
Nearly Instance Optimal Sample Complexity Bounds for Top-k Arm Selection

Lijie Chen

IIS, Tsinghua University

Jian Li

IIS, Tsinghua University

Mingda Qiao

IIS, Tsinghua University

Abstract

In the Best- k -Arm problem, we are given n stochastic bandit arms, each associated with an unknown reward distribution. We are required to identify the k arms with the largest means by taking as few samples as possible. In this paper, we make progress towards a complete characterization of the instance-wise sample complexity bounds for the Best- k -Arm problem. On the lower bound side, we obtain a novel complexity term to measure the sample complexity that every Best- k -Arm instance requires. This is derived by an interesting and nontrivial reduction from the Best-1-Arm problem. We also provide an elimination-based algorithm that matches the instance-wise lower bound within doubly-logarithmic factors. The sample complexity of our algorithm strictly dominates the state-of-the-art for Best- k -Arm (module constant factors).

1 INTRODUCTION

The stochastic multi-armed bandit is a classical and well-studied model for characterizing the exploration-exploitation tradeoff in various decision-making problems in stochastic settings. The most well-known objective in the multi-armed bandit model is to maximize the cumulative gain (or equivalently, to minimize the cumulative regret) that the agent achieves. Another line of research, called the pure exploration multi-armed bandit problem, which is motivated by a variety of practical applications including medical trials [Rob85, AB10], communication network [AB10], and crowdsourcing [ZCL14, CLTL15], has also attracted significant attention recently. In the pure exploration problem, the agent draws samples from the

arms adaptively (the exploration phase), and finally commits to one of the feasible solutions specified by the problem. In a sense, the exploitation phase in the pure exploration problem simply consists of exploiting the solution to which the agent commits indefinitely. Therefore, the agent's objective is to identify the optimal (or near-optimal) feasible solution with high probability.

In this paper, we focus on the problem of identifying the top- k arms (i.e., the k arms with the largest means) in a stochastic multi-armed bandit model. The problem is known as the Best- k -Arm problem, and has been extensively studied in the past decade [KS10, GGL12, GGLB11, KTAS12, BWV12, KK13, ZCL14, KCG15, SJR16]. We formally define the Best- k -Arm problem as follows.

Definition 1.1 (Best- k -Arm). *An instance of Best- k -Arm is a set of stochastic arms $\mathcal{I} = \{A_1, A_2, \dots, A_n\}$. Each arm has a 1-sub-Gaussian reward distribution with an unknown mean in $[0, 1/2]$.*

At each step, algorithm \mathbb{A} chooses an arm and observes an i.i.d. sample from its reward distribution. The goal of \mathbb{A} is to identify the k arms with the largest means in \mathcal{I} using as few samples as possible. Let $\mu_{[i]}$ denote the i -th largest mean in an instance of Best- k -Arm. We assume that $\mu_{[k]} > \mu_{[k+1]}$ in order to ensure the uniqueness of the solution.

Note that in our upper bound, we assume that all reward distributions are 1-sub-Gaussian¹, which is a standard assumption in multi-armed bandit literature. In our lower bound (Theorem 1.1), however, we assume that all reward distributions are Gaussian with unit variance.²

When we only want to identify the single best arm, we get the following Best-1-Arm problem, which is a well-studied special case of Best- k -Arm. The problem plays

¹A distribution \mathcal{D} is σ -sub-Gaussian, if it holds that $\mathbb{E}_{X \sim \mathcal{D}}[\exp(tX - t\mathbb{E}_{X \sim \mathcal{D}}[X])] \leq \exp(\sigma^2 t^2 / 2)$ for all $t \in \mathbb{R}$.

²For arbitrary distributions, one may be able to distinguish two distributions with very close means using very few samples. It is impossible to establish a nontrivial lower bound in such generality.

an important role in our lower bound for Best- k -Arm.

Definition 1.2 (Best-1-Arm). *The Best-1-Arm problem is a special case of Best- k -Arm where $k = 1$.*

Generally, we focus on algorithms that solve Best- k -Arm with probability at least $1 - \delta$.

Definition 1.3 (δ -correct Algorithms). *\mathbb{A} is a δ -correct algorithm for Best- k -Arm if and only if \mathbb{A} returns the correct answer with probability at least $1 - \delta$ on every Best- k -Arm instance \mathcal{I} .*

1.1 Our Results

Before stating our results on the Best- k -Arm problem, we first define a few useful notations that characterize the hardness of Best- k -Arm instances.

1.1.1 Notations

Means and gaps. Let μ_A denote the mean of arm A . $\mu_{[i]}$ denotes the i -th largest mean among all arms in a specific instance. We define the *gap* of arm A as

$$\Delta_A = \begin{cases} \mu_A - \mu_{[k+1]}, & \mu_A \geq \mu_{[k]}, \\ \mu_{[k]} - \mu_A, & \mu_A \leq \mu_{[k+1]}. \end{cases}$$

Note that the gap of an arm is the minimum value by which its mean needs to change in order to alter the top k arms. We let $\Delta_{[i]}$ denote the gap of the i -th largest arm.

Arm groups. Let ε_r denote 2^{-r} . For an instance \mathcal{I} of Best- k -Arm and positive integer r , we define the arm groups as

$$G_r^{\text{large}} = \{A \in \mathcal{I} : \mu_A \geq \mu_{[k]}, \Delta_A \in (\varepsilon_{r+1}, \varepsilon_r]\}, \text{ and}$$

$$G_r^{\text{small}} = \{A \in \mathcal{I} : \mu_A \leq \mu_{[k+1]}, \Delta_A \in (\varepsilon_{r+1}, \varepsilon_r]\}.$$

In other words, G_r^{large} and G_r^{small} contain the arms with gaps in $(\varepsilon_{r+1}, \varepsilon_r]$ among and outside the best k arms, respectively.

Note that since we assume that the mean of each arm is in $[0, 1/2]$, the gap of every arm is at most $1/2$. Therefore by definition each arm is contained in one of the arm groups.

