Proceedings of Machine Learning Research vol 65:1–58, 2017

Towards Instance Optimal Bounds for Best Arm Identification

Lijie Chen[*] CHENLJ13@MAILS.TSINGHUA.EDU.CN
Jian Li[*] LIJIAN83@MAIL.TSINGHUA.EDU.CN
Mingda Qiao[*] QMD14@MAILS.TSINGHUA.EDU.CN

Abstract

In the classical best arm identification (Best-1-Arm) problem, we are given n stochastic bandit arms, each associated with a reward distribution with an unknown mean. Upon each play of an arm, we can get a reward sampled i.i.d. from its reward distribution. We would like to identify the arm with the largest mean with probability at least $1 - \delta$, using as few samples as possible. The problem has a long history and understanding its sample complexity has attracted significant attention since the last decade. However, the optimal sample complexity of the problem is still unknown.

Recently, Chen and Li (2016) made an interesting conjecture, called gap-entropy conjecture, concerning the instance optimal sample complexity of Best-1-Arm. Given a Best-1-Arm instance I (i.e., a set of arms), let $\mu_{[i]}$ denote the ith largest mean and $\Delta_{[i]} = \mu_{[1]} - \mu_{[i]}$ denote the corresponding gap. $H(I) = \sum_{i=2}^{n} \Delta_{[i]}^{-2}$ denotes the complexity of the instance. The gap-entropy conjecture states that for any instance I, $\Omega\left(H(I) \cdot \left(\ln \delta^{-1} + \mathsf{Ent}(I)\right)\right)$ is an instance lower bound, where $\mathsf{Ent}(I)$ is an entropy-like term determined by the gaps, and there is a δ-correct algorithm for Best-1-Arm with sample complexity $O\left(H(I) \cdot \left(\ln \delta^{-1} + \mathsf{Ent}(I)\right) + \Delta_{[2]}^{-2} \ln\ln \Delta_{[2]}^{-1}\right)$. We note that $\Theta\left(\Delta_{[2]}^{-2} \ln\ln \Delta_{[2]}^{-1}\right)$ is necessary and sufficient to solve the two-arm instance with the best and second best arms. If the conjecture is true, we would have a complete understanding of the instance-wise sample complexity of Best-1-Arm (up to constant factors).

In this paper, we make significant progress towards a complete resolution of the gap-entropy conjecture. For the upper bound, we provide a highly nontrivial algorithm which requires

$$O\left(H(I) \cdot \left(\ln \delta^{-1} + \mathsf{Ent}(I)\right) + \Delta_{[2]}^{-2} \ln\ln \Delta_{[2]}^{-1} \mathrm{polylog}(n, \delta^{-1})\right)$$

samples in expectation for any instance I. For the lower bound, we show that for any Gaussian Best-1-Arm instance with gaps of the form 2^{-k}, any δ-correct monotone algorithm requires at least

$$\Omega\left(H(I) \cdot \left(\ln \delta^{-1} + \mathsf{Ent}(I)\right)\right)$$

samples in expectation. Here, a monotone algorithm is one which uses no more samples (in expectation) on I' than on I, if I' is a sub-instance of I obtained by removing some sub-optimal arms.

Keywords: best arm identification, instance optimality, gap-entropy

[*] Institute for Interdisciplinary Information Sciences (IIIS), Tsinghua University, Beijing, China. Research supported in part by the National Basic Research Program of China grants 2015CB358700, 2011CBA00300, 2011CBA00301, and the National NSFC grant 61632016.

1. Introduction

The stochastic multi-armed bandit is one of the most popular and well-studied models for capturing the exploration-exploitation tradeoffs in many application domains. There is a huge body of literature on numerous bandit models from several fields including stochastic control, statistics, operation research, machine learning and theoretical computer science. The basic stochastic multi-armed bandit model consists of n stochastic arms with unknown distributions. One can adaptively take samples from the arms and make decision depending on the objective. Popular objectives include maximizing the cumulative sum of rewards, or minimizing the cumulative regret (see e.g., Cesa-Bianchi and Lugosi (2006); Bubeck et al. (2012)).

In this paper, we study another classical multi-armed bandit model, called *pure exploration* model, where the decision-maker first performs a *pure-exploration phase* by sampling from the arms, and then identifies an optimal (or nearly optimal) arm, which serves as the exploitation phase. The model is motivated by many application domains such as medical trials Robbins (1985); Audibert and Bubeck (2010), communication network Audibert and Bubeck (2010), online advertisement Chen et al. (2014), crowdsourcing Zhou et al. (2014); Cao et al. (2015). The *best arm identification* problem (Best-1-Arm) is the most basic pure exploration problem in stochastic multi-armed bandits. The problem has a long history (first formulated in Bechhofer (1954)) and has attracted significant attention since the last decade Audibert and Bubeck (2010); Even-Dar et al. (2006); Mannor and Tsitsiklis (2004); Jamieson et al. (2014); Karnin et al. (2013); Chen and Li (2015); Carpentier and Locatelli (2016); Garivier and Kaufmann (2016). Now, we formally define the problem and set up some notations.

Definition 1.1 *Best-*1*-Arm: We are given a set of n arms $\{A_1, \ldots, A_n\}$. Arm A_i has a reward distribution $\mathcal{D}_i$ with an unknown mean $\mu_i \in [0,1]$. We assume that all reward distributions are Gaussian distributions with unit variance. Upon each play of A_i, we get a reward sampled i.i.d. from $\mathcal{D}_i$. Our goal is to identify the arm with the largest mean using as few samples as possible. We assume here that the largest mean is strictly larger than the second largest (i.e., $\mu_{[1]} > \mu_{[2]}$) to ensure the uniqueness of the solution, where $\mu_{[i]}$ denotes the ith largest mean.*

Remark 1.2 *Some previous algorithms for Best-*1*-Arm take a sequence (instead of a set) of n arms as input. In this case, we may simply assume that the algorithm randomly permutes the sequence at the beginning. Thus the algorithm will have the same behaviour on two different orderings of the same set of arms.*

Remark 1.3 *For the upper bound, everything proved in this paper also holds if the distributions are 1-sub-Gaussian, which is a standard assumption in the bandit literature. On the lower bound side, we need to assume that the distributions are from some family parametrized by the means and satisfy certain properties. See Remark D.4. Otherwise, it is possible to distinguish two distributions using 1 sample even if their means are very close. We cannot hope for a nontrivial lower bound in such generality.*

The Best-1-Arm problem for Gaussian arms was first formulated in Bechhofer (1954). Most early works on Best-1-Arm did not analyze the sample complexity of the algorithms (they proved their algorithms are δ-correct though). The early advances are summarized in the monograph Bechhofer et al. (1968).

For the past two decades, significant research efforts have been devoted to understanding the optimal sample complexity of the Best-1-Arm problem. On the lower bound side, Mannor and Tsitsiklis (2004) proved that any δ-correct algorithm for Best-1-Arm takes $\Omega(\sum_{i=2}^{n} \Delta_{[i]}^{-2} \ln \delta^{-1})$ samples in expectation. In fact, their result is an instance-wise lower bound (see Definition 1.6). Kaufmann et al. (2015) also provided an $\Omega(\sum_{i=2}^{n} \Delta_{[i]}^{-2} \ln \delta^{-1})$ lower bound for Best-1-Arm, which improved the constant factor in Mannor and Tsitsiklis (2004). Garivier and Kaufmann (2016) focused on the asymptotic sample complexity of Best-1-Arm as the confidence level δ approaches zero (treating the gaps as fixed), and obtained a complete resolution of this case (even for the leading constant).[1] Chen and Li (2015) showed that for each n there exists a Best-1-Arm instance with n arms that require $\Omega\left(\sum_{i=2}^{n} \Delta_{[i]}^{-2} \ln \ln n\right)$ samples, which further refines the lower bound.

The algorithms for Best-1-Arm have also been significantly improved in the last two decades Even-Dar et al. (2002); Gabillon et al. (2012); Kalyanakrishnan et al. (2012); Karnin et al. (2013); Jamieson et al. (2014); Chen and Li (2015); Garivier and Kaufmann (2016). Karnin et al. (2013) obtained an upper bound of

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

The same upper bound was obtained by Jamieson et al. (2014) using a UCB-type algorithm called lil'UCB. Recently, the upper bound was improved to

$$O\left(\Delta_{[2]}^{-2} \ln \ln \Delta_{[2]}^{-1} + \sum_{i=2}^{n} \Delta_{[i]}^{-2} \left(\ln \ln \min(\Delta_{[i]}^{-1}, n) + \ln \delta^{-1}\right)\right)$$

by Chen and Li (2015). There is still a gap between the best known upper and lower bound.

To understand the sample complexity of Best-1-Arm, it is important to study a special case, which we term as SIGN-ξ. The problem can be viewed as a special case of Best-1-Arm where there are only two arms, and we know the mean of one arm. SIGN-ξ will play a very important role in our lower bound proof.

Definition 1.4 *SIGN-ξ: ξ is a fixed constant. We are given a single arm with unknown mean $\mu \neq \xi$. The goal is to decide whether $\mu > \xi$ or $\mu < \xi$. Here, the gap of the problem is defined to be $\Delta = |\mu - \xi|$. Again, we assume that the distribution of the arm is a Gaussian distribution with unit variance.*

In this paper, we are interested in algorithms (either for Best-1-Arm or for SIGN-ξ) that can identify the correct answer with probability at least $1 - \delta$. This is often called the *fixed confidence* setting in the bandit literature.

Definition 1.5 *For any $\delta \in (0, 1)$, we say that an algorithm $\mathbb{A}$ for Best-1-Arm (or SIGN-ξ) is δ-correct, if on any Best-1-Arm (or SIGN-ξ) instance, $\mathbb{A}$ returns the correct answer with probability at least $1 - \delta$.*

1.1. Almost Instance-wise Optimality Conjecture

It is easy to see that no function $f(n, \delta)$ (only depending on n and δ) can serve as an upper bound of the sample complexity of Best-1-Arm (with n arms and confidence level $1 - \delta$). Instead,

[1] In contrast, our work focus on the situation that both δ and all gaps are variables that tend to zero. In fact, if we let the gaps (i.e., $\Delta_{[i]}$'s) tend to 0 while maintaining δ fixed, their lower bound is not tight.

the sample complexity depends on the gaps. Intuitively, the smaller the gaps are, the harder the instance is (i.e., more samples are required). Since the gaps completely determine an instance (for Gaussian arms with unit variance, up to shifting), we use $\Delta_{[i]}$'s as the parameters to measure the sample complexity.

Now, we formally define the notion of instance-wise lower bounds and instance optimality.For algorithm $\mathbb{A}$ and instance I, we use $T_{\mathbb{A}}(I)$ to denote the expected number of samples taken by $\mathbb{A}$ on instance I.

Definition 1.6 (Instance-wise Lower Bound)

For a Best-1-Arm instance I and a confidence level δ, we define the instance-wise lower bound of I as

$$\mathcal{L}(I,\delta) := \inf_{\mathbb{A}:\mathbb{A} \text{ is } \delta\text{-correct for Best-1-Arm}} T_{\mathbb{A}}(I).$$

We say a Best-1-Arm algorithm $\mathbb{A}$ is instance optimal, if it is δ-correct, and for every instance I, $T_{\mathbb{A}}(I) = O(\mathcal{L}(I,\delta))$.

Now, we consider the Best-1-Arm problem from the perspective of instance optimality. Unfortunately, even for the two-arm case, no instance optimal algorithm may exist. In fact, Farrell (1964) showed that for any δ-correct algorithm $\mathbb{A}$ for SIGN-ξ, we must have

$$\liminf_{\Delta\to 0} \frac{T_{\mathbb{A}}(I)}{\Delta^{-2}\ln\ln\Delta^{-1}} = \Omega(1).$$

This implies that any δ-correct algorithm requires $\Delta^{-2}\ln\ln\Delta^{-1}$ samples in the worst case. Hence, the upper bound of $\Delta^{-2}\ln\ln\Delta^{-1}$ for SIGN-ξ is generally not improvable. However, for a particular SIGN-ξ instance I_Δ with gap Δ, there is an δ-correct algorithm that only needs $O(\Delta^{-2}\ln\delta^{-1})$ samples for this instance, implying $\mathcal{L}(I_\Delta,\delta) = \Theta(\Delta^{-2}\ln\delta^{-1})$. See Chen and Li (2015) for details.

Despite the above fact, Chen and Li (2016) conjectured that the two-arm case is the *only* obstruction toward an instance optimal algorithm. Moreover, based on some evidence from the previous work Chen and Li (2015), they provided an explicit formula and conjecture that $\mathcal{L}(I,\delta)$ can be expressed by the formula. Interestingly, the formula involves an entropy term (similar entropy terms also appear in Afshani et al. (2009) for completely different problems). In order to state Chen and Li's conjecture formally, we define the entropy term first.

