower bound results to show the optimality of our algorithm in the worst case. Although we only consider the case when the reward distribution is supported on $[0, 1]$, it is straightforward to extend our results to sub-Gaussian reward distributions.

For future directions, it is worthwhile to consider more general problem of pure exploration of MAB under matroid constraints, which includes the multiple-arm identification as a special case, or other polynomial-time-computable combinatorial constraints such as matchings. It is also interesting to extend the current work to finding top-K arms in a linear contextual bandit framework.

Appendix

Proof of Lemma 18

Proof: We partition all a_i's to groups $G_1, \ldots, G_{\lceil \log_2 M \rceil}$ where $G_j = \{i \in [n] \mid a_i \in [2^{j-1}, 2^j)\}$. Let $S = \sum_{i=1}^n a_i \geq n$. For each $j \in \{1, 2, \ldots, \lceil \log_2 M \rceil\}$, let $\delta_j = \frac{1}{S} \sum_{i \in G_j} a_i$. Observe that we have $\sum_j \delta_j = 1$. We will show for each G_j that

$$\sum_{i \in G_j} a_i \log(n/i) \leq O\left(\delta_j \log M + \delta_j \log \delta_j^{-1} + 1\right) S. \tag{52}$$

Once we establish (52), we prove the lemma as follows. We sum up the inequalities for all j and get

$$\sum_{i \in [n]} a_i \log(n/i) \leq \sum_{j=1}^{\lceil \log_2 M \rceil} O\left(\delta_j \log M + \delta_j \log \delta_j^{-1} + 1\right) S$$

$$= O(\lceil \log M \rceil) S + O(S) \sum_{j=1}^{\lceil \log_2 M \rceil} \delta_j \log \delta_j^{-1} \leq O(\lceil \log M \rceil) S + O(S) \cdot \log \lceil \log_2 M \rceil \leq O(\lceil \log M \rceil) S,$$

where the second last inequality is by Jensen's inequality and the convexity of $\delta \log \delta$ over $\delta \in (0, 1)$.

Now we prove (52) for each group $G = G_j$ and $\delta = \delta_j$. Let $b = \max_{i \in G}\{a_i\} \leq M$. By our partition rule we have that $b \leq 2a_i$ for all $i \in G$. Observe that

$$\sum_{i \in G} a_i \log(n/i) \leq b \sum_{i=1}^{|G|} \log(n/i) = b|G| \log n - b \log(|G|!) \leq b|G| \log(n/|G|) + O(b|G|). \qquad (53)$$

The last inequality of (53) is by Stirling's approximation. Since $b|G| \geq \sum_{i \in G} a_i = \delta S$, we have $|G| \geq \frac{\delta S}{b} \geq \frac{\delta n}{b}$. We finish the proof of (52) by upper-bounding the RHS of (53) by

$$b|G| \log(b\delta^{-1}) + O(b|G|) \leq 2\delta S \log(b\delta^{-1}) + O(b|G|) \leq O\left(\delta S \log(M\delta^{-1}) + S\right),$$

where the first inequality is because $b|G| \leq \sum_{i \in G} 2a_i = 2\delta S$.

$\square$

References

Y. Abbasi-Yadkori, P. Bartlett, X. Chen, and A. Malek. Large-scale markov decision problems with KL control cost and its application to crowdsourcing. In *Proceedings of International Conference on Machine Learning (ICML)*, 2015.

J. Audibert, S. Bubeck, and R. Munos. Best arm identification in multi-armed bandits. In *Proceedings of the Conference on Learning Theory (COLT)*, 2010.

S. Bubeck, T. Wang, and N. Viswanathan. Multiple identifications in multi-armed bandits. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2013.

W. Cao, J. Li, Y. Tao, and Z. Li. On top-k selection in multi-armed bandits and hidden bipartite graphs. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2015.

L. Chen, A. Gupta, and J. Li. Pure exploration of multi-armed bandit under matroid constraints. In *Proceedings of the Conference on Learning Theory (COLT)*, 2016a.

L. Chen, J. Li, and M. Qiao. Towards instance optimal bounds for best arm identification. arXiv preprint arXiv:1608.06031, 2016b.

S. Chen, T. Lin, I. King, M. R. Lyu, and W. Chen. Combinatorial pure exploration of multi-armed bandits. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2014.

X. Chen, Q. Lin, and D. Zhou. Statistical decision making for optimal budget allocation in crowd labeling. *Journal of Machine Learning Research*, 16:1–46, 2015.

E. Even-Dar, S. Mannor, and Y. Mansour. PAC bounds for multi-armed bandit and markov decision processes. In *Proceedings of the Annual Conference on Learning Theory (COLT)*, 2002.

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

W. Feller. Generalization of a probability limit theorem of cramer. *Trans. Amer. Math. Soc*, 54(3): 361–372, 1943.

V. Gabillon, M. Ghavamzadeh, A. Lazaric, and S. Bubeck. Multi-bandit best arm identification. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2011.

V. Gabillon, M. Ghavamzadeh, and A. Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2012.

V. Gabillon, A. Lazaric, M. Ghavamzadeh, R. Ortner, and P. Bartlett. Improved learning complexity in combinatorial pure exploration bandits. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2016.

K. Jamieson, M. Malloy, R. Nowak, , and S. Bubeck. UCB : An optimal exploration algorithm for multi-armed bandits. In *Proceedings of the Conference on Learning Theory (COLT)*, 12 2014. arXiv preprint arXiv:1312.7308v1.

K.-S. Jun, K. Jamieson, R. Nowak, and X. Zhu. Top arm identification in multi-armed bandits with batch arm pulls. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2016.

S. Kalyanakrishnan, A. Tewari, P. Auer, and P. Stone. PAC subset selection in stochastic multi-armed bandits. In *Proceedings of International Conference on Machine Learning (ICML)*, 2012.

Z. Karnin, T. Koren, and O. Somekh. Almost optimal exploration in multi-armed bandits. In *Proceedings of International Conference on Machine Learning (ICML)*, 2013.

E. Kaufmann, O. Cappé, and A. Garivier. On the complexity of best arm identification in multi-armed bandit models. *Journal of Machine Learning Research*, 17:1–42, 2016.

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

J. Matousek and J. Vondrák. The probabilistic method. Lecture Notes, 2008.

D. Russo. Simple bayesian algorithms for best arm identification. In *Proceedings of the Conference on Learning Theory (COLT)*, 2016.

R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng. Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 254–263, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics. URL `http://dl.acm.org/citation.cfm?id=1613715.1613751`.

M. Soare, A. Lazaric, and R. Munos. Best-arm identification in linear bandits. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2014.

Y. Zhou, X. Chen, and J. Li. Optimal PAC multiple arm identification with applications to crowdsourcing. In *Proceedings of International Conference on Machine Learning (ICML)*, 2014.