We also use the following shorthand notations:

$$G_{\geq r}^{\text{large}} = \bigcup_{i=r}^{\infty} G_i^{\text{large}} \text{ and } G_{\geq r}^{\text{small}} = \bigcup_{i=r}^{\infty} G_i^{\text{small}}.$$

1.1.2 Lower Bound

In order to state our instance-wise lower bound precisely, we need to elaborate what is an instance. By

Definition 1.1, a given instance is a *set* of arms, meaning the particular input order of the arms should not matter. Note that there indeed exists algorithms that take advantage of the input order and may perform better for some ‘‘lucky’’ input orders than the others.³ In order to prove a tighter lower bound, we need to consider all possible input orders and take the average. From technical perspective, we use the following definition of an instance.

Definition 1.4 (Instance). *An instance is considered as a random permutation of a sequence of arms. Consequently, the sample complexity of an algorithm on an instance should be considered as the average of the number of samples over all permutations.*

In fact, the random permutation is crucial to establishing *instance-wise* lower bounds for Best- k -Arm (i.e., the minimum number of samples that *every* δ -correct algorithm for Best- k -Arm needs to take on an instance). Without the random permutation, the algorithm might use fewer samples on some ‘‘lucky’’ permutations than on others, and it is impossible to prove a tight instance-wise lower bound as ours. The use of random permutation to define instance-wise lower bounds is also used in computational geometry [ABC09] and the Best-1-Arm problem [CL15, CL16b].

We say that an instance of Best- k -Arm is Gaussian, if all reward distributions are normal distributions with unit variance.

Theorem 1.1. *There exists a constant $\delta_0 > 0$, such that for any $\delta < \delta_0$, every δ -correct algorithm for Best- k -Arm takes*

$$\Omega(H \ln \delta^{-1} + H^{\text{large}} + H^{\text{small}})$$

samples in expectation on every Gaussian instance. Here $H = \sum_{i=1}^n \Delta_{[i]}^{-2}$,

$$H^{\text{large}} = \sum_{i=1}^{\infty} |G_i^{\text{large}}| \cdot \max_{j \leq i} \varepsilon_j^{-2} \ln |G_{\geq j}^{\text{small}}|, \text{ and}$$

$$H^{\text{small}} = \sum_{i=1}^{\infty} |G_i^{\text{small}}| \cdot \max_{j \leq i} \varepsilon_j^{-2} \ln |G_{\geq j}^{\text{large}}|.$$

We notice that Simchowitz et al. [SJR16], independently of our work, derived instance-wise lower bounds for Best- k -Arm similar to Theorem 1.1, using a somewhat different method.

³For example, a sorting algorithm can first check if the input sequence a_1, \dots, a_n is in increasing order in $O(n)$ time, and then run an $O(n \log n)$ time algorithm. This algorithm is particularly fast for a particular input order.

1.1.3 Upper Bound

Theorem 1.2. *For all $\delta > 0$, there is a δ -correct algorithm for Best- k -Arm that takes*

$$O\left(H \ln \delta^{-1} + \tilde{H} + \tilde{H}^{\text{large}} + \tilde{H}^{\text{small}}\right)$$

samples in expectation on every instance. Here

$$\tilde{H} = \sum_{i=1}^n \Delta_{[i]}^{-2} \ln \ln \Delta_{[i]}^{-1},$$

$$\tilde{H}^{\text{large}} = \sum_{i=1}^{\infty} |G_i^{\text{large}}| \sum_{j=1}^i \varepsilon_j^{-2} \ln |G_j^{\text{small}}|, \text{ and}$$

$$\tilde{H}^{\text{small}} = \sum_{i=1}^{\infty} |G_i^{\text{small}}| \sum_{j=1}^i \varepsilon_j^{-2} \ln |G_j^{\text{large}}|.$$

The following theorem relates the \tilde{H}^{large} and \tilde{H}^{small} terms to H^{large} and H^{small} in the lower bound.

Theorem 1.3. *For every Best- k -Arm instance, the following statements hold:*

1. $\tilde{H}^{\text{large}} + \tilde{H}^{\text{small}} = O\left((H^{\text{large}} + H^{\text{small}}) \ln \ln n\right)$.
2. $\tilde{H}^{\text{large}} + \tilde{H}^{\text{small}} = O(H \ln k)$.

Combining Theorems 1.1, 1.2 and 1.3(1), our algorithm is *instance-wise optimal* within doubly-logarithmic factors (i.e., $\ln \ln n, \ln \ln \Delta_{[i]}^{-1}$). In other words, the sample complexity of our algorithm on *every* single instance nearly matches the minimum number of samples that *every* δ -correct algorithm has to take on that instance.

Theorem 1.2 and Theorem 1.3(2) also imply that our algorithm strictly dominates the state-of-the-art algorithm for Best- k -Arm obtained in [CGL16], which achieves a sample complexity of

$$O\left(\sum_{i=1}^n \Delta_{[i]}^{-2} \left(\ln \delta^{-1} + \ln k + \ln \ln \Delta_{[i]}^{-1}\right)\right) \\ = O\left(H \ln \delta^{-1} + H \ln k + \tilde{H}\right).$$

In particular, we give a specific example in Appendix A in which the sample complexity achieved by Theorem 1.2 is significantly better than that obtained in [CGL16]. See Table 1 for more previous upper bounds on the sample complexity of Best- k -Arm.

1.2 Related Work

Best-1-Arm. In the Best-1-Arm problem, the algorithm is required to identify the arm with the largest

Table 1: Upper Bounds of Best- k -Arm

Source	Sample Complexity
[GGL12]	$O\left(H \ln \delta^{-1} + H \ln H\right)$
[KTAS12]	$O\left(H \ln \delta^{-1} + H \ln H\right)$
[CLK ⁺ 14]	$O\left(H \ln \delta^{-1} + H \ln H\right)$
[CGL16]	$O\left(H \ln \delta^{-1} + \tilde{H} + H \ln k\right)$
This paper	$O\left(H \ln \delta^{-1} + \tilde{H} + \tilde{H}^{\text{large}} + \tilde{H}^{\text{small}}\right)$

mean. As a special case of Best- k -Arm, the problem has a history dating back to 1954 [Bec54]. The problem continues to attract significant attention over the past decade [AB10, EDMM06, MT04, JMN14, KKS13, CL15, CL16a, GK16, CLQ16].