Definition 1.7 *Given a Best-1-Arm instance I and $k \in \mathbb{N}$, let*

$$G_k = \{i \in [2,n] \mid 2^{-(k+1)} < \Delta_{[i]} \le 2^{-k}\}, \quad H_k = \sum\nolimits_{i\in G_k} \Delta_{[i]}^{-2}, \quad \textit{and} \quad p_k = H_k / \sum\nolimits_j H_j.$$

*We can view $\{p_k\}$ as a discrete probability distribution. We define the following quantity as the **gap entropy** of instance I:*

$$\mathsf{Ent}(I) = \sum\nolimits_{k\in\mathbb{N}:G_k\neq\emptyset} p_k \ln p_k^{-1}.[2]$$

Remark 1.8 *We choose to partition the arms based on the powers of 2. There is nothing special about the constant 2, and replacing it by any other constant only changes $\mathsf{Ent}(I)$ by a constant factor.*

[2] Note that it is exactly the Shannon entropy for the distribution defined by $\{p_k\}$.

Conjecture 1.9 (Gap-Entropy Conjecture (Chen and Li, 2016)) *There is an algorithm for Best-1-Arm with sample complexity*

$$O\left(\mathcal{L}(I,\delta) + \Delta_{[2]}^{-2}\ln\ln\Delta_{[2]}^{-1}\right),$$

for any instance I and $\delta < 0.01$. And we say such an algorithm is almost instance-wise optimal for Best-1-Arm. Moreover,

$$\mathcal{L}(I,\delta) = \Theta\left(\sum\nolimits_{i=2}^{n}\Delta_{[i]}^{-2}\cdot\left(\ln\delta^{-1} + \mathsf{Ent}(I)\right)\right).$$

Remark 1.10 *As we mentioned before, the term $\Delta^{-2}\ln\ln\Delta^{-1}$ is sufficient and necessary for distinguishing the best and the second best arm, even though it is not an instance-optimal bound. The gap entropy conjecture states that modulo this additive term, we can obtain an instance optimal algorithm. Hence, the resolution of the conjecture would provide a complete understanding of the sample complexity of Best-1-Arm (up to constant factors). All the previous bounds for Best-1-Arm agree with Conjecture 1.9, i.e., existing upper (lower) bounds are no smaller (larger) the conjectured bound. See Chen and Li (2016) for details.*

1.2. Our Results

In this paper, we make significant progress toward the resolution of the gap-entropy conjecture. On the upper bound side, we provide an algorithm that almost matches the conjecture.

Theorem 1.11 *There is a δ-correct algorithm for Best-1-Arm with expected sample complexity*

$$O\left(\sum\nolimits_{i=2}^{n}\Delta_{[i]}^{-2}\cdot\left(\ln\delta^{-1} + \mathsf{Ent}(I)\right) + \Delta_{[2]}^{-2}\ln\ln\Delta_{[2]}^{-1}\cdot\mathrm{polylog}(n,\delta^{-1})\right).$$

Our algorithm matches the main term $\sum_{i=2}^{n}\Delta_{[i]}^{-2}\cdot\left(\ln\delta^{-1} + \mathsf{Ent}(I)\right)$ in Conjecture 1.9. For the additive term (which is typically small), we lose a $\mathrm{polylog}(n,\delta^{-1})$ factor. In particular, for those instances where the additive term is $\mathrm{polylog}(n,\delta^{-1})$ times smaller than the main term, our algorithm is optimal.

On the lower bound side, despite that we are not able to completely solve the lower bound, we do obtain a rather strong bound. We need to introduce some notations first. We say an instance is *discrete*, if the gaps of all the sub-optimal arms are of the form 2^{-k} for some positive integer k. We say an instance I' is a *sub-instance* of an instance I, if I' can be obtained by deleting some *sub-optimal* arms from I. Formally, we have the following theorem.

Theorem 1.12 *For any discrete instance I, confidence level $\delta < 0.01$, and any δ-correct algorithm $\mathbb{A}$ for Best-1-Arm, there exists a sub-instance I' of I such that*

$$T_{\mathbb{A}}(I') \geq c\cdot\left(\sum\nolimits_{i=2}^{n}\Delta_{[i]}^{-2}\cdot\left(\ln\delta^{-1} + \mathsf{Ent}(I)\right)\right),$$

where c is a universal constant.

We say an algorithm is *monotone*, if $T_{\mathbb{A}}(I') \leq T_{\mathbb{A}}(I)$ for every I' and I such that I' is a sub-instance of I. Then we immediately have the following corollary.

Corollary 1.13 *For any discrete instance I, and confidence level $\delta < 0.01$, for any monotone δ-correct algorithm $\mathbb{A}$ for Best-1-Arm, we have that*

$$T_{\mathbb{A}}(I) \geq c \cdot \left(\sum\nolimits_{i=2}^{n} \Delta_{[i]}^{-2} \cdot \left(\ln \delta^{-1} + \mathsf{Ent}(I)\right)\right),$$

where c is a universal constant.

We remark that all previous algorithms for Best-1-Arm have monotone sample complexity bounds. The above corollary also implies that if an algorithm has a monotone sample complexity bound, then the bound must be $\Omega\left(\sum\nolimits_{i=2}^{n} \Delta_{[i]}^{-2} \cdot \left(\ln \delta^{-1} + \mathsf{Ent}(I)\right)\right)$ on all discrete instances.

2. Related Work

SIGN-ξ and A/B testing. In the A/B testing problem, we are asked to decide which arm between the two given arms has the larger mean. A/B testing is in fact equivalent to the SIGN-ξ problem. It is easy to reduce SIGN-ξ to A/B testing by constructing a fictitious arm with mean ξ. For the other direction, given an instance of A/B testing, we may define an arm as the difference between the two given arms and the problem reduces to SIGN-ξ where $\xi = 0$. In particular, our refined lower bound for SIGN-ξ stated in Lemma 4.1 also holds for A/B testing. Kaufmann et al. (2015); Garivier and Kaufmann (2016) studied the limiting behavior of the sample complexity of A/B testing as the confidence level δ approaches to zero. In contrast, we focus on the case that both δ and the gap Δ tend to zero, so that the complexity term due to not knowing the gap in advance will not be dominated by the $\ln \delta^{-1}$ term.

Best-k-Arm. The Best-k-Arm problem, in which we are required to identify the k arms with the k largest means, is a natural extension of Best-1-Arm. Best-k-Arm has been extensively studied in the past few years Kalyanakrishnan and Stone (2010); Gabillon et al. (2011, 2012); Kalyanakrishnan et al. (2012); Bubeck et al. (2013); Kaufmann and Kalyanakrishnan (2013); Zhou et al. (2014); Kaufmann et al. (2015); Chen et al. (2017), and most results for Best-k-Arm are generalizations of those for Best-1-Arm. As in the case of Best-1-Arm, the sample complexity bounds of Best-k-Arm depend on the gap parameters of the arms, yet the gap of an arm is typically defined as the distance from its mean to either $\mu_{[k+1]}$ or $\mu_{[k]}$ (depending on whether the arm is among the best k arms or not) in the context of Best-k-Arm problem. The *Combinatorial Pure Exploration* problem, which further generalizes the cardinality constraint in Best-k-Arm (i.e., to choose exactly k arms) to general combinatorial constraints, was also studied Chen et al. (2014, 2016); Gabillon et al. (2016).

PAC learning. The sample complexity of Best-1-Arm and Best-k-Arm in the probably approximately correct (PAC) setting has also been well studied in the past two decades. For Best-1-Arm, the tight *worst-case* sample complexity bound was obtained by Even-Dar et al. (2002); Mannor and Tsitsiklis (2004); Even-Dar et al. (2006). Kalyanakrishnan and Stone (2010); Kalyanakrishnan et al. (2012); Zhou et al. (2014); Cao et al. (2015) also studied the worst case sample complexity of Best-k-Arm in the PAC setting.

3. Preliminaries

Throughout the paper, I denotes an instance of Best-1-Arm (i.e., I is a set of arms). The arm with the largest mean in I is called the optimal arm, while all other arms are *sub-optimal*. We assume that every instance has a unique optimal arm. A_i denotes the arm in I with the i-th largest mean, unless stated otherwise. The mean of an arm A is denoted by μ_A, and we use $\mu_{[i]}$ as a shorthand notation for μ_{A_i} (i.e., the i-th largest mean in an instance). Define $\Delta_A = \mu_{[1]} - \mu_A$ as the gap of arm A, and let $\Delta_{[i]} = \Delta_{A_i}$ denote the gap of arm A_i. We assume that $\Delta_{[2]} > 0$ to ensure the optimal arm is unique.

We partition the sub-optimal arms into different groups based on their gaps. For each $k \in \mathbb{N}$, group G_k is defined as $\left\{A_i : \Delta_{[i]} \in \left(2^{-(k+1)}, 2^{-k}\right]\right\}$. For brevity, let $G_{\geq k}$ and $G_{\leq k}$ denoted $\bigcup_{i=k}^{\infty} G_i$ and $\bigcup_{i=1}^{k} G_i$ respectively. The *complexity* of arm A_i is defined as $\Delta_{[i]}^{-2}$, while the complexity of instance I is denoted by $H(I) = \sum_{i=2}^{n} \Delta_{[i]}^{-2}$ (or simply H, if the instance is clear from the context). Moreover, $H_k = \sum_{A \in G_k} \Delta_A^{-2}$ denotes the total complexity of the arms in group G_k. $(H_k)_{k=1}^{\infty}$ naturally defines a probability distribution on $\mathbb{N}$, where the probability of k is given by $p_k = H_k/H$. The gap-entropy of the instance I is then denoted by

$$\mathsf{Ent}(I) = \sum_k p_k \ln p_k^{-1}.$$

Here and in the following, we adopt the convention that $0 \ln 0^{-1} = 0$.

4. A Sketch of the Lower Bound

4.1. A Comparison with Previous Lower Bound Techniques

We briefly discuss the novelty of our new lower bound technique, and argue why the previous techniques are not sufficient to obtain our result. To obtain a lower bound on the sample complexity of Best-1-Arm, all the previous work Mannor and Tsitsiklis (2004); Chen et al. (2014); Kaufmann et al. (2015); Garivier and Kaufmann (2016) are based on creating two similar instances with different answers, and then applying the *change of distribution* method (originally developed in Kaufmann et al. (2015)) to argue that a certain number of samples are necessary to distinguish such two instances. The idea was further refined by Garivier and Kaufmann (2016). They formulated a max-min game between the algorithm and some instances (with different answers than the given instance) created by an adversary. The value of the game at equilibrium would be a lower bound of the samples one requires to distinguish the current instance and several worst adversary instances. However, we notice that even in the two-arm case, one cannot prove the $\Omega(\Delta^{-2} \ln\ln \Delta^{-1})$ lower bound by considering only one max-min game to distinguish the current instance from other instance. Roughly speaking, the $\ln\ln \Delta^{-1}$ factor is due to not knowing the actual gap Δ, and any lower bound that can bring out the $\ln\ln \Delta^{-1}$ factor should reflect the union bound paid for the uncertainty of the instance. In fact, for the Best-1-Arm problem with n arms, the gap entropy $\mathsf{Ent}(I)$ term exists for a similar reason (not knowing the gaps). Hence, any lower bound proof for Best-1-Arm that can bring out the $\mathsf{Ent}(I)$ term necessarily has to consider the uncertainty of current instance as well (in fact, the random permutation of all arms is the kind of uncertainty we need for the new lower bound). In our actual lower bound proof, we first obtain a very tight understanding of the SIGN-ξ problem (Lemma 4.1).[3] Then, we provide an elegant reduction from SIGN-ξ to Best-1-Arm, by embedding the SIGN-ξ problem to a collection of Best-1-Arm instances.

[3] Farrell's lower bound Farrell (1964) is not sufficient for our purpose.

4.2. Proof of Theorem 1.12

Following the approach in Chen and Li (2015), we establish the lower bound by a reduction from SIGN-ξ to discrete Best-1-Arm instances, together with a more refined lower bound for SIGN-ξ stated in the following lemma.

Lemma 4.1 *Suppose $\delta \in (0, 0.04)$, $m \in \mathbb{N}$ and $\mathbb{A}$ is a δ-correct algorithm for SIGN-ξ. P is a probability distribution on $\{2^{-1}, 2^{-2}, \dots, 2^{-m}\}$ defined by $P(2^{-k}) = p_k$. $\mathsf{Ent}(P)$ denotes the Shannon entropy of distribution P. Let $T_{\mathbb{A}}(\mu)$ denote the expected number of samples taken by $\mathbb{A}$ when it runs on an arm with distribution $\mathcal{N}(\mu, 1)$ and $\xi = 0$. Define $\alpha_k = T_{\mathbb{A}}(2^{-k})/4^k$. Then,*

$$\sum_{k=1}^{m} p_k \alpha_k = \Omega(\mathsf{Ent}(P) + \ln \delta^{-1}).$$

It is well known that to distinguish the normal distribution $\mathcal{N}(2^{-k}, 1)$ from $\mathcal{N}(-2^{-k}, 1)$, $\Omega(4^k)$ samples are required. Thus, $\alpha_k = T_{\mathbb{A}}(2^{-k})/4^k$ denotes the ratio between the expected number of samples taken by $\mathbb{A}$ and the corresponding lower bound, which measures the "loss" due to not knowing the gap in advance. Then Lemma 4.1 can be interpreted as follows: when the gap is drawn from a distribution P, the *expected loss* is lower bounded by the sum of the entropy of P and $\ln \delta^{-1}$. We defer the proof of Lemma 4.1 to Appendix D.