Combinatorial pure exploration. The combinatorial pure exploration problem, which further generalizes the cardinality constraint in Best- k -Arm (i.e., to choose exactly k arms) to combinatorial constraints (e.g., matroid constraints), was also studied [CLK⁺14, CGL16, GLG⁺16].

PAC learning. In the PAC learning setting, the algorithm is required to find an approximate solution to the pure exploration problem. The sample complexity of Best-1-Arm and Best- k -Arm in PAC setting has been extensively studied. A tight (worst case) bound of $\Theta(n\varepsilon^{-2} \ln \delta^{-1})$ was obtained for the PAC version of the Best-1-Arm problem in [EDMM02, EDMM06, MT04]. The worst case sample complexity of Best- k -Arm in the PAC setting has also been well-studied [KS10, KTAS12, ZCL14, CLTL15].

2 PRELIMINARIES

Kullback-Leibler divergence. Let $\text{KL}(P, Q)$ denote the Kullback-Leibler divergence from distribution Q to P . The following well-known fact (e.g., a special case of [Duc07]) states the Kullback-Leibler divergence between two normal distributions with unit variance.

Fact 2.1. *Let $\mathcal{N}(\mu, \sigma^2)$ denote the normal distribution with mean μ and variance σ^2 . It holds that*

$$\text{KL}(\mathcal{N}(\mu_1, 1), \mathcal{N}(\mu_2, 1)) = \frac{(\mu_1 - \mu_2)^2}{2}.$$

Binary relative entropy. Let

$$d(x, y) = x \ln(x/y) + (1-x) \ln[(1-x)/(1-y)]$$

be the binary relative entropy function. The monotonicity of $d(\cdot, \cdot)$ is useful to our following analysis.

Fact 2.2. *For $0 \leq y \leq y_0 \leq x_0 \leq x \leq 1$, $d(x, y) \geq d(x_0, y_0)$.*

Probability and expectation. $\Pr_{\mathbb{A}, \mathcal{I}}$ and $\mathbb{E}_{\mathbb{A}, \mathcal{I}}$ denote the probability and expectation when algorithm \mathbb{A} runs on instance \mathcal{I} . These notations are useful since we frequently consider the execution of different algorithms on various instances in our proof of the lower bound.

Change of Distribution. The following ‘‘Change of Distribution’’ lemma, developed in [KCG15], is a useful tool to quantify the behavior of an algorithm when the instance is modified.

Lemma 2.1 (Change of Distribution). *Suppose algorithm \mathbb{A} runs on n arms. $\mathcal{I} = (A_1, A_2, \dots, A_n)$ and $\mathcal{I}' = (A'_1, A'_2, \dots, A'_n)$ are two sequences of arms. τ_i denotes the number of samples taken on A_i . For any event \mathcal{E} in \mathcal{F}_σ , where σ is an almost-surely finite stopping time with respect to the filtration $\{\mathcal{F}_t\}_{t \geq 0}$, it holds that*

$$\sum_{i=1}^n \mathbb{E}_{\mathbb{A}, \mathcal{I}}[\tau_i] \text{KL}(A_i, A'_i) \geq d \left(\Pr_{\mathbb{A}, \mathcal{I}}[\mathcal{E}], \Pr_{\mathbb{A}, \mathcal{I}'}[\mathcal{E}] \right).$$

3 LOWER BOUND

Throughout our proof of the lower bound, we assume that the reward distributions of all arms are Gaussian distributions with unit variance. Moreover, we assume that the number of arms is sufficiently large. This assumption is used only once in the proof of Lemma 3.3. Note that when there is only a constant number of arms, our lower bound $\Omega(H^{\text{large}} + H^{\text{small}})$ is implied by the $\Omega(H \ln \delta^{-1})$ term.

3.1 Instance Embedding

The following simple lemma is useful in lower bounding the expected number of samples taken from an arm in the top- k set, by restricting to a Best-1-Arm instance embedded in the original Best- k -Arm instance. We postpone its proof to Appendix C.

Lemma 3.1 (Instance Embedding). *Let \mathcal{I} be a Best- k -Arm instance. Let A be an arm among the top k arms, and \mathcal{I}^{emb} be a Best-1-Arm instance consisting of A and a subset of arms in \mathcal{I} outside the top k arms. If some algorithm \mathbb{A} solves \mathcal{I} with probability $1 - \delta$ while taking less than N samples on A in expectation, there exists another algorithm \mathbb{A}^{emb} that solves \mathcal{I}^{emb} with probability $1 - \delta$ while taking less than N samples on A in expectation.*

3.2 Proof of Theorem 1.1

We show a lower bound on the number of samples required by each arm separately, and then the lower bound stated in Theorem 1.1 follows from a direct summation. Formally, we have the following lemma.

Lemma 3.2. *Let \mathcal{I} be an instance of Best- k -Arm. There exist universal constants δ and c such that for all $1 \leq j \leq i$, any δ -correct algorithm for Best- k -Arm takes at least $c\epsilon_j^{-2} \ln |G_{\geq j}^{\text{small}}|$ samples on every arm $A \in G_i^{\text{large}}$. The same holds if we swap G^{large} and G^{small} .*

Before proving Lemma 3.2, we show that Theorem 1.1 follows from Lemma 3.2 directly.

Proof of Theorem 1.1. Since the $\Omega(H \ln \delta^{-1})$ lower bound has been established in Theorem 2 of [CLK⁺14], it remains to show that the sample complexity is lower bounded by both $\Omega(H^{\text{large}})$ and $\Omega(H^{\text{small}})$. Let \mathbb{A} be a δ -correct algorithm for Best- k -Arm. According to Lemma 3.2, \mathbb{A} draws at least $c \cdot \max_{j \leq i} \epsilon_j^{-2} \ln |G_{\geq j}^{\text{small}}|$ samples from each arm in G_i^{large} . Therefore \mathbb{A} draws at least

$$\sum_{i=1}^{\infty} |G_i^{\text{large}}| \cdot c \cdot \max_{j \leq i} \epsilon_j^{-2} \ln |G_{\geq j}^{\text{small}}| = \Omega(H^{\text{large}})$$

samples in total from the arms in G^{large} . The $\Omega(H^{\text{small}})$ lower bound is analogous. \square

3.3 Reduction to Best-1-Arm

In order to prove Lemma 3.2, we construct a Best-1-Arm instance consisting of one arm in G_i^{large} and all arms in $G_{\geq j}^{\text{small}}$. By Instance Embedding (Lemma 3.1), to lower bound the number of samples taken on each arm in G_i^{large} , it suffices to prove that every algorithm for Best-1-Arm takes sufficiently many samples on the best arm. Formally, we would like to show the following key technical lemma.