Now we prove Theorem 1.12 by applying Lemma 4.1 and an elegant reduction from SIGN-ξ to Best-1-Arm.

Proof [Proof of Theorem 1.12] Let c_0 be the hidden constant in the big-Ω in Lemma 4.1, i.e.,

$$\sum_{k=1}^{m} p_k \alpha_k \ge c_0 \cdot (\mathsf{Ent}(P) + \ln \delta^{-1}).$$

We claim that Theorem 1.12 holds for constant $c = 0.25c_0$.

Suppose towards a contradiction that $\mathbb{A}$ is a δ-correct (for some $\delta < 0.01$) algorithm for Best-1-Arm and $I = \{A_1, A_2, \dots, A_n\}$ is a discrete instance, while for all sub-instance I' of I,

$$T_{\mathbb{A}}(I') < c \cdot H(I)(\mathsf{Ent}(I) + \ln \delta^{-1}).$$

Recall that $H(I)$ and $\mathsf{Ent}(I)$ denote the complexity and entropy of instance I, respectively.

Construct a distribution of SIGN-ξ instances. Let n_k be the number of arms in I with gap 2^{-k}, and m be the greatest integer such that $n_m > 0$. Since I is discrete, the complexity of instance I is given by

$$H(I) = \sum_{k=1}^{m} 4^k n_k.$$

Let $p_k = 4^k n_k / H(I)$. Then $(p_k)_{k=1}^{m}$ defines a distribution P on $\{2^{-1}, 2^{-2}, \dots, 2^{-m}\}$. Moreover, the Shannon entropy of distribution P is exactly the entropy of instance I, i.e., $\mathsf{Ent}(P) = \mathsf{Ent}(I)$. Our goal is to construct an algorithm for SIGN-ξ that violates Lemma 4.1 on distribution P.

A family of sub-instances of I. Let $U = \{k \in [m] : n_k > 0\}$ be the set of "types" of arms that are present in I. We consider the following family of instances obtained from I. For $S \subseteq U$, define I_S as the instance obtained from I by removing exactly one arm of gap 2^{-k} for each $k \in S$. Note that I_S is a sub-instance of I.

Let $\overline{S}$ denote $U \setminus S$, the complement of set S relative to U. For $S \subseteq U$ and $k \in \overline{S}$, let τ_k^S denote the expected number of samples taken on all the n_k arms with gap 2^{-k} when $\mathbb{A}$ runs on I_S. Define $\alpha_k^S = 4^{-k}\tau_k^S/n_k$. We note that $4^k\alpha_k^S$ is the expected number of samples taken on *every* arm with gap 2^{-k} in instance I_S.[4]

We have the following inequality:

$$\sum_{S\subseteq U}\sum_{k\in\overline{S}} 4^k n_k \alpha_k^S = \sum_{S\subseteq U}\sum_{k\in\overline{S}} \tau_k^S \le \sum_{S\subseteq U} T_{\mathbb{A}}(I_S) < c \cdot 2^{|U|} H(I)(\mathsf{Ent}(I) + \ln\delta^{-1}). \tag{1}$$

The second step holds because the lefthand side only counts part of the samples taken by $\mathbb{A}$. The last step follows from our assumption and the fact that I_S is a sub-instance of I.

Construct algorithm $\mathbb{A}^{\mathsf{new}}$ from $\mathbb{A}$. Now we define an algorithm $\mathbb{A}^{\mathsf{new}}$ for SIGN-ξ with $\xi = 0$. Given an arm A, we first choose a set $S \subseteq U$ uniformly at random from all subsets of U. Recall that $\mu_{[1]}$ denotes the mean of the optimal arm in I. $\mathbb{A}^{\mathsf{new}}$ runs the following four algorithms $\mathbb{A}_1$ through $\mathbb{A}_4$ in parallel:

1. Algorithm $\mathbb{A}_1$ simulates $\mathbb{A}$ on $I_S \cup \{\mu_{[1]} + A\}$.
2. Algorithm $\mathbb{A}_2$ simulates $\mathbb{A}$ on $I_{\overline{S}} \cup \{\mu_{[1]} + A\}$.
3. Algorithm $\mathbb{A}_3$ simulates $\mathbb{A}$ on $I_S \cup \{\mu_{[1]} - A\}$.
4. Algorithm $\mathbb{A}_4$ simulates $\mathbb{A}$ on $I_{\overline{S}} \cup \{\mu_{[1]} - A\}$.

More precisely, when one of the four algorithms requires a new sample from $\mu_{[1]} + A$ (or $\mu_{[1]} - A$), we draw a sample x from arm A, feed $\mu_{[1]} + x$ to $\mathbb{A}_1$ and $\mathbb{A}_2$, and then feed $\mu_{[1]} - x$ to $\mathbb{A}_3$ and $\mathbb{A}_4$. Note that the samples taken by the four algorithms are the same up to negation and shifting.

$\mathbb{A}^{\mathsf{new}}$ terminates as soon as one of the four algorithms terminates. If one of $\mathbb{A}_1$ and $\mathbb{A}_2$ identifies $\mu_{[1]} + A$ as the optimal arm, or one of $\mathbb{A}_3$ and $\mathbb{A}_4$ identifies an arm other than $\mu_{[1]} - A$ as the optimal arm, $\mathbb{A}^{\mathsf{new}}$ outputs "$\mu_A > 0$"; otherwise it outputs "$\mu_A < 0$".

Clearly, $\mathbb{A}^{\mathsf{new}}$ is correct if all of $\mathbb{A}_1$ through $\mathbb{A}_4$ are correct, which happens with probability at least $1 - 4\delta$. Note that since $4\delta < 0.04$, the condition of Lemma 4.1 is satisfied.

Upper bound the sample complexity of $\mathbb{A}^{\mathsf{new}}$. The crucial observation is that when $\mu_A = -2^{-k}$ and $k \in S$, $\mathbb{A}_1$ effectively simulates the execution of $\mathbb{A}$ on $I_{S\setminus\{k\}}$. In fact, since all arms are Gaussian distributions with unit variance, the arm $\mu_{[1]} + A$ is the same as an arm with gap 2^{-k} in the original Best-1-Arm instance. Recall that the number of samples taken on each of the arms with gap 2^{-k} in instance $I_{S\setminus\{k\}}$ is $4^k\alpha_k^{S\setminus\{k\}}$. Therefore, the expected number of samples taken on A is upper bounded by $4^k\alpha_k^{S\setminus\{k\}}$.[5] Likewise, when $\mu_A = -2^{-k}$ and $k \in \overline{S}$, $\mathbb{A}_2$ is equivalent to the

[4] Recall that a Best-1-Arm algorithm is defined on a *set* of arms, so the arms with identical means in the instance cannot be distinguished by $\mathbb{A}$. See Remark 1.2 for details.

[5] Recall that if $\mathbb{A}_1$ terminates after taking T samples from $\mu_{[1]} + A$, the number of samples taken by $\mathbb{A}^{\mathsf{new}}$ on A is also T (rather than $4T$).

execution of $\mathbb{A}$ on $I_{\overline{S}\setminus\{k\}}$, and thus the expected number of samples on A is less than or equal to $4^k\alpha_k^{\overline{S}\setminus\{k\}}$. Analogous claims hold for the case $\mu_A = +2^{-k}$ and algorithms $\mathbb{A}_3$ and $\mathbb{A}_4$ as well.

It remains to compute the expected loss of $\mathbb{A}^{\mathsf{new}}$ on distribution P and derive a contradiction to Lemma 4.1. It follows from a simple calculation that

$$\begin{aligned}
\sum_{k=1}^{m} p_k\alpha_k &\le \sum_{k\in U} p_k \cdot \frac{1}{2^{|U|}}\left(\sum_{S\subseteq U:k\in S}\alpha_k^{S\setminus\{k\}} + \sum_{S\subseteq U:k\in\overline{S}}\alpha_k^{\overline{S}\setminus\{k\}}\right)\\
&= \frac{1}{2^{|U|-1}}\sum_{k\in U}\sum_{S\subseteq U:k\in S} p_k\alpha_k^{S\setminus\{k\}}\\
&= \frac{1}{2^{|U|-1}}\sum_{S\subseteq U}\sum_{k\in\overline{S}}\frac{4^k n_k}{H(I)}\cdot\alpha_k^{S}\\
&\le \frac{2^{|U|}}{2^{|U|-1}}\cdot c\cdot(\mathsf{Ent}(I)+\ln\delta^{-1}) < c_0(\mathsf{Ent}(P)+\ln(4\delta)^{-1}).
\end{aligned}$$

The first step follows from our discussion on algorithm $\mathbb{A}^{\mathsf{new}}$. The third step renames the variables and rearranges the summation. The last line applies (1). This leads to a contradiction to Lemma 4.1 and thus finishes the proof. ■

5. Warmup: Best-1-Arm with Known Complexity

To illustrate the idea of our algorithm for Best-1-Arm, we consider the following simplified yet still non-trivial version of Best-1-Arm: the complexity of the instance, $H(I) = \sum_{i=2}^{n}\Delta_{[i]}^{-2}$, is given, yet the means of the arms are still unknown.

5.1. Building Blocks

We introduce some subroutines that are used throughout our algorithm.

Uniform sampling. The first building block is a uniform sampling procedure, $\mathsf{Unif\text{-}Sampl}(S,\varepsilon,\delta)$, which takes $2\varepsilon^{-2}\ln(2/\delta)$ samples from each arm in set S. Let $\hat{\mu}_A$ be the empirical mean of arm A (i.e., the average of all sampled values from A). It obtains an ε-approximation of the mean of each arm with probability $1-\delta$. The following fact directly follows by the Chernoff bound.

Fact 5.1 *$\mathsf{Unif\text{-}Sampl}(S,\varepsilon,\delta)$ takes $O(|S|\varepsilon^{-2}\ln\delta^{-1})$ samples. For each arm $A\in S$, we have*

$$\Pr\left[|\hat{\mu}_A-\mu_A|\le\varepsilon\right]\ge 1-\delta.$$

We say that a call to procedure $\mathsf{Unif\text{-}Sampl}(S,\varepsilon,\delta)$ returns correctly, if $|\hat{\mu}_A-\mu_A|\le\varepsilon$ holds for every arm $A\in S$. Fact 5.1 implies that when $|S|=1$, the probability of returning correctly is at least $1-\delta$.

Median elimination. Even-Dar et al. (2002) introduced the Median Elimination algorithm for the PAC version of Best-1-Arm. Med-Elim(S, ε, δ) returns an arm in S with mean at most ε away from the largest mean. Let $\mu_{[1]}(S)$ denote the largest mean among all arms in S. The performance guarantees of Med-Elim is formally stated in the next fact.

Fact 5.2 *Med-Elim(S, ε, δ) takes $O(|S|\varepsilon^{-2} \ln \delta^{-1})$ samples. Let A be the arm returned by Med-Elim. Then*

$$\Pr[\mu_A \geq \mu_{[1]}(S) - \varepsilon] \geq 1 - \delta.$$

We say that Med-Elim(S, ε, δ) returns correctly, if it holds that $\mu_A \geq \mu_{[1]}(S) - \varepsilon$.

Fraction test. Procedure Frac-Test$(S, c^{\text{low}}, c^{\text{high}}, \theta^{\text{low}}, \theta^{\text{high}}, \delta)$ decides whether a sufficiently large fraction (compared to thresholds θ^{low} and θ^{high}) of arms in S have small means (compared to thresholds c^{low} and c^{high}). The procedure randomly samples a certain number of arms from S and estimates their means using Unif-Sampl. Then it compares the fraction of arms with small means to the thresholds and returns an answer accordingly. The detailed implementation of Frac-Test is relegated to Appendix A, where we also prove the following fact.

Fact 5.3 *Frac-Test$(S, c^{\text{low}}, c^{\text{high}}, \theta^{\text{low}}, \theta^{\text{high}}, \delta)$ takes $O\left((\varepsilon^{-2} \ln \delta^{-1}) \cdot (\Delta^{-2} \ln \Delta^{-1})\right)$ samples, where $\varepsilon = c^{\text{high}} - c^{\text{low}}$ and $\Delta = \theta^{\text{high}} - \theta^{\text{low}}$. With probability $1 - \delta$, the following two claims hold simultaneously:*

- *If Frac-Test returns True, $|\{A \in S : \mu_A < c^{\text{high}}\}| > \theta^{\text{low}}|S|$.*
- *If Frac-Test returns False, $|\{A \in S : \mu_A < c^{\text{low}}\}| < \theta^{\text{high}}|S|$.*

We say that a call to procedure Frac-Test returns correctly, if both the two claims above hold; otherwise the call fails.

Elimination. Finally, procedure Elimination$(S, d^{\text{low}}, d^{\text{high}}, \delta)$ eliminates the arms with means smaller than threshold d^{low} from S. More precisely, the procedure guarantees that at most a 0.1 fraction of arms in the result have means smaller than d^{low}. On the other hand, for each arm with mean greater than d^{high}, with high probability it is not eliminated. We postpone the pseudocode of procedure Elimination and the proof of the following fact to Appendix A.

Fact 5.4 *Elimination$(S, d^{\text{low}}, d^{\text{high}}, \delta)$ takes $O(|S|\varepsilon^{-2} \ln \delta^{-1})$ samples in expectation, where $\varepsilon = d^{\text{high}} - d^{\text{low}}$. Let S' denote the set returned by Elimination$(S, d^{\text{low}}, d^{\text{high}}, \delta)$. Then with probability at least $1 - \delta/2$,*

$$|\{A \in S' : \mu_A < d^{\text{low}}\}| \leq 0.1|S'|.$$

Moreover, for each arm $A \in S$ with $\mu_A \geq d^{\text{high}}$, we have

$$\Pr\left[A \in S'\right] \geq 1 - \delta/2.$$

We say that a call to Elimination returns correctly if both $|\{A \in S' : \mu_A < d^{\text{low}}\}| \leq 0.1|S'|$ and $A_1(S) \in S'$ hold; otherwise the call fails. Here $A_1(S)$ denotes the arm with the largest mean in set S. Fact 5.4 directly implies that procedure Elimination returns correctly with probability at least $1 - \delta$.

5.2. Algorithm

Now we present our algorithm for the special case that the complexity of the instance is known in advance. The Known-Complexity algorithm takes as its input a Best-1-Arm instance I, the complexity H of the instance, as well as a confidence level δ. The algorithm proceeds in rounds, and maintains a sequence $\{S_r\}$ of arm sets, each of which denotes the set of arms that are still considered as candidate answers at the beginning of round r.

Roughly speaking, the algorithm eliminates the arms with $\Omega(\varepsilon_r)$ gaps at the r-th round, if they constitute a large fraction of the remaining arms. Here $\varepsilon_r = 2^{-r}$ is the accuracy parameter that we use in round r. To this end, Known-Complexity first calls procedures Med-Elim and Unif-Sampl to obtain $\hat{\mu}_{a_r}$, which is an estimation of the largest mean among all arms in S_r up to an $O(\varepsilon_r)$ error. After that, Frac-Test is called to determine whether a large proportion of arms in S_r have $\Omega(\varepsilon_r)$ gaps. If so, Frac-Test returns True, and then Known-Complexity calls the Elimination procedure with carefully chosen parameters to remove suboptimal arms from S_r.

```
Algorithm 1: Known-Complexity$(I, H, \delta)$
Input: Instance $I$ with complexity $H$ and risk $\delta$.
Output: The best arm.
$S_1 \leftarrow I$; $\hat{H} \leftarrow 4096H$;
for $r = 1$ to $\infty$ do
    if $|S_r| = 1$ then return the only arm in $S_r$; ;
    $\varepsilon_r \leftarrow 2^{-r}$; $\delta_r \leftarrow \delta/(10r^2)$;
    $a_r \leftarrow$ Med-Elim$(S_r, 0.125\varepsilon_r, 0.01)$;
    $\hat{\mu}_{a_r} \leftarrow$ Unif-Sampl$(\{a_r\}, 0.125\varepsilon_r, \delta_r)$;
    if Frac-Test$(S_r, \hat{\mu}_{a_r} - 1.75\varepsilon_r, \hat{\mu}_{a_r} - 1.125\varepsilon_r, 0.3, 0.5, \delta_r)$ then
        $\delta'_r \leftarrow \left(|S_r|\varepsilon_r^{-2}/\hat{H}\right)\delta$;
        $S_{r+1} \leftarrow$ Elimination$(S_r, \hat{\mu}_{a_r} - 0.75\varepsilon_r, \hat{\mu}_{a_r} - 0.625\varepsilon_r, \delta'_r)$;
    else
        $S_{r+1} \leftarrow S_r$;
end
```

The following two lemmas imply that there is a δ-correct algorithm for Best-1-Arm that matches the instance-wise lower bound up to an $O\left(\Delta_{[2]}^{-2} \ln\ln \Delta_{[2]}^{-1}\right)$ additive term.[6]

Lemma 5.5 *For any Best-1-Arm instance I and $\delta \in (0, 0.01)$, Known-Complexity$(I, H(I), \delta)$ returns the optimal arm in I with probability at least $1 - \delta$.*

Lemma 5.6 *For any Best-1-Arm instance I and $\delta \in (0, 0.01)$, conditioning on an event that happens with probability $1 - \delta$, Known-Complexity$(I, H(I), \delta)$ takes*

$$O\left(H(I) \cdot (\ln \delta^{-1} + \mathsf{Ent}(I)) + \Delta_{[2]}^{-2} \ln\ln \Delta_{[2]}^{-1}\right)$$

samples in expectation.

[6] Lemma 5.6 only bounds the number of samples conditioning on an event that happens with probability $1 - \delta$, so the algorithm may take arbitrarily many samples when the event does not occur. However, Known-Complexity can be transformed to a δ-correct algorithm with the same (unconditional) sample complexity bound, using the "parallel simulation" technique in the proof of Theorem 1.11 in Appendix C.

5.3. Observations

We state a few key observations on Known-Complexity, which will be used throughout the analysis. The proofs are exactly identical to those of Observations A.3 through A.5 in Appendix A. The following observation bounds the value of $\hat{\mu}_{a_r}$ at round r, assuming the correctness of Unif-Sampl and Med-Elim.

Observation 5.7 *If Unif-Sampl returns correctly at round r, $\hat{\mu}_{a_r} \le \mu_{[1]}(S_r) + 0.125\varepsilon_r$. Here $\mu_{[1]}(S_r)$ denotes the largest mean of arms in S_r. If both Unif-Sampl and Med-Elim return correctly, $\hat{\mu}_{a_r} \ge \mu_{[1]}(S_r) - 0.25\varepsilon_r$.*

The following two observations bound the thresholds used in Frac-Test and Elimination by applying Observation 5.7.

Observation 5.8 *At round r, let $c_r^{\mathrm{low}} = \hat{\mu}_{a_r} - 1.75\varepsilon_r$ and $c_r^{\mathrm{high}} = \hat{\mu}_{a_r} - 1.125\varepsilon_r$ denote the two thresholds used in Frac-Test. If Unif-Sampl returns correctly, $c_r^{\mathrm{high}} \le \mu_{[1]}(S_r) - \varepsilon_r$. If both Med-Elim and Unif-Sampl return correctly, $c_r^{\mathrm{low}} \ge \mu_{[1]}(S_r) - 2\varepsilon_r$.*

Observation 5.9 *Let $d_r^{\mathrm{low}} = \hat{\mu}_{a_r} - 0.75\varepsilon_r$ and $d_r^{\mathrm{high}} = \hat{\mu}_{a_r} - 0.625\varepsilon_r$ denote the two thresholds used in Elimination. If Unif-Sampl returns correctly, $d_r^{\mathrm{high}} \le \mu_{[1]}(S_r) - 0.5\varepsilon_r$. If both Med-Elim and Unif-Sampl return correctly, $d_r^{\mathrm{low}} \ge \mu_{[1]}(S_r) - \varepsilon_r$.*

5.4. Correctness

We define $\mathcal{E}$ as the event that all calls to procedures Unif-Sampl, Frac-Test, and Elimination return correctly. We will prove in the following that Known-Complexity returns the correct answer with probability 1 conditioning on $\mathcal{E}$, and $\Pr[\mathcal{E}] \ge 1 - \delta$. Note that Lemma 5.5 directly follows from these two claims.

Event $\mathcal{E}$ implies correctness. It suffices to show that conditioning on $\mathcal{E}$, Known-Complexity never removes the best arm, and the algorithm eventually terminates. Suppose that $A_1 \in S_r$. Observation 5.9 guarantees that at round r, the upper threshold used by Elimination is smaller than or equal to $\mu_{[1]}(S_r) - 0.5\varepsilon_r < \mu_{[1]}$. By Fact 5.4, the correctness of Elimination guarantees that $A_1 \in S_{r+1}$.

It remains to prove that Known-Complexity terminates conditioning on $\mathcal{E}$. Define $r_{\max} := \max_{G_r \ne \emptyset} r$. Suppose r^* is the smallest integer greater than $r_{\max}$ such that Med-Elim returns correctly at round r^*.[7] By Observation 5.9, the lower threshold in Elimination is greater than or equal to $\mu_{[1]} - \varepsilon_{r^*}$. The correctness of Elimination implies that

$$|S_{r^*+1}|-1 = |S_{r^*+1} \cap G_{\le r_{\max}}| \le |S_{r^*+1} \cap G_{<r^*}| = |\{A \in S_{r^*+1} : \mu_A < \mu_{[1]} - \varepsilon_{r^*}\}| < 0.1|S_{r^*+1}|.$$

It follows that $|S_{r^*+1}| = 1$. Therefore, the algorithm terminates either before or at round $r^* + 1$.

[7] Med-Elim returns correctly with probability at least 0.99 in each round, so r^* is well-defined with probability 1.

$\mathcal{E}$ happens with high probability. We first note that at round r, the probability that either Unif-Sampl or Frac-Test fails (i.e., returns incorrectly) is at most $2\delta_r$. By a union bound, the probability that at least one call to Unif-Sampl or Frac-Test returns incorrectly is upper bounded by

$$\sum_{r=1}^{\infty} 2\delta_r = \sum_{r=1}^{\infty} \frac{\delta}{5r^2} < \delta/2.$$

It remains to bound the probability that Elimination fails at some round, yet procedures Unif-Sampl and Frac-Test are always correct. Define $P(r, S_r)$ as the probability that, given the value of S_r at the beginning of round r, at least one call to Elimination returns incorrectly in round r or later, yet Unif-Sampl and Frac-Test always return correctly. We prove by induction that for any S_r that contains the optimal arm A_1,

$$P(r, S_r) \le \frac{\delta}{\hat{H}} \left(128C(r, S_r) + 16M(r, S_r)\varepsilon_r^{-2}\right), \tag{2}$$

where $M(r, S_r) := |S_r \cap G_{\le r-2}|$ and

$$C(r, S_r) := \sum_{i=r-1}^{\infty} |S_r \cap G_i| \sum_{j=r}^{i+1} \varepsilon_j^{-2} + \sum_{i=r}^{r_{\max}+1} \varepsilon_i^{-2}.$$

The details of the induction are postponed to Appendix E.

Observe that $M(1, I) = 0$ and

$$\begin{aligned} C(1, I) &= \sum_{i=0}^{\infty} |S_r \cap G_i| \sum_{j=1}^{i+1} 4^j + \sum_{i=1}^{r_{\max}+1} 4^i \\ &\le \frac{16}{3} \left(\sum_{i=0}^{\infty} |S_r \cap G_i| 4^i + 4^{r_{\max}} \right) \\ &\le \frac{16}{3} \left(\sum_{i=0}^{\infty} \sum_{A \in S_r \cap G_i} \Delta_A^{-2} + \Delta_{[2]}^{-2} \right) \le \frac{32}{3} H(I). \end{aligned}$$

Therefore we conclude that

$$\begin{aligned} \Pr\left[\mathcal{E}\right] &\ge 1 - P(1, S_1) - \frac{\delta}{2} \\ &\ge 1 - \frac{\delta}{\hat{H}} \left(128C(1, I) + 16M(1, I)\varepsilon_1^{-2}\right) - \frac{\delta}{2} \\ &\ge 1 - 128 \cdot \frac{\delta}{4096H} \cdot \frac{32H}{3} - \frac{\delta}{2} \ge 1 - \delta, \end{aligned}$$

which completes the proof of correctness. Here the first step applies a union bound. The second step follows from inequality (2), and the third step plugs in $C(1, I) \le 32H(I)/3$ and $\hat{H} = 4096H$.

5.5. Sample Complexity

As in the proof of Lemma 5.5, we define $\mathcal{E}$ as the event that all calls to procedures $\textsf{Unif-Sampl}$, $\textsf{Frac-Test}$, and $\textsf{Elimination}$ return correctly. We prove that $\textsf{Known-Complexity}$ takes

$$O\left(H(I)(\ln\delta^{-1}+\mathsf{Ent}(I))+\Delta_{[2]}^{-2}\ln\ln\Delta_{[2]}^{-1}\right)$$

samples in expectation conditioning on $\mathcal{E}$.

Samples taken by $\textsf{Unif-Sampl}$ and $\textsf{Frac-Test}$. By Facts 5.1 and 5.3, procedures $\textsf{Unif-Sampl}$ and $\textsf{Frac-Test}$ take $O\left(\varepsilon_r^{-2}\ln\delta_r^{-1}\right)=O\left(\varepsilon_r^{-2}(\ln\delta^{-1}+\ln r)\right)$ samples in total at round r.