Lemma 3.3. *Let \mathcal{I} be an instance of Best-1-Arm consisting of one arm with mean μ and n arms with means on $[\mu - \Delta, \mu)$. There exist universal constants δ and c (independent of n and Δ) such that for any algorithm \mathbb{A} that correctly solves \mathcal{I} with probability $1 - \delta$, the expected number of samples drawn from the optimal arm is at least $c\Delta^{-2} \ln n$.*

The proof of Lemma 3.3 is somewhat technical and we present it in the next subsection. Now we prove Lemma 3.2 from Lemma 3.3, by reducing a Best-1-Arm instance to an instance of Best- k -Arm using the Instance Embedding technique. Intuitively, if an algorithm \mathbb{A} solves Best- k -Arm without taking sufficient number of samples from a specific arm, we may extract an instance of Best-1-Arm and derive a contradiction to Lemma 3.3.

Proof of Lemma 3.2. Let δ_0 and c_0 be the constants in Lemma 3.3. We claim that Lemma 3.2 holds for constants $\delta = \delta_0$ and $c = c_0/4$.

Suppose for a contradiction that when δ -correct algorithm \mathbb{A} runs on Best- k -Arm instance \mathcal{I} , the number of samples drawn from arm $A \in G_i^{\text{large}}$ is less than $c\varepsilon_j^{-2} \ln |G_{\geq j}^{\text{small}}|$ for some $j \leq i$.

We construct a Best-1-Arm instance \mathcal{I}^{new} consisting of A and all arms in $G_{\geq j}^{\text{small}}$. By Instance Embedding (Lemma 3.1), there exists algorithm \mathbb{A}^{new} that solves \mathcal{I}^{new} with probability $1 - \delta$, while the number of samples drawn from arm A is upper bounded by $c\varepsilon_j^{-2} \ln |G_{\geq j}^{\text{small}}|$ in expectation.

However, Lemma 3.3 implies that \mathbb{A}^{new} must take more than

$$c_0 \Delta^{-2} \ln n \geq 4c(\varepsilon_i + \varepsilon_j)^{-2} \ln |G_{\geq j}^{\text{small}}| \geq c\varepsilon_j^{-2} \ln |G_{\geq j}^{\text{small}}|$$

samples on the optimal arm, which leads to a contradiction. The case that G^{large} and G^{small} are swapped is analogous. \square

3.4 Reduction to Symmetric Best-1-Arm

In order to prove Lemma 3.3, we first study a special case that the instance consists of one optimal arm and several sub-optimal arms with equal means (we call it a Symmetric Best-1-Arm instance). For the symmetric Best-1-Arm instances, we have the following lower bound on the best arm.

Lemma 3.4. *Let \mathcal{I} be an instance of Best-1-Arm with one arm with mean μ and n arms with mean $\mu - \Delta$. There exist universal constants δ and c (independent of n and Δ) such that for any algorithm \mathbb{A} that correctly solves \mathcal{I} with probability $1 - \delta$, the expected number of samples drawn from the optimal arm is at least $c\Delta^{-2} \ln n$.*

Proof of Lemma 3.4. We claim that the lemma holds for constants $\delta = 0.5$ and $c = 1$.

Recall that $\mathcal{N}(\mu, \sigma^2)$ denotes the normal distribution with mean μ and variance σ^2 . Let \mathcal{I} be the instance consisting of arm A^* with mean μ and n arms with mean $\mu - \Delta$, and \mathcal{I}^{new} be the instance obtained from \mathcal{I} by replacing the reward distribution of A^* with $\mathcal{N}(\mu - \Delta, 1)$. τ denotes the number of samples drawn from A^* .

Let \mathcal{E} be the event that \mathbb{A} identifies arm A^* as the best arm. Recall that $\Pr_{\mathbb{A}, \mathcal{I}}$ and $E_{\mathbb{A}, \mathcal{I}}$ denote the probability and expectation when algorithm \mathbb{A} runs on instance \mathcal{I} respectively. Since \mathbb{A} solves \mathcal{I} correctly with probability at least $1 - \delta$, we have $\Pr_{\mathbb{A}, \mathcal{I}}[\mathcal{E}] \geq 1 - \delta$. On the other hand, \mathcal{I}^{new} consists of $n + 1$ completely identical arms. By Definition 1.4, \mathbb{A} takes a random permutation of \mathcal{I}^{new} as its input. Therefore the probability that \mathbb{A} returns each arm is the same, and it follows that $\Pr_{\mathbb{A}, \mathcal{I}^{\text{new}}}[\mathcal{E}] \leq 1/(n + 1)$.

By Change of Distribution (Lemma 2.1), we have

$$\begin{aligned} \frac{1}{2} E_{\mathbb{A}, \mathcal{I}}[\tau] \Delta^2 &= E_{\mathbb{A}, \mathcal{I}}[\tau] \cdot \text{KL}(\mathcal{N}(\mu, 1), \mathcal{N}(\mu - \Delta, 1)) \\ &\geq d \left(\Pr_{\mathbb{A}, \mathcal{I}}[\mathcal{E}], \Pr_{\mathbb{A}, \mathcal{I}^{\text{new}}}[\mathcal{E}] \right) \\ &\geq d(1 - \delta, 1/(n + 1)) \\ &\geq (1 - \delta) \ln n. \end{aligned}$$

Therefore we conclude that

$$E_{\mathbb{A}, \mathcal{I}}[\tau] \geq 2(1 - \delta) \Delta^{-2} \ln n \geq c \Delta^{-2} \ln n. \quad \square$$

Given Lemma 3.4, Lemma 3.3 may appear to be quite intuitive, as the symmetric instance \mathcal{I}^{sym} seems to be the worst case. However, a rigorous proof of Lemma 3.3 is still quite nontrivial and is in fact the most technical part of the lower bound proof. The proof consists of several steps which transform a general instance \mathcal{I} of Best-1-Arm to a symmetric instance \mathcal{I}^{sym} .

Suppose that some algorithm \mathbb{A} violates Lemma 3.3 on a Best-1-Arm instance \mathcal{I} . We divide the interval $[\mu - \Delta, \mu]$ into $n^{0.9}$ short segments, then at least one segment contains $n^{0.1}$ arms. We construct a smaller and denser instance $\mathcal{I}^{\text{dense}}$ consisting of the optimal arm and $n^{0.1}$ arms from the same segment. By Instance Embedding, there exists algorithm \mathbb{A}^{new} that solves $\mathcal{I}^{\text{dense}}$ while taking few samples on the optimal arm. Note that the reduction crucially relies on the fact that since our lower bound is logarithmic in n , the bound merely shrinks by a constant factor after the number of arms decreases to $n^{0.1}$.

Finally, we transform $\mathcal{I}^{\text{dense}}$ into a symmetric Best-1-Arm instance \mathcal{I}^{sym} consisting of the optimal arm in $\mathcal{I}^{\text{dense}}$ along with $n^{0.1}$ copies of one of the sub-optimal arms. We also define an algorithm \mathbb{A}^{sym} that solves \mathcal{I}^{sym} with few samples drawn from the optimal arm, thus contradicting Lemma 3.4. The full proof of Lemma 3.3 is postponed to Appendix C.

4 UPPER BOUND

4.1 Building Blocks

We start by introducing three subroutines that are useful for building our algorithm for Best- k -Arm.

PAC algorithm for Best- k -Arm. PAC-Best- k is a PAC algorithm for Best- k -Arm adapted from the PAC-SamplePrune algorithm in [CGL16]. PAC-Best- k is guaranteed to partition the given arm set into two sets S^{large} and S^{small} , such that S^{large} approximates the best k arms with high probability.

Lemma 4.1. *PAC-Best- $k(S, k, \varepsilon, \delta)$ takes*

$$O(|S|\varepsilon^{-2} [\ln \delta^{-1} + \ln \min(k, |S| - k)])$$

samples and returns a partition $(S^{\text{large}}, S^{\text{small}})$ of S with $|S^{\text{large}}| = k$ and $|S^{\text{small}}| = |S| - k$. Let $\mu_{[k]}$ and $\mu_{[k+1]}$ denote the k -th and the $(k+1)$ -th largest means in S . With probability $1 - \delta$, it holds that

$$\mu_A \geq \mu_{[k]} - \varepsilon \text{ for all } A \in S^{\text{large}}, \quad (1)$$

$$\mu_A \leq \mu_{[k+1]} + \varepsilon \text{ for all } A \in S^{\text{small}}. \quad (2)$$

Lemma 4.1 is proved in Appendix D. We say that a specific call to PAC-Best- k returns correctly if both (1) and (2) hold.

PAC algorithms for Best-1-Arm. EstMean-Large and EstMean-Small approximate the largest and the smallest mean among several arms respectively. Both algorithms can be easily implemented by calling PAC-Best- k with $k = 1$, and then sampling the best arm identified by PAC-Best- k .

Lemma 4.2. *Both EstMean-Large(S, ε, δ) and EstMean-Small(S, ε, δ) take $O(|S|\varepsilon^{-2} \ln \delta^{-1})$ samples and output a real number. Each of the following inequalities holds with probability $1 - \delta$:*

$$\left| \text{EstMean-Large}(S, \varepsilon, \delta) - \max_{A \in S} \mu_A \right| \leq \varepsilon \quad (3)$$

$$\left| \text{EstMean-Small}(S, \varepsilon, \delta) - \min_{A \in S} \mu_A \right| \leq \varepsilon \quad (4)$$

Lemma 4.2 is proved in Appendix D. We say that a specific call to EstMean-Large (or EstMean-Small) returns correctly if inequality (3) (or (4)) holds.

Elimination procedures. Finally, Elim-Large and Elim-Small are two elimination procedures. Roughly speaking, Elim-Large guarantees that after the elimination, the fraction of arms with means above the larger threshold θ^{large} is bounded by a constant. Meanwhile, a fixed arm with mean below the smaller threshold θ^{small} are unlikely to be eliminated. Analogously, Elim-Small removes arms with means below θ^{small} , and preserves arms above θ^{large} . The properties of Elim-Large and Elim-Small are formally stated below.

Lemma 4.3. *Both Elim-Large($S, \theta^{\text{small}}, \theta^{\text{large}}, \delta$) and Elim-Small($S, \theta^{\text{small}}, \theta^{\text{large}}, \delta$) take $O(|S|\varepsilon^{-2} \ln \delta^{-1})$ samples and return a set $T \subseteq S$. For Elim-Large and a fixed arm $A^* \in S$ with $\mu_{A^*} \leq \theta^{\text{small}}$, it holds with probability $1 - \delta$ that $A^* \in T$ and*

$$|\{A \in T : \mu_A \geq \theta^{\text{large}}\}| \leq |T|/10. \quad (5)$$

Similarly, for Elim-Small and fixed $A^ \in S$ with $\mu_{A^*} \geq \theta^{\text{large}}$, it holds with probability $1 - \delta$ that $A^* \in T$ and*

$$|\{A \in T : \mu_A \leq \theta^{\text{small}}\}| \leq |T|/10. \quad (6)$$

Lemma 4.3 is proved in Appendix D. We say that a call to Elim-Large (or Elim-Small) returns correctly if inequality (5) (or (6)) holds.

4.2 Algorithm

Our algorithm for Best- k -Arm, Bilateral-Elimination, is formally described below. Bilateral-Elimination takes a parameter k , an instance \mathcal{I} of Best- k -Arm and a confidence level δ as input, and returns the best k arms in \mathcal{I} .

Algorithm 1: Bilateral-Elimination

Input: Parameter k , instance \mathcal{I} , and confidence δ .