In the proof of correctness, we showed that conditioning on $\mathcal{E}$, the algorithm does not terminate before or at round k (for $k\ge r_{\max}+1$) implies that $\textsf{Med-Elim}$ fails between round $r_{\max}+1$ and round $k-1$, which happens with probability at most $0.01^{k-r_{\max}-1}$. Thus for $k\ge r_{\max}+1$, the expected number of samples taken by $\textsf{Unif-Sampl}$ and $\textsf{Frac-Test}$ at round k is upper bounded by

$$O\left(0.01^{k-r_{\max}-1}\cdot\varepsilon_k^{-2}(\ln\delta^{-1}+\ln k)\right).$$

Summing over all $k=1,2,\ldots$ yields the following upper bound:

$$\begin{aligned}&\sum_{k=1}^{r_{\max}}\varepsilon_k^{-2}(\ln\delta^{-1}+\ln k)+\sum_{k=r_{\max}+1}^{\infty}0.01^{k-r_{\max}-1}\cdot\varepsilon_k^{-2}(\ln\delta^{-1}+\ln k)\\ =&O\left(4^{r_{\max}}(\ln\delta^{-1}+\ln r_{\max})\right)=O\left(\Delta_{[2]}^{-2}\left(\ln\delta^{-1}+\ln\ln\Delta_{[2]}^{-1}\right)\right).\end{aligned}$$

Here the first step holds since the first summation is dominated by the last term ($k=r_{\max}$), while the second one is dominated by the first term ($k=r_{\max}+1$). The second step follows from the observation that $r_{\max}=\max_{G_r\neq\emptyset}r=\left\lfloor\log_2\Delta_{[2]}^{-1}\right\rfloor$.

Samples taken by $\textsf{Med-Elim}$ and $\textsf{Elimination}$. By Facts 5.2 and 5.4, $\textsf{Med-Elim}$ and $\textsf{Elimination}$ (if called) take

$$O(|S_r|\varepsilon_r^{-2})+O(|S_r|\varepsilon_r^{-2}\ln(1/\delta'_r))=O\left(|S_r|\varepsilon_r^{-2}\left(\ln\delta^{-1}+\ln\frac{H}{|S_r|\varepsilon_r^{-2}}\right)\right)$$

samples in total at round r.

We upper bound the number of samples by a charging argument. For each round i, define r_i as the largest integer r such that $|G_{\ge r}|\ge 0.5|S_i|$.[8] Then we define

$$T_{i,j}=\begin{cases}0, & j<r_i,\\ \varepsilon_i^{-2}\left(\ln\delta^{-1}+\ln\dfrac{H}{|G_j|\varepsilon_i^{-2}}\right), & j\ge r_i\end{cases}$$

as the number of samples that each arm in G_j is charged at round i.

[8] Note that $|G_{\ge 0}|=n-1\ge 0.5|S_i|$ and $|G_{\ge r}|=0<0.5|S_i|$ for sufficiently large r, so r_i is well-defined.

We prove in Appendix E that for any i, $\sum_j |G_j| T_{i,j}$ is an upper bound on the number of samples taken by Med-Elim and Elimination at the i-th round. Moreover, the expected number of samples that each arm in group G_j is charged is upper bounded by

$$\sum_i \mathrm{E}[T_{i,j}] = O\left(\varepsilon_j^{-2}\left(\ln \delta^{-1} + \ln \frac{H}{|G_j|\varepsilon_j^{-2}}\right)\right).$$

Note that $H_k = \sum_{A \in G_k} \Delta_A^{-2} = \Theta(|G_k|\varepsilon_k^{-2})$. Therefore, Med-Elim and Elimination take

$$\begin{aligned} O\left(\sum_{i,j} |G_j| \mathrm{E}[T_{i,j}]\right) &= O\left(\sum_j |G_j|\varepsilon_j^{-2}\left(\ln \delta^{-1} + \ln \frac{H}{|G_j|\varepsilon_j^{-2}}\right)\right) \\ &= O\left(\sum_j H_j\left(\ln \delta^{-1} + \ln \frac{H}{H_j}\right)\right) \\ &= O\left(H(I)\left(\ln \delta^{-1} + \mathsf{Ent}(I)\right)\right) \end{aligned}$$

samples in expectation conditioning on $\mathcal{E}$.

In total, algorithm Known-Complexity takes

$$\begin{aligned} &O\left(\Delta_{[2]}^{-2}\left(\ln \delta^{-1} + \ln\ln \Delta_{[2]}^{-1}\right)\right) + O\left(H(I)\left(\ln \delta^{-1} + \mathsf{Ent}(I)\right)\right) \\ =&O\left(H(I)\left(\ln \delta^{-1} + \mathsf{Ent}(I)\right) + \Delta_{[2]}^{-2} \ln\ln \Delta_{[2]}^{-1}\right) \end{aligned}$$

samples in expectation conditioning on $\mathcal{E}$. This proves Lemma 5.6.

5.6. Discussion

In the Known-Complexity algorithm, knowing the complexity H in advance is crucial to the efficient allocation of confidence levels (δ_r''s) to different calls of Elimination. When H is unknown, our approach is to run an elimination procedure similar to Known-Complexity with a guess of H. The major difficulty is that when our guess is much smaller than the actual complexity, the total confidence that we allocate will eventually exceed the total confidence δ. Thus, we cannot assume in our analysis that all calls to the Elimination procedure are correct. We present our Complexity-Guessing algorithm for the Best-1-Arm problem in Appendix A.

References

Peyman Afshani, Jérémy Barbay, and Timothy M Chan. Instance-optimal geometric algorithms. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*, pages 129–138. IEEE, 2009.

Jean-Yves Audibert and Sébastien Bubeck. Best arm identification in multi-armed bandits. In *COLT-23th Conference on Learning Theory-2010*, pages 13–p, 2010.

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.

Robert Eric Bechhofer, Jack Kiefer, and Milton Sobel. *Sequential identification and ranking procedures: with special reference to Koopman-Darmois populations*, volume 3. University of Chicago Press, 1968.

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

Sébastien Bubeck, Tengyao Wang, and Nitin Viswanathan. Multiple identifications in multi-armed bandits. In *International Conference on Machine Learning*, pages 258–265, 2013.

Wei Cao, Jian Li, Yufei Tao, and Zhize Li. On top-k selection in multi-armed bandits and hidden bipartite graphs. In *Advances in Neural Information Processing Systems*, pages 1036–1044, 2015.

Alexandra Carpentier and Andrea Locatelli. Tight (lower) bounds for the fixed budget best arm identification bandit problem. In *Proceedings of the 29th Conference on Learning Theory*, 2016.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Lijie Chen and Jian Li. On the optimal sample complexity for best arm identification. *arXiv preprint arXiv:1511.03774*, 2015.

Lijie Chen and Jian Li. Open problem: Best arm identification: Almost instance-wise optimality and the gap entropy conjecture. In *29th Annual Conference on Learning Theory*, pages 1643–1646, 2016.

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.

Lijie Chen, Jian Li, and Mingda Qiao. Nearly instance optimal sample complexity bounds for top-k arm selection. In *Artificial Intelligence and Statistics*, pages 101–110, 2017.

Shouyuan Chen, Tian Lin, Irwin King, Michael R Lyu, and Wei Chen. Combinatorial pure exploration of multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 379–387, 2014.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *International Conference on Computational Learning Theory*, pages 255–270. Springer, 2002.

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

RH Farrell. Asymptotic behavior of expected sample size in certain one sided tests. *The Annals of Mathematical Statistics*, pages 36–72, 1964.

Victor Gabillon, Mohammad Ghavamzadeh, Alessandro Lazaric, and Sébastien Bubeck. Multi-bandit best arm identification. In *Advances in Neural Information Processing Systems*, pages 2222–2230, 2011.

Victor Gabillon, Mohammad Ghavamzadeh, and Alessandro Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Advances in Neural Information Processing Systems*, pages 3212–3220, 2012.

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.

Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In *Proceedings of the 29th Conference On Learning Theory (to appear)*, 2016.

Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sébastien Bubeck. lilucb: An optimal exploration algorithm for multi-armed bandits. In *Conference on Learning Theory*, pages 423–439, 2014.

Shivaram Kalyanakrishnan and Peter Stone. Efficient selection of multiple bandit arms: Theory and practice. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 511–518, 2010.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. Pac subset selection in stochastic multi-armed bandits. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 655–662, 2012.

Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 1238–1246, 2013.

Emilie Kaufmann and Shivaram Kalyanakrishnan. Information complexity in bandit subset selection. In *Conference on Learning Theory*, pages 228–251, 2013.

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.

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

Herbert Robbins. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pages 169–177. Springer, 1985.

Yuan Zhou, Xi Chen, and Jian Li. Optimal pac multiple arm identification with applications to crowdsourcing. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 217–225, 2014.

Organization of the Appendix

The appendix contains the proofs of our main results. In Section A, we present our algorithm for Best-1-Arm along with a few useful observations. In Section B and Section C, we prove the correctness and the sample complexity of our algorithm, thus proving Theorem 1.11. We present the complete proof of Theorem 1.12 in Section D. Finally, Section E contains the complete proofs of Lemma 5.5 and Lemma 5.6.

Appendix A. Upper Bound

A.1. Building Blocks

We start by presenting the missing implementation and performance guarantees of our subroutines Frac-Test and Elimination.

Fraction test. Recall that on input $(S, c^{\text{low}}, c^{\text{high}}, \theta^{\text{low}}, \theta^{\text{high}}, \delta)$, procedure Frac-Test decides whether a sufficiently large fraction (with respect to θ^{low} and θ^{high}) of arms in S have means smaller than the thresholds c^{low} and c^{high}. The pseudocode of Frac-Test is shown below.

Algorithm 2: Frac-Test$(S, c^{\text{low}}, c^{\text{high}}, \theta^{\text{low}}, \theta^{\text{high}}, \delta)$

```
Input: An arm set $S$, thresholds $c^{\text{low}}$, $c^{\text{high}}$, $\theta^{\text{low}}$, $\theta^{\text{high}}$, and confidence level $\delta$.
$\varepsilon \leftarrow c^{\text{high}} - c^{\text{low}}$; $\Delta \leftarrow \theta^{\text{high}} - \theta^{\text{low}}$;
$m \leftarrow (\Delta/6)^{-2} \ln(2/\delta)$; cnt $\leftarrow 0$;
for $i = 1$ to $m$ do
    Pick $A \in S$ uniformly at random;
    $\hat{\mu}_A \leftarrow$ Unif-Sampl$(\{A\}, \varepsilon/2, \Delta/6)$;
    if $\hat{\mu}_A < (c^{\text{low}} + c^{\text{high}})/2$ then
        cnt $\leftarrow$ cnt $+ 1$;
end
if cnt$/m > (\theta^{\text{low}} + \theta^{\text{high}})/2$ then
    return True;
else
    return False;
```

Now we prove Fact 5.3.

Fact 5.3 (restated) *Frac-Test$(S, c^{\text{low}}, c^{\text{high}}, \theta^{\text{low}}, \theta^{\text{high}}, \delta)$ takes $O((\varepsilon^{-2} \ln \delta^{-1}) \cdot (\Delta^{-2} \ln \Delta^{-1}))$ samples, where $\varepsilon = c^{\text{high}} - c^{\text{low}}$ and $\Delta = \theta^{\text{high}} - \theta^{\text{low}}$. With probability $1 - \delta$, the following two claims hold simultaneously:*

- *If Frac-Test returns True, $|\{A \in S : \mu_A < c^{\text{high}}\}| > \theta^{\text{low}}|S|$.*
- *If Frac-Test returns False, $|\{A \in S : \mu_A < c^{\text{low}}\}| < \theta^{\text{high}}|S|$.*

Proof The first claim directly follows from Fact 5.1 and

$$m \cdot O(\varepsilon^{-2} \ln \Delta^{-1}) = O((\varepsilon^{-2} \ln \delta^{-1}) \cdot (\Delta^{-2} \ln \Delta^{-1})).$$

It remains to prove the contrapositive of the second claim: $|\{A \in S : \mu_A < c^{\text{low}}\}| \geq \theta^{\text{high}}|S|$ implies Frac-Test returns True, and $|\{A \in S : \mu_A < c^{\text{high}}\}| \leq \theta^{\text{low}}|S|$ implies Frac-Test returns False.

Suppose $|\{A \in S : \mu_A < c^{\text{low}}\}| \ge \theta^{\text{high}}|S|$. Then in each iteration of the for-loop, it holds that $\mu_A < c^{\text{low}}$ with probability at least θ^{high}. Conditioning on $\mu_A < c^{\text{low}}$, by Fact 5.1 we have

$$\hat{\mu}_A \le \mu_A + \varepsilon/2 < c^{\text{low}} + \varepsilon/2 = (c^{\text{low}} + c^{\text{high}})/2$$

with probability at least $1 - \Delta/6$. Thus, the expected increment of counter cnt is lower bounded by

$$\theta^{\text{high}}(1 - \Delta/6) \ge \theta^{\text{high}} - \Delta/6.$$

Thus, cnt/m is the mean of m i.i.d. Bernoulli random variables with means greater than or equal to $\theta^{\text{high}} - \Delta/6$. By the Chernoff bound, it holds with probability $1 - \delta/2$ that

$$\text{cnt}/m \ge \theta^{\text{high}} - \Delta/6 - \Delta/6 > (\theta^{\text{low}} + \theta^{\text{high}})/2.$$

An analogous argument proves $\text{cnt}/m < (\theta^{\text{low}} + \theta^{\text{high}})/2$ with probability $1 - \delta/2$, given $|\{A \in S : \mu_A < c^{\text{high}}\}| \le \theta^{\text{low}}|S|$. This completes the proof. ■

Elimination. We implement procedure Elimination by repeatedly calling Frac-Test to determine whether a large fraction of the remaining arms have means smaller than the thresholds. If so, we uniformly sample the arms, and eliminate those with low empirical means.