Output: The best k arms in \mathcal{I} .

```

1  $S_1 \leftarrow \mathcal{I}; T_1 \leftarrow \emptyset;$ 
2 for  $r = 1$  to  $\infty$  do
3    $k_r^{\text{large}} \leftarrow k - |T_r|; k_r^{\text{small}} \leftarrow |S_r| - k_r^{\text{large}};$ 
4   if  $k_r^{\text{large}} = 0$  then return  $T_r;$ 
5   if  $k_r^{\text{small}} = 0$  then return  $T_r \cup S_r;$ 
6    $\delta_r \leftarrow \delta / (20r^2);$ 
7    $(S_r^{\text{large}}, S_r^{\text{small}}) \leftarrow \text{PAC-Best-}k(S_r, k_r^{\text{large}}, \varepsilon_r/8, \delta_r);$ 
8    $\theta_r^{\text{large}} \leftarrow \text{EstMean-Large}(S_r^{\text{small}}, \varepsilon_r/8, \delta_r);$ 
9    $\theta_r^{\text{small}} \leftarrow \text{EstMean-Small}(S_r^{\text{large}}, \varepsilon_r/8, \delta_r);$ 
10   $\delta'_r \leftarrow \delta / \min(k_r^{\text{large}}, k_r^{\text{small}});$ 
11   $S_{r+1} \leftarrow$ 
      $\text{Elim-Large}(S_r^{\text{large}}, \theta_r^{\text{large}} + \varepsilon_r/8, \theta_r^{\text{large}} + \varepsilon_r/4, \delta'_r) \cup$ 
      $\text{Elim-Small}(S_r^{\text{small}}, \theta_r^{\text{small}} - \varepsilon_r/4, \theta_r^{\text{small}} - \varepsilon_r/8, \delta'_r);$ 
12   $T_{r+1} \leftarrow T_r \cup (S_r^{\text{large}} \setminus S_{r+1});$ 

```

Throughout the algorithm, Bilateral-Elimination maintains two sets of arms S_r and T_r for each round r . S_r contains the arms that are still under consideration at the beginning of round r , while T_r denotes the set of arms that have been included in the answer. We say that an arm is removed (or eliminated) at round r , if it is in $S_r \setminus S_{r+1}$. Note that we may remove an arm either because its mean is so small that it cannot be among the best k arms, or its mean is large enough so that we decide to include it in the answer. This justifies the name of our algorithm, Bilateral-Elimination.

In each round r , Bilateral-Elimination performs the following four steps.

Step 1: Initialization. Bilateral-Elimination first calculates k_r^{large} and k_r^{small} , which indicate that it needs to identify the k_r^{large} largest arms (or equivalently, the k_r^{small} smallest arms) in S_r . In the base case that either $k_r^{\text{large}} = 0$ or $k_r^{\text{small}} = 0$, it directly returns the answer.

Step 2: Find a PAC solution. Then Bilateral-Elimination calls PAC-Best- k to partition S_r into S_r^{large} and S_r^{small} with size k_r^{large} and k_r^{small} respectively, such that S_r^{large} denotes an approximation of the best k_r^{large}

arms in S_r .

Step 3: Estimate Thresholds. After that, Bilateral-Elimination calls EstMean-Large and EstMean-Small to compute two thresholds θ_r^{large} and θ_r^{small} . θ_r^{large} is an estimation of the largest mean in S_r^{small} , which is approximately the mean of the $(k_r^{\text{large}} + 1)$ -th largest arm in S_r . Analogously, θ_r^{small} approximates the k_r^{large} -th largest mean in S_r .

It might seem weird at first glance that θ_r^{large} and θ_r^{small} approximates the $(k_r^{\text{large}} + 1)$ -th mean and the k_r^{large} -th mean respectively, implying that θ_r^{large} is expected to be *smaller* than θ_r^{small} . In fact, the superscript “large” in θ_r^{large} indicates that it is the threshold used for eliminating arms in S_r^{large} .

Step 4: Elimination. Finally, Bilateral-Elimination calls Elim-Large and Elim-Small to eliminate the arms in S_r^{large} that are significantly larger than θ_r^{large} , and the arms in S_r^{small} that are much smaller than θ_r^{small} . The arms removed from S_r^{large} are included into the answer.

Caveats. Note that our algorithm uses a different confidence level, δ'_r , in Step 4. Intuitively, at most $\min(k_r^{\text{large}}, k_r^{\text{small}})$ arms among the best k_r^{large} arms in S_r are misclassified as “small arms” by PAC-Best- k . Therefore during the elimination process, it is crucial that such misclassified arms are not mistakenly eliminated. As a result, we need a union bound on these arms, which contributes to the $\min(k_r^{\text{large}}, k_r^{\text{small}})$ factor in our confidence level.

4.3 Observations

We start our analysis of Bilateral-Elimination with a few simple yet useful observations.

Good events. We define $\mathcal{E}_r^{\text{good}}$ as the event that in round r , all the five calls to PAC-Best- k , EstMean, and Elim return correctly. These events are crucial to our following analysis, as they guarantee that the partition $(S_r^{\text{large}}, S_r^{\text{small}})$ and thresholds θ_r^{large} and θ_r^{small} are sufficiently accurate, and additionally, Elim eliminates a sufficiently large fraction of arms. The following observation, due to a simple union bound, lower bounds the probability of each good event.

Observation 4.1. $\Pr[\mathcal{E}_r^{\text{good}}] \geq 1 - 5\delta_r$.

Valid executions. We say that an execution of Bilateral-Elimination is *valid* at round r , if and only if the following two conditions are satisfied:

- For each $1 \leq i < r$, event $\mathcal{E}_i^{\text{good}}$ happens. (i.e., all calls to subroutines return correctly in previous rounds.)
- The union of T_r and the best k_r^{large} arms in S_r is the correct answer of the Best- k -Arm instance.

In other words, no arms have been incorrectly eliminated in previous rounds.

Moreover, an execution is valid if it is valid at every round before it terminates. We define $\mathcal{E}^{\text{valid}}$ to be the event that an execution of Bilateral-Elimination is valid.

Thresholds. In the following, we bound the thresholds θ_r^{large} and θ_r^{small} returned by subroutine EstMean conditioning on $\mathcal{E}_r^{\text{good}}$. Let μ_r^{large} and μ_r^{small} denote the means of the k_r^{large} -th and the $(k_r^{\text{large}} + 1)$ -th largest arms in S_r . We show that θ_r^{large} and θ_r^{small} are $O(\varepsilon_r)$ -approximations of μ_r^{small} and μ_r^{large} conditioning on the good event $\mathcal{E}_r^{\text{good}}$. The proof of the following observation is postponed to Appendix D.

Observation 4.2. Conditioning on $\mathcal{E}_r^{\text{good}}$,

$$\theta_r^{\text{large}} \in [\mu_r^{\text{small}} - \varepsilon_r/8, \mu_r^{\text{small}} + \varepsilon_r/4],$$

$$\theta_r^{\text{small}} \in [\mu_r^{\text{large}} - \varepsilon_r/4, \mu_r^{\text{large}} + \varepsilon_r/8].$$

Number of remaining arms. Finally, we show that conditioning on the validity of an execution, the number of remaining arms at the beginning of each round can be upper bounded in terms of $|G_{\geq r}^{\text{large}}|$ and $|G_{\geq r}^{\text{small}}|$. The following observation, proved in Appendix D, is crucial to analyzing the sample complexity of our algorithm.

Observation 4.3. Conditioning on $\mathcal{E}^{\text{valid}}$, it holds that $k_r^{\text{large}} \leq 2|G_{\geq r}^{\text{large}}|$ and $k_r^{\text{small}} \leq 2|G_{\geq r}^{\text{small}}|$.

4.4 Correctness

Recall that $\mathcal{E}^{\text{valid}}$ denotes the event that the execution of Bilateral-Elimination is valid. The following lemma, proved in Appendix D, shows that event $\mathcal{E}^{\text{valid}}$ happens with high probability.

Lemma 4.4. $\Pr[\mathcal{E}^{\text{valid}}] \geq 1 - \delta$.

We show that Bilateral-Elimination always returns the correct answer conditioning on $\mathcal{E}^{\text{valid}}$, thus proving that Bilateral-Elimination is δ -correct.

Lemma 4.5. *Bilateral-Elimination returns the correct answer with probability at least $1 - \delta$.*

Proof of Lemma 4.5. It suffices to show that conditioning on $\mathcal{E}^{\text{valid}}$, the algorithm always returns the correct answer. In fact, if Bilateral-Elimination terminates at round r , it either returns T_r at Line 4 or returns $T_r \cup S_r$ at Line 5. According to the second property guaranteed by the validity at round r , the answer returned by Bilateral-Elimination must be correct.

It remains to show that Bilateral-Elimination does not run forever. Recall that $\Delta_{[k]} = \mu_{[k]} - \mu_{[k+1]}$ is the gap between the k -th and the $(k + 1)$ -th largest means in the original instance \mathcal{I} . We choose a sufficiently

large r^* that satisfies $\varepsilon_{r^*} < \Delta_{[k]}$. By definition, we have $G_{\geq r^*}^{\text{large}} = G_{\geq r^*}^{\text{small}} = \emptyset$. Then Observation 4.3 implies that $k_{r^*}^{\text{large}} = k_{r^*}^{\text{small}} = 0$, if the algorithm does not terminate before round r^* . Therefore the algorithm either terminates at or before round r^* . This completes the proof. \square

4.5 Sample Complexity

We prove the following Lemma 4.6, which bounds the sample complexity of Bilateral-Elimination conditioning on $\mathcal{E}^{\text{valid}}$. Then Theorem 1.2 directly follows from Lemma 4.5 and Lemma 4.6. The proof of Theorem 1.3 is postponed to the appendix.

Lemma 4.6. *Conditioning on event $\mathcal{E}^{\text{valid}}$, Bilateral-Elimination takes $O(H \ln \delta^{-1} + \tilde{H}^{\text{large}} + \tilde{H}^{\text{small}} + \tilde{H})$ samples.*

Proof of Lemma 4.6. We consider the r -th round of the algorithm. Recall that $k_r^{\text{large}} + k_r^{\text{small}} = |S_r|$. According to Lemmas 4.1 through 4.3, PAC-Best-k takes

$$O(|S_r| \varepsilon_r^{-2} [\ln \delta_r^{-1} + \ln \min(k_r^{\text{large}}, k_r^{\text{small}})]) \quad (7)$$

samples. EstMean-Large and EstMean-Small take

$$O((k_r^{\text{large}} + k_r^{\text{small}}) \varepsilon_r^{-2} \ln \delta_r^{-1}) = O(|S_r| \varepsilon_r^{-2} \ln \delta_r^{-1})$$

samples in total, while Elim-Large and Elim-Small take

$$\begin{aligned} & O(k_r^{\text{large}} \varepsilon_r^{-2} \ln \delta_r'^{-1}) + O(k_r^{\text{small}} \varepsilon_r^{-2} \ln \delta_r'^{-1}) \\ & = O(|S_r| \varepsilon_r^{-2} [\ln \delta_r^{-1} + \ln \min(k_r^{\text{large}}, k_r^{\text{small}})]) \end{aligned}$$

samples conditioning on $\mathcal{E}^{\text{valid}}$. Clearly the sample complexity in round r is dominated by (7).

Simplify and split the sum: By Observation 4.3, conditioning on event $\mathcal{E}^{\text{valid}}$, k_r^{large} and k_r^{small} are bounded by $2|G_{\geq r}^{\text{large}}|$ and $2|G_{\geq r}^{\text{small}}|$ respectively. Thus it suffices to bound the sum of $H_r^{(1)} + H_r^{(2,\text{large})} + H_r^{(2,\text{small})}$, where

$$H_r^{(1)} = (|G_{\geq r}^{\text{large}}| + |G_{\geq r}^{\text{small}}|) \varepsilon_r^{-2} (\ln \delta^{-1} + \ln r),$$

$$H_r^{(2,\text{large})} = \varepsilon_r^{-2} |G_{\geq r}^{\text{large}}| \ln |G_{\geq r}^{\text{small}}|,$$

$$H_r^{(2,\text{small})} = \varepsilon_r^{-2} |G_{\geq r}^{\text{small}}| \ln |G_{\geq r}^{\text{large}}|.$$

In fact, since

$$\ln \delta_r^{-1} = \ln \delta^{-1} + \ln(20r^2) = O(\ln \delta^{-1} + \ln r),$$

the $|S_r| \varepsilon_r^{-2} \ln \delta_r^{-1}$ term in (7) is bounded by $H_r^{(1)}$. Moreover, the $|S_r| \varepsilon_r^{-2} \ln \min(k_r^{\text{large}}, k_r^{\text{small}})$ term is smaller than or equal to

$$\varepsilon_r^{-2} (k_r^{\text{large}} \ln k_r^{\text{small}} + k_r^{\text{small}} \ln k_r^{\text{large}}),$$

and is thus upper bounded by $H_r^{(2,\text{large})} + H_r^{(2,\text{small})}$.