Algorithm 3: Elimination$(S, d^{\text{low}}, d^{\text{high}}, \delta)$

```
Input: An arm set $S$, thresholds $d^{\text{low}}$, $d^{\text{high}}$, and confidence level $\delta$.
Output: Arm set after the elimination.
$S_1 \leftarrow S$;
$d^{\text{mid}} \leftarrow (d^{\text{low}} + d^{\text{high}})/2$;
for $r = 1$ to $+\infty$ do
    $\delta_r \leftarrow \delta/(10 \cdot 2^r)$;
    if Frac-Test$(S_r, d^{\text{low}}, d^{\text{mid}}, 0.05, 0.1, \delta_r)$ then
        $\hat{\mu} \leftarrow$ Unif-Sampl$(S_r, (d^{\text{high}} - d^{\text{mid}})/2, \delta_r)$;
        $S_{r+1} \leftarrow \left\{A \in S_r : \hat{\mu}_A > (d^{\text{mid}} + d^{\text{high}})/2\right\}$;
    else
        return $S_r$;
end
```

We prove Fact 5.4 in the following.

Fact 5.4 (restated) *Elimination$(S, d^{\text{low}}, d^{\text{high}}, \delta)$ takes $O(|S|\varepsilon^{-2}\ln\delta^{-1})$ samples in expectation, where $\varepsilon = d^{\text{high}} - d^{\text{low}}$. Let S' be the set returned by Elimination$(S, d^{\text{low}}, d^{\text{high}}, \delta)$. Then we have*

$$\Pr[|\{A \in S' : \mu_A < d^{\text{low}}\}| \le 0.1|S'|] \ge 1 - \delta/2.$$

Moreover, for each arm $A \in S$ with $\mu_A \ge d^{\text{high}}$, we have

$$\Pr[A \in S'] \ge 1 - \delta/2.$$

Proof Let $\varepsilon = d^{\text{high}} - d^{\text{low}}$. To bound the number of samples taken by Elimination, we note that the number of samples taken in the r-th iteration is dominated by that taken by Unif-Sampl, $O(|S_r|\varepsilon^{-2}\ln\delta_r^{-1})$. It suffices to show that $|S_r|$ decays exponentially (in expectation); a direct summation over all r proves the sample complexity bound.

We fix a particular round r. Suppose Frac-Test returns correctly (which happens with probability at least $1-\delta_r$) and the algorithm does not terminate at round r. Then by Fact 5.3, it holds that

$$|\{A \in S_r : \mu_A < d^{\text{mid}}\}| > 0.05|S_r|.$$

For each $A \in S_r$ with $\mu_A < d^{\text{mid}}$, it holds with probability $1-\delta_r$ that

$$\hat{\mu}_A < \mu_A + (d^{\text{high}} - d^{\text{mid}})/2 < d^{\text{mid}} + (d^{\text{high}} - d^{\text{mid}})/2 = (d^{\text{mid}} + d^{\text{high}})/2.$$

Note that $\delta_r = \delta/(10 \cdot 2^r) \le 0.1$. Thus, at most a 0.1 fraction of arms in $\{A \in S_r : \mu_A < d^{\text{mid}}\}$ would remain in S_{r+1} in expectation. It follows that conditioning on the correctness of Frac-Test at round r, the expectation of $|S_{r+1}|$ is upper bounded by

$$0.05|S_r| \cdot \delta_r + 0.95|S_r| \le 0.05|S_r|/10 + 0.95|S_r| = 0.955|S_r|.$$

Moreover, even if Frac-Test returns incorrectly, which happens with probability at most 0.1, we still have $|S_{r+1}| \le |S_r|$. Therefore,

$$\mathrm{E}[|S_{r+1}|] \le 0.9 \cdot 0.955\mathrm{E}[|S_r|] + 0.1\mathrm{E}[|S_r|] < 0.96\mathrm{E}[|S_r|].$$

A simple induction yields $\mathrm{E}[|S_r|] \le 0.96^{r-1}|S|$. Then the sample complexity of Elimination is upper bounded by

$$\begin{aligned}\sum_{r=1}^{\infty}\mathrm{E}[|S_r|]\varepsilon^{-2}\ln\delta_r^{-1} &= O\left(|S|\varepsilon^{-2}\sum_{r=1}^{\infty}0.96^{r-1}(\ln\delta^{-1}+r)\right)\\ &= O\left(|S|\varepsilon^{-2}\ln\delta^{-1}\right).\end{aligned}$$

Then we proceed to the proof of the second claim. Let $\mathcal{E}$ denote the event that all calls to procedure Frac-Test returns correctly. By Fact 5.3 and a union bound,

$$\Pr\left[\mathcal{E}_A\right] \ge 1 - \sum_{r=1}^{\infty}\delta_r \ge 1 - \delta/2.$$

Conditioning on event $\mathcal{E}$, if the algorithm terminates and returns S_r at round r, Fact 5.3 implies that

$$|\{A \in S_r : \mu_A < d^{\text{low}}\}| < 0.1|S_r|.$$

This proves the second claim.

Finally, fix an arm $A \in S$ with $\mu_A > d^{\text{high}}$. Define $\mathcal{E}_A$ as the event that every call to Frac-Test returns correctly in the algorithm, and $|\hat{\mu}_A - \mu_A| < (d^{\text{high}} - d^{\text{mid}})/2$ in every round. By Facts 5.1 and 5.3,

$$\Pr\left[\mathcal{E}_A\right] \ge 1 - \sum_{r=1}^{\infty}2\delta_r \ge 1 - \delta/2.$$

Then in each round r, it holds conditioning on $\mathcal{E}_A$ that

$$\hat{\mu}_A \ge \mu_A - (d^{\text{high}} - d^{\text{mid}})/2 > d^{\text{high}} - (d^{\text{high}} - d^{\text{mid}})/2 = (d^{\text{mid}} + d^{\text{high}})/2.$$

Thus, with probability $1-\delta/2$, A is never removed from S_r. ■

A.2. Overview

As shown in Section 5, we can solve Best-1-Arm using

$$O\left(H \cdot (\mathsf{Ent} + \ln \delta^{-1}) + \Delta_{[2]}^{-2} \ln \ln \Delta_{[2]}^{-1}\right)$$

samples, if we know in advance the complexity of the instance, i.e., $H = \sum_{i=2}^{n} \Delta_{[i]}^{-2}$.

The value of H is essential for allocating appropriate confidence levels to different calls of Elimination and achieving the near-optimal sample complexity. When H is unknown, our strategy is to guess its value. The major difficulty with our approach is that when our guess, $\hat{H}$, is much smaller than the actual complexity H, the total confidence that we allocate will exceed the total confidence δ. To prevent this from happening, we maintain the total confidence that we have allocated so far, and terminate the algorithm as soon as the sum exceeds δ.[9] After that, we try a guess that is a hundred times larger. As we will see later, the most challenging part of the analysis is to ensure that our algorithm does not return an incorrect answer when $\hat{H}$ is too small.

We also keep track of the number of samples that have been taken so far. Roughly speaking, when the number exceeds $100\hat{H}$, we also terminate the algorithm and try the next guess of $\hat{H}$. This simplifies the analysis by ensuring that the number of samples we take for each guess grows exponentially, and thus it suffices to bound the number of samples taken on the last guess.

A.3. Algorithm

Algorithm Entropy-Elimination takes an instance of Best-1-Arm, a confidence δ and a guess of complexity $\hat{H}_t = 100^t$. It either returns an optimal arm (i.e., "accept" $\hat{H}_t$) or reports an error indicating that the given $\hat{H}_t$ is much smaller than the actual complexity (i.e., "reject" $\hat{H}_t$).

Throughout the algorithm, we maintain S_r, H_r and T_r for each round r. S_r denotes the collection of arms that are still under consideration at the beginning of round r. We say that an arm is removed (or eliminated) at round r, if it is in $S_r \setminus S_{r+1}$. Roughly speaking, H_r is an estimate of the total complexity of arms in group $G_1, G_2, \ldots, G_r$. When this quantity exceeds our guess $\hat{H}_t$, Entropy-Elimination directly rejects (i.e., returns an error). T_r is an upper bound on the number of samples taken by Med-Elim and Elimination[10] before round r. As mentioned before, we also terminate the algorithm when T_r exceeds $100\hat{H}_t$. Intuitively, this prevents Entropy-Elimination from taking too many samples on small guesses of H, which gives rise to an inferior sample complexity.

In each round of Entropy-Elimination, we first call Med-Elim to obtain a near-optimal arm a_r. Then we use Unif-Sampl to estimate the mean of a_r, denoted by $\hat{\mu}_{a_r}$. After that, we call Frac-Test with appropriate parameters to find out whether a considerable fraction of arms in S_r have gaps larger than ε_r. If so, we call procedure Elimination and update the value of H_{r+1} accordingly. Note that we set the thresholds $\{\theta_r\}$ of Frac-Test such that the intervals $[\theta_{r-1}, \theta_r]$ are disjoint. In particular, this property is essential for proving Lemma B.6 in the analysis of the correctness of the algorithm.

Our algorithm for Best-1-Arm guesses the complexity of the instance and invokes Entropy-Elimination to check whether the guess is reasonable. If Entropy-Elimination reports an error,

[9] For ease of analysis, we actually use δ^2 instead of δ in the algorithm.

[10] As we will see later, the analysis of the sample complexity of Med-Elim and Elimination are different from the other two procedures.

Algorithm 4: Entropy-Elimination$(I, \delta, \hat{H}_t)$

```
Input: Instance $I$, confidence $\delta$ and a guess of complexity $\hat{H}_t = 100^t$.
Output: The best arm, or an error indicating the guess is wrong.
$S_1 \leftarrow I$; $H_1 \leftarrow 0$; $T_1 \leftarrow 0$;
$\theta_0 \leftarrow 0.3$; $c \leftarrow \log_4 100$;
for $r = 1$ to $\infty$ do
    if $|S_r| = 1$ then
        return the only arm in $S_r$;
    $\varepsilon_r \leftarrow 2^{-r}$; $\delta_r \leftarrow \delta/(50r^2t^2)$;
    $\delta'_r \leftarrow (4|S_r|\varepsilon_r^{-2}/\hat{H})\delta^2$;
    $T_{r+1} \leftarrow T_r + |S_r|\varepsilon_r^{-2} \ln\left(|S_r|\varepsilon_r^{-2}\delta/\hat{H}_t\right)^{-1}$;
    if $(H_r + 4|S_r|\varepsilon_r^{-2} \geq \hat{H}_t)$ or $(T_{r+1} \geq 100\hat{H}_t)$ then
        return error;
    $a_r \leftarrow$ Med-Elim$(S_r, 0.125\varepsilon_r, 0.01)$;
    $\hat{\mu}_{a_r} \leftarrow$ Unif-Sampl$(\{a_r\}, 0.125\varepsilon_r, \delta_r)$;
    $\theta_r \leftarrow \theta_{r-1} + (ct - r)^{-2}/10$;
    if Frac-Test$(S_r, \hat{\mu}_{a_r} - 1.75\varepsilon_r, \hat{\mu}_{a_r} - 1.125\varepsilon_r, \delta_r, \theta_{r-1}, \theta_r)$ then
        $H_{r+1} \leftarrow H_r + 4|S_r|\varepsilon_r^{-2}$;
        $S_{r+1} \leftarrow$ Elimination$(S_r, \hat{\mu}_{a_r} - 0.75\varepsilon_r, \hat{\mu}_{a_r} - 0.625\varepsilon_r, \delta'_r)$;
    else
        $S_{r+1} \leftarrow S_r$;
        $H_{r+1} \leftarrow H_r$;
end
```

we try a guess that is a hundred times larger. Otherwise, we return the arm chosen by Entropy-Elimination.

Algorithm 5: Complexity-Guessing

```
Input: Instance $I$ and confidence $\delta$.
Output: The best arm.
for $t = 1$ to $\infty$ do
    $\hat{H}_t \leftarrow 100^t$;
    Call Entropy-Elimination$(I, \delta, \hat{H}_t)$;
    if Entropy-Elimination does not return an error then
        return the arm returned by Entropy-Elimination;
end
```

A.4. Observations

We start with a few simple observations on Entropy-Elimination that will be used throughout the analysis.

We first note that Entropy-Elimination lasts $O(t)$ rounds on guess $\hat{H}_t$, and our definition of θ_r ensures that all θ_r are in $[0.3, 0.5]$.

Observation A.1 *The for-loop in Entropy-Elimination$(I, \delta, \hat{H}_t)$ is executed at most ct times, where $c = \log_4 100$.*

Proof When $r \ge ct - 1$,

$$H_r + 4|S_r|\varepsilon_r^{-2} \ge 4 \cdot 4^{ct-1} = \hat{H}_t.$$

Thus Entropy-Elimination rejects at the if-statement. ■

Observation A.2 *For all $t \ge 1$ and $1 \le r \le ct - 1$, $0.3 \le \theta_{r-1} \le \theta_r \le 0.5$.*

Proof Clearly $\theta_r \ge \theta_0 = 0.3$. Moreover,

$$\theta_r = \theta_0 + \sum_{k=1}^{r} (ct - k)^{-2}/10 \le 0.3 + \frac{1}{10}\sum_{k=1}^{\infty} k^{-2} \le 0.5.$$

■

The following observation bounds the value of $\hat{\mu}_{a_r}$ at round r, conditioning on the correctness of Unif-Sampl and Med-Elim.