In Appendix D, we show with a straightforward calculation that

$$\sum_{r=1}^{\infty} H_r^{(1)} = O(H \ln \delta^{-1} + \tilde{H}),$$

$$\sum_{r=1}^{\infty} H_r^{(2,\text{large})} = O(\tilde{H}^{\text{large}}), \text{ and}$$

$$\sum_{r=1}^{\infty} H_r^{(2,\text{small})} = O(\tilde{H}^{\text{small}}).$$

Then the lemma directly follows. \square

Finally, we prove our main result on the upper bound side.

Proof of Theorem 1.2. Let

$$T = H \ln \delta^{-1} + \tilde{H} + \tilde{H}^{\text{large}} + \tilde{H}^{\text{small}}.$$

Lemma 4.5 and Lemma 4.6 together imply that conditioning on an event that happens with probability $1 - \delta$, Bilateral-Elimination returns the correct answer and takes $O(T)$ samples. Using the parallel simulation trick in [CL15, Theorem H.5], we can obtain an algorithm which uses $O(T)$ samples in expectation (unconditionally), thus proving Theorem 1.2. \square

References

- [AB10] Jean-Yves Audibert and Sébastien Bubeck. Best arm identification in multi-armed bandits. In *COLT*, 2010.
- [ABC09] Peyman Afshani, Jérémy Barbay, and Timothy M Chan. Instance-optimal geometric algorithms. In *FOCS*, 2009.
- [Bec54] Robert E Bechhofer. A single-sample multiple decision procedure for ranking means of normal populations with known variances. *The Annals of Mathematical Statistics*, pages 16–39, 1954.
- [BWV12] Sébastien Bubeck, Tengyao Wang, and Nitin Viswanathan. Multiple identifications in multi-armed bandits. *arXiv preprint arXiv:1205.3181*, 2012.
- [CGL16] Lijie Chen, Anupam Gupta, and Jian Li. Pure exploration of multi-armed bandit under matroid constraints. In *29th Annual Conference on Learning Theory*, pages 647–669, 2016.

- [CL15] Lijie Chen and Jian Li. On the optimal sample complexity for best arm identification. *arXiv preprint arXiv:1511.03774*, 2015.
- [CL16a] Alexandra Carpentier and Andrea Locatelli. Tight (lower) bounds for the fixed budget best arm identification bandit problem. *arXiv preprint arXiv:1605.09004*, 2016.
- [CL16b] Lijie Chen and Jian Li. Open problem: Best arm identification: Almost instance-wise optimality and the gap entropy conjecture. In *COLT*, 2016.
- [CLK⁺14] Shouyuan Chen, Tian Lin, Irwin King, Michael R Lyu, and Wei Chen. Combinatorial pure exploration of multi-armed bandits. In *NIPS*, pages 379–387, 2014.
- [CLQ16] Lijie Chen, Jian Li, and Mingda Qiao. Towards instance optimal bounds for best arm identification. *arXiv preprint arXiv:1608.06031*, 2016.
- [CLTL15] Wei Cao, Jian Li, Yufei Tao, and Zhize Li. On top-k selection in multi-armed bandits and hidden bipartite graphs. In *NIPS*, pages 1036–1044, 2015.
- [Duc07] John Duchi. Derivations for linear algebra and optimization. *Berkeley, California*, 2007.
- [EDMM02] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *COLT*, pages 255–270. Springer, 2002.
- [EDMM06] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *JMLR*, 7:1079–1105, 2006.
- [GGL12] Victor Gabillon, Mohammad Ghavamzadeh, and Alessandro Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. In *NIPS*, pages 3212–3220, 2012.
- [GGLB11] Victor Gabillon, Mohammad Ghavamzadeh, Alessandro Lazaric, and Sébastien Bubeck. Multi-bandit best arm identification. In *NIPS*, pages 2222–2230, 2011.
- [GK16] Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In *Conference on Learning Theory (COLT)*, 2016.
- [GLG⁺16] Victor Gabillon, Alessandro Lazaric, Mohammad Ghavamzadeh, Ronald Ortner, and Peter Bartlett. Improved learning complexity in combinatorial pure exploration bandits. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 1004–1012, 2016.
- [JMN14] Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sébastien Bubeck. lil'ucb: An optimal exploration algorithm for multi-armed bandits. *COLT*, 2014.
- [KCG15] Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best arm identification in multi-armed bandit models. *The Journal of Machine Learning Research*, 2015.
- [KK13] Emilie Kaufmann and Shivaram Kalyanakrishnan. Information complexity in bandit subset selection. In *COLT*, pages 228–251, 2013.
- [KKS13] Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *ICML*, pages 1238–1246, 2013.
- [KS10] Shivaram Kalyanakrishnan and Peter Stone. Efficient selection of multiple bandit arms: Theory and practice. In *ICML*, pages 511–518, 2010.
- [KTAS12] Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. Pac subset selection in stochastic multi-armed bandits. In *ICML*, pages 655–662, 2012.
- [MT04] Shie Mannor and John N Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *JMLR*, 5:623–648, 2004.
- [Rob85] Herbert Robbins. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pages 169–177. Springer, 1985.
- [SJR16] Max Simchowitz, Kevin Jamieson, and Benjamin Recht. Towards a richer understanding of adaptive sampling in the moderate-confidence regime. 2016.

- [ZCL14] Yuan Zhou, Xi Chen, and Jian Li. Optimal pac multiple arm identification with applications to crowdsourcing. In *ICML*, pages 217–225, 2014.