Observation A.3 *If Unif-Sampl returns correctly at round r, $\hat{\mu}_{a_r} \le \mu_{[1]}(S_r) + 0.125\varepsilon_r$. Here $\mu_{[1]}(S_r)$ denotes the largest mean of arms in S_r. If both Unif-Sampl and Med-Elim return correctly, $\hat{\mu}_{a_r} \ge \mu_{[1]}(S_r) - 0.25\varepsilon_r$.*

Proof By definition, $\mu_{a_r} \le \mu_{[1]}(S_r)$. When Unif-Sampl$(\{a_r\}, 0.125\varepsilon_r, \delta_r)$ returns correctly, it holds that

$$\hat{\mu}_{a_r} \le \mu_{a_r} + 0.125\varepsilon_r \le \mu_{[1]} + 0.125\varepsilon_r.$$

When both Med-Elim and Unif-Sampl are correct, $\mu_{a_r} \ge \mu_{[1]}(S_r) - 0.125\varepsilon_r$, and thus

$$\hat{\mu}_{a_r} \ge \mu_{a_r} - 0.125\varepsilon_r \ge \mu_{[1]}(S_r) - 0.25\varepsilon_r.$$

■

The following two observations bound the thresholds used in Frac-Test and Elimination by applying Observation A.3.

Observation A.4 *At round r, let $c_r^{\text{low}} = \hat{\mu}_{a_r} - 1.75\varepsilon_r$ and $c_r^{\text{high}} = \hat{\mu}_{a_r} - 1.125\varepsilon_r$ denote the two thresholds used in Frac-Test. If Unif-Sampl returns correctly, $c_r^{\text{high}} \le \mu_{[1]}(S_r) - \varepsilon_r$. If both Med-Elim and Unif-Sampl return correctly, $c_r^{\text{low}} \ge \mu_{[1]}(S_r) - 2\varepsilon_r$.*

Proof Observation A.3 implies that when Unif-Sampl is correct,

$$c_r^{\text{high}} \le \mu_{[1]}(S_r) + 0.125\varepsilon_r - 1.125\varepsilon_r = \mu_{[1]}(S_r) - \varepsilon_r$$

and when both Med-Elim and Unif-Sampl return correctly,

$$c_r^{\text{low}} \ge \mu_{[1]}(S_r) - 0.25\varepsilon_r - 1.75\varepsilon_r = \mu_{[1]}(S_r) - 2\varepsilon_r.$$

■

Observation A.5 *Let $d_r^{\text{low}} = \hat{\mu}_{a_r} - 0.75\varepsilon_r$ and $d_r^{\text{high}} = \hat{\mu}_{a_r} - 0.625\varepsilon_r$ denote the two thresholds used in Elimination. If Unif-Sampl returns correctly, $d_r^{\text{high}} \le \mu_{[1]}(S_r) - 0.5\varepsilon_r$. If both Med-Elim and Unif-Sampl return correctly, $d_r^{\text{low}} \ge \mu_{[1]}(S_r) - \varepsilon_r$.*

Proof By the same argument, we have

$$d_r^{\text{high}} \le \mu_{[1]}(S_r) + 0.125\varepsilon_r - 0.625\varepsilon_r = \mu_{[1]}(S_r) - 0.5\varepsilon_r$$

when Unif-Sampl returns correctly, and

$$d_r^{\text{low}} \ge \mu_{[1]}(S_r) - 0.25\varepsilon_r - 0.75\varepsilon_r = \mu_{[1]}(S_r) - \varepsilon_r$$

when both Med-Elim and Unif-Sampl are correct. ■

Appendix B. Analysis of Correctness

B.1. Overview

We start with a high-level overview of the proof of our algorithm's correctness. We first define a good event on which we condition in the rest of the analysis. Let $\mathcal{E}_1$ be the event that in a particular run of Complexity-Guessing, all calls of procedure Unif-Sampl and Frac-Test return correctly. Recall that δ_r, the confidence of Unif-Sampl and Frac-Test, is set to be $\delta/(50r^2t^2)$ in the r-th round of iteration t. By a union bound,

$$\Pr[\mathcal{E}_1] \geq 1 - 2\sum_{t=1}^{\infty}\sum_{r=1}^{\infty} \delta/(50t^2r^2) = 1 - 2\delta(\pi^2/6)^2/50 \geq 1 - \delta/3.$$

The δ-correctness of our algorithm is guaranteed by the following two lemmas. The first lemma states that Entropy-Elimination accepts a guess $\hat{H}_t$ and returns correctly with high probability when $\hat{H}_t$ is sufficiently large. The second lemma guarantees that Entropy-Elimination rejects a guess $\hat{H}_t$ when $\hat{H}_t$ is significantly smaller than H, the actual complexity. More precisely, we define the following two thresholds:

$$t_{\max} = \lfloor \log_{100} H \rfloor - 2$$

and

$$t'_{\max} = \left\lceil \log_{100} \left[H(\mathsf{Ent} + \ln \delta^{-1})\delta^{-1} \right] \right\rceil + 2.$$

The precise statements of the two lemmas are shown below.

Lemma B.1 *With probability $1 - \delta/3$ conditioning on event $\mathcal{E}_1$, Complexity-Guessing halts before or at iteration $t'_{\max}$ and it never returns a sub-optimal arm between iteration $t_{\max} + 1$ and $t'_{\max}$.*

Lemma B.2 *With probability $1 - \delta/3$ conditioning on event $\mathcal{E}_1$, Complexity-Guessing never returns a sub-optimal arm in the first $t_{\max}$ iterations.*

Lemma B.1 and Lemma B.2 directly imply the following theorem.

Theorem B.3 *Complexity-Guessing is a δ-correct algorithm for Best-1-Arm.*

Proof Recall that $\Pr[\mathcal{E}_1] \geq 1 - \delta/3$. It follows directly from Lemma B.1 and Lemma B.2 that with probability $1 - \delta$, Entropy-Elimination accepts at least one of $\hat{H}_1, \hat{H}_2, \ldots, \hat{H}_{t'_{\max}}$. Moreover, when Entropy-Elimination accepts, it returns the optimal arm. Therefore, Complexity-Guessing is δ-correct. ■

B.2. Useful Lemmas

To analyze our algorithm, it is essential to bound the probability that a specific guess $\hat{H}_t$ gets rejected by Entropy-Elimination. We hope that this probability is high when $\hat{H}_t$ is small (compared to the true complexity H), while it is reasonably low when $\hat{H}_t$ is large enough.

It turns out to be useful to consider the following procedure $\mathbb{P}$ obtained from Entropy-Elimination by removing the if-statement that checks whether $H_r + 4|S_r|\varepsilon_r^{-2} \geq \hat{H}_t$ and $T_{r+1} \geq 100\hat{H}_t$. In other words, the modified procedure $\mathbb{P}$ never rejects, regardless the value of $\hat{H}_t$. Note that r, the number of rounds, may exceed ct in $\mathbb{P}$, which leads to invalid values of θ_r. In this case, we simply assume that the thresholds used in Frac-Test are 0.3 and 0.5 respectively, and the following analysis still works. Define random variable H_∞ and T_∞ to be the final estimation of the complexity and the number of samples at the end of $\mathbb{P}$. More precisely, if $\mathbb{P}$ terminates at round r^*, then H_∞ and T_∞ are defined as H_{r^*} and T_{r^*}, respectively.

Note that there is a natural mapping from an execution of $\mathbb{P}$ to an execution of Entropy-Elimination. In particular, if both $H_\infty < \hat{H}_t$ and $T_\infty < 100\hat{H}_t$ hold in an execution of procedure $\mathbb{P}$, then Entropy-Elimination accepts in the corresponding run. Therefore, we may upper bound the probability of rejection by establishing upper bounds of H_∞ and T_∞. The following two lemmas bound the expectation of H_∞ and T_∞ conditioning on the event that Elimination always returns correctly.

Lemma B.4 $\mathrm{E}[H_\infty | \textit{all Elimination return correctly}] \le 256H.$

Lemma B.5 *Suppose* $\hat{H}_t \ge H$. $\mathrm{E}[T_\infty | \textit{all Elimination return correctly}] \le 16(H(\mathsf{Ent} + \ln\delta^{-1} + \ln(\hat{H}_t/H)))$.

Note that it is crucial for the two lemmas above that all Elimination are correct. The following lemma gives an upper bound on the probability that *some* call of Elimination returns incorrectly. Lemmas B.4 through B.6 together can be used to upper bound the probability of rejecting a guess $\hat{H}$. In the statement of Lemma B.6, we abuse the notation a little bit by assuming $A_1 \in G_\infty$ and $\Delta_{[1]}^{-2} = +\infty$.

Lemma B.6 *Suppose that* $s \in \{2, 3, \ldots, n\}$ *and* $r^* \in \mathbb{N} \cup \{\infty\}$ *satisfy* $A_{s-1} \in G_{r^*}$. *When Entropy-Elimination runs on parameter* $\hat{H}_t < \Delta_{[s-1]}^{-2}$, *the probability that there exists a call of procedure Elimination that returns incorrectly before round* r^* *is upper bounded by*

$$3000s\left(\sum_{i=s}^{n}\Delta_{[i]}^{-2}\right)\delta^2/\hat{H}_t.$$

The proofs of the three lemmas above are shown below.

Proof [Proof of Lemma B.4] In the following analysis, we always implicitly condition on the event that all Elimination return correctly. Define $H(r, S)$ as the expectation of $H_\infty - H_r$ at the beginning of the r-th round of Entropy-Elimination, when the current set of arms is $S_r = S$. Let $r_{\max}$ denote $\left\lfloor \log_2 \Delta_{[2]}^{-1} \right\rfloor$. Define

$$C(r, S) = \sum_{i=r-1}^{\infty} |S \cap G_i| \sum_{j=r}^{i+1} \varepsilon_j^{-2} + \sum_{i=r}^{r_{\max}+1} \varepsilon_i^{-2}$$

and $M(r, S) = |S \cap G_{\le r-2}|$. We prove by induction on r that

$$H(r, S) \le 128C(r, S) + 16M(r, S)\varepsilon_r^{-2}. \tag{3}$$

We start with the base case at round $r_{\max} + 2$. Recall that c_r^{low} and d_r^{low} denote the lower threshold of Frac-Test and Elimination in round r respectively. For all $r \ge r_{\max} + 2$, if Med-Elim returns correctly at round r (which happens with probability 0.99), according to Observation A.4 and Observation A.5, we have

$$d_r^{\text{low}} \geq c_r^{\text{low}} \geq \mu_{[1]} - 2\varepsilon_r \geq \mu_{[1]} - 2^{-(r_{\max}+1)} \geq \mu_{[2]}.$$

Since Frac-Test returns correctly (contioning on $\mathcal{E}_1$) and

$$|\{A \in S_r : \mu_A \leq c_r^{\text{low}}\}| \geq |\{A \in S_r : \mu_A \leq \mu_{[2]}\}| = |S_r| - 1 \geq 0.5|S_r| \geq \theta_r|S_r|$$

(the last step applies Observation A.2), Frac-Test must return True and Elimination will be called. Since we assume that all calls of Elimination return correctly, we have

$$|S_{r+1}| - 1 = |\{A \in S_{r+1} : \mu_A \leq \mu_{[2]}\}| \leq |\{A \in S_{r+1} : \mu_A \leq d_r^{\text{low}}\}| \leq 0.1|S_{r+1}|,$$

which guarantees that S_{r+1} only contains the optimal arm and the algorithm will return correctly in the next round. Let r_0 denote the first round after round $r_{\max} + 2$ (inclusive) in which Med-Elim returns correctly. Then according to the discussion above, we have $\Pr[r_0 = r] \leq 0.01^{r - r_{\max} - 2}$ for all $r \geq r_{\max} + 2$. Thus it follows from a direct summation on possible values of r_0 that

$$\begin{aligned} H(r_{\max}+2, S) &\leq \sum_{r=r_{\max}+2}^{\infty} \Pr[r_0 = r] \cdot 4|S|\varepsilon_r^{-2} \\ &\leq \sum_{r=r_{\max}+2}^{\infty} 4|S|\varepsilon_r^{-2} 0.01^{r-r_{\max}-2} \\ &\leq 8|S|\varepsilon_{r_{\max}+2}^{-2} \leq 16M(r_{\max}+2, S)\varepsilon_{r_{\max}+2}^{-2}, \end{aligned}$$

which proves the base case.

Before proving the induction step, we note the following fact: for $r = 1, 2, \ldots, r_{\max} + 1$,

$$\begin{aligned} C(r, S) - C(r+1, S) &= \sum_{i=r-1}^{\infty} |S \cap G_i| \sum_{j=r}^{i+1} \varepsilon_j^{-2} - \sum_{i=r}^{\infty} |S \cap G_i| \sum_{j=r+1}^{i+1} \varepsilon_j^{-2} + \varepsilon_r^{-2} \\ &= \sum_{i=r-1}^{\infty} |S \cap G_i| \varepsilon_r^{-2} + \varepsilon_r^{-2} \\ &= (|S \cap G_{\geq r-1}| + 1)\varepsilon_r^{-2}. \end{aligned} \tag{4}$$

Suppose inequality (3) holds for $r + 1$. Consider the following three cases of the execution of Entropy-Elimination in round r. Let $N_{\mathsf{cur}} = |S \cap G_{r-1}|$ and $N_{\mathsf{big}} = |S \cap G_{\geq r}|$. For brevity, let N_{sma} denote $M(r, S)$ in the following. We have $N_{\mathsf{sma}} + N_{\mathsf{cur}} + N_{\mathsf{big}} = |S| - 1$. Note that S_{r+1} is the set of arms that survive round r.

Case 1: Med-Elim returns correctly and Frac-Test returns True.

According to the induction hypothesis, the expectation of $H_\infty - H_r$ in this case can be bounded by:

$$\begin{aligned}
&H(r+1, S_{r+1}) + 4|S|\varepsilon_r^{-2}\\
\le&128C(r+1, S_{r+1}) + 16M(r+1, S_{r+1})\varepsilon_{r+1}^{-2} + 4|S|\varepsilon_r^{-2}\\
\le&128C(r+1, S) + 16[(N_{\mathsf{sma}} + N_{\mathsf{cur}})/10] \cdot (4\varepsilon_r^{-2}) + 4|S|\varepsilon_r^{-2}\\
=&128[C(r, S) - (N_{\mathsf{cur}} + N_{\mathsf{big}} + 1)\varepsilon_r^{-2}] + (6.4N_{\mathsf{sma}} + 6.4N_{\mathsf{cur}} + 4|S|)\varepsilon_r^{-2}\\
=&128C(r, S) + (10.4N_{\mathsf{sma}} - 117.6N_{\mathsf{cur}} - 124N_{\mathsf{big}} - 124)\varepsilon_r^{-2}\\
\le&128C(r, S) + 10.4N_{\mathsf{sma}}\varepsilon_r^{-2}.
\end{aligned}$$

Here the third line follows from the fact that $S_{r+1} \subseteq S$ and $C(r+1, S)$ is monotone in S. Moreover, the correctness of the Elimination procedure implies that $M(r+1, S_{r+1}) \le (N_{\mathsf{sma}} + N_{\mathsf{cur}})/10$. The fourth line applies identity (4).

Case 2: Med-Elim returns correctly and Frac-Test returns False.

Since Frac-Test is always correct (conditioning on $\mathcal{E}_1$) and it returns False, Fact 5.3, Observation A.2 and Observation A.4 together imply $N_{\mathsf{sma}} \le \theta_r|S| \le |S|/2$. Thus $N_{\mathsf{sma}} \le |S| - N_{\mathsf{sma}} = N_{\mathsf{cur}} + N_{\mathsf{big}} + 1$. As Elimination is not called in this round, the expectation of $H_\infty - H_r$ in this case can be bounded by

$$\begin{aligned}
H(r+1, S) \le&128C(r+1, S) + 16M(r+1, S)\varepsilon_{r+1}^{-2}\\
\le&128[C(r, S) - (N_{\mathsf{cur}} + N_{\mathsf{big}} + 1)\varepsilon_r^{-2}] + 64(N_{\mathsf{sma}} + N_{\mathsf{cur}})\varepsilon_r^{-2}\\
=&128C(r, S) + (64N_{\mathsf{sma}} - 64N_{\mathsf{cur}} - 128N_{\mathsf{big}} - 128)\varepsilon_r^{-2} \le 128C(r, S).
\end{aligned}$$

Here the last step follows from $64N_{\mathsf{sma}} - 64N_{\mathsf{cur}} - 128N_{\mathsf{big}} - 128 \le 64(N_{\mathsf{sma}} - N_{\mathsf{cur}} - N_{\mathsf{big}} - 1) \le 0$.

Case 3: Med-Elim returns incorrectly.

In this case, the worst scenario happens when we add $4|S|\varepsilon_r^{-2}$ to the complexity H_r, but no arms are eliminated. Then the expectation of $H_\infty - H_r$ in this case can be bounded by

$$\begin{aligned}
&H(r+1, S) + 4|S|\varepsilon_r^{-2}\\
\le&128C(r+1, S) + 16M(r+1, S)\varepsilon_{r+1}^{-2} + 4|S|\varepsilon_r^{-2}\\
\le&128[C(r, S) - (N_{\mathsf{cur}} + N_{\mathsf{big}} + 1)\varepsilon_r^{-2}] + [64(N_{\mathsf{sma}} + N_{\mathsf{cur}}) + 4|S|]\varepsilon_r^{-2}\\
=&128C(r, S) + (68N_{\mathsf{sma}} - 60N_{\mathsf{cur}} - 124N_{\mathsf{big}} - 124)\varepsilon_r^{-2} \le 128C(r, S) + 68N_{\mathsf{sma}}\varepsilon_r^{-2}.
\end{aligned}$$

Recall that Case 3 happens with probability at most 0.01. Thus we have:

$$\begin{aligned}
H(r, S) &\le 0.01\left[128C(r, S) + 68M(r, S)\varepsilon_r^{-2}\right] + 0.99\left[128C(r, S) + 10.4M(r, S)\varepsilon_r^{-2}\right]\\
&\le 128C(r, S) + 16M(r, S)\varepsilon_r^{-2}.
\end{aligned}$$

The induction is completed. Note that (3) directly implies our bound:

$$\begin{aligned}
&\mathrm{E}\left[H_\infty | \text{all } \textsf{Elimination} \text{ return correctly}\right] \\
=&H(1,S) \le 128C(1,S) + 16M(1,S) \\
=&128\sum_{i=0}^{\infty}|S\cap G_i|\cdot\left(\sum_{j=0}^{i+1}4^j\right) \\
\le&256\sum_{i=0}^{\infty}4^{i+1}|S\cap G_i| \\
\le&256\sum_{i=0}^{\infty}\sum_{A\in S\cap G_i}\Delta_A^{-2} = 256H.
\end{aligned}$$

■

Then we prove Lemma B.5, which is restated below.

Lemma B.5. (restated) *Suppose* $\hat{H}_t \ge H$. $\mathrm{E}[T_\infty | \textit{all } \textsf{Elimination} \textit{ return correctly}] \le 16(H(\mathsf{Ent} + \ln\delta^{-1} + \ln(\hat{H}_t/H)))$.

Proof [Proof of Lemma B.5] Recall that T_∞ is the sum of

$$|S_r|\varepsilon_r^{-2}\ln\left(\frac{|S_r|\varepsilon_r^{-2}}{\hat{H}_t}\delta\right)^{-1} = |S_r|\varepsilon_r^{-2}\left(\ln\frac{H}{|S_r|\varepsilon_r^{-2}} + \ln\delta^{-1} + \ln\frac{\hat{H}_t}{H}\right) \tag{5}$$

for all round r. T_∞ serves as an upper bound on the expected number of samples taken by Med-Elim and Elimination (up to a constant factor). Before the technical proof, we discuss the intuition of our analysis.

In order to bound T_∞, we attribute each term in (5) to a specific subset of arms. For simplicity, we assume for now that this term is just $|S_r|\varepsilon_r^{-2} = 4^r|S_r|$. Roughly speaking, we "charge" a cost of $\varepsilon_r^{-2} = 4^r$ to each arm in group $G_{\ge r}$. We expect that $|G_{\ge r}|$ is at least a constant times $|S_r|$, so that the number of samples (i.e., $4^r|S_r|$) can be covered by the total charges. Then the analysis reduces to calculating the total cost that each arm is charged. Fix an arm $A \in G_{r'}$ for some r'. As described above, A is charged 4^r in round r ($1 \le r \le r'$), and thus the total charge is bounded by $4^{r'}$, which is the actual complexity of A.

Now we start the formal proof. Consider the execution of procedure $\mathbb{P}$ on $\hat{H}_t$. We define a collection of random variables $\{T_{i,j} : i, j \ge 1\}$, where $T_{i,j}$ corresponds to the cost we charge each arm in G_j at round i. For each i, let r_i denote the largest integer such that $|G_{\ge r_i}| \ge 0.5|S_i|$. Note that such an r_i always exists, as $|G_{\ge 1}| = |S_1| \ge 0.5|S_i|$ and $|G_{\ge r}| = 0$ for sufficiently large r. We define $T_{i,j}$ as

$$T_{i,j} = \begin{cases} 0, & j < r_i, \\ \varepsilon_i^{-2}\left(\ln\frac{H}{|G_j|\varepsilon_i^{-2}} + \ln\delta^{-1} + \ln\frac{\hat{H}_t}{H}\right), & j \ge r_i. \end{cases}$$

Note that this slightly differs from the proof idea described above: $T_{i,j}$ might be positive when $i > j$ (i.e., we may not always charge $G_{\ge i}$ in round i). In fact, the charging argument described in the proof idea works only if, ideally, all calls of Med-Elim are correct. Since actually some Med-Elim may return incorrectly, we have to slightly modify the charging method. Nevertheless, we will show that this difference only incurs a reasonably small cost in expectation.

We first claim that

$$T_\infty \le 2\sum_{i,j} |G_j| \cdot T_{i,j}. \tag{6}$$

In other words, the total cost we charge is indeed an upper bound on T_∞. Note that the contribution of round i to T_∞ is $|S_i|\varepsilon_i^{-2}\left[\ln(H/(|S_r|\varepsilon_r^{-2})) + \ln\delta^{-1} + \ln(\hat{H}_t/H)\right]$, while its contribution to the right-hand side of (6) is

$$\begin{aligned}
2\sum_j |G_j| \cdot T_{i,j} &= 2\sum_j |G_j| \cdot \varepsilon_i^{-2}\left(\ln(H/(|G_j|\varepsilon_i^{-2})) + \ln\delta^{-1} + \ln(\hat{H}_t/H)\right) \\
&\ge 2|G_{\ge r_i}| \cdot \varepsilon_i^{-2}\left[\ln(H/(|S_r|\varepsilon_r^{-2})) + \ln\delta^{-1} + \ln(\hat{H}_t/H)\right] \\
&\ge |S_i|\varepsilon_i^{-2}\left[\ln(H/(|S_r|\varepsilon_r^{-2})) + \ln\delta^{-1} + \ln(\hat{H}_t/H)\right].
\end{aligned}$$

Then identity (6) directly follows from a summation on i.

Then we bound the expectation of each $T_{i,j}$. When $i \le j$, we have the trivial bound

$$\mathrm{E}[T_{i,j}] \le \varepsilon_i^{-2}\left(\ln\frac{H}{|G_j|\varepsilon_i^{-2}} + \ln\delta^{-1} + \ln\frac{\hat{H}_t}{H}\right).$$

When $i > j$, we bound the probability that $T_{i,j} > 0$. By definition, $T_{i,j} > 0$ if and only if $r_i \le j$, where r_i is the largest integer that satisfies $|G_{\ge r_i}| \ge 0.5|S_i|$. It follows that $T_{i,j} > 0$ only if $|G_{\ge j+1}| < 0.5|S_i|$.

Observe that in order to have $|G_{\ge j+1}| < 0.5|S_i|$, Med-Elim must return incorrectly between round $j+1$ and round $i-1$. In fact, suppose towards a contradiction that Med-Elim is correct in round $k \in [j+1, i-1]$. Then we have

$$|G_{\ge j+1}| \ge |G_{\ge k}| \ge |S_{k+1} \cap G_{\ge k}| > 0.5|S_{k+1}| \ge 0.5|S_i|,$$

a contradiction. Here the third step is due to the fact that when Elimination returns correctly at round k, the fraction of arms in S_{k+1} with gap greater than 2^{-k} is less than 0.1.

Note that for each specific round, the probability that Med-Elim returns incorrectly is at most 0.01. Thus, the probability that $T_{i,j} > 0$ for $i > j$ is upper bounded by 0.01^{i-j-1}. Therefore,

$$\mathrm{E}[T_{i,j}] \le 0.01^{i-j-1}\varepsilon_i^{-2}\left(\ln\frac{H}{|G_j|\varepsilon_i^{-2}} + \ln\delta^{-1} + \ln\frac{\hat{H}_t}{H}\right).$$

It remains to sum up the upper bounds of $\mathrm{E}[T_{i,j}]$ to yield our bound of $\mathrm{E}[T_\infty]$.

$$\mathrm{E}[T_\infty] \le 2\sum_{i,j}|G_j| \cdot \mathrm{E}[T_{i,j}] = 2\sum_{i\le j}|G_j| \cdot \mathrm{E}[T_{i,j}] + 2\sum_{i>j}|G_j| \cdot \mathrm{E}[T_{i,j}].$$

Here the first part can be bounded by

$$\begin{aligned}2\sum_{i\le j}|G_j|\cdot \mathrm{E}[T_{i,j}] &\le 2\sum_{j}\sum_{i=1}^{j}|G_j|\cdot 4^i\left(\ln\frac{H}{|G_j|4^i}+\ln\delta^{-1}+\ln\frac{\hat{H}_t}{H}\right)\\ &\le 4\sum_{j}|G_j|\cdot 4^j\left(\ln\frac{H}{|G_j|4^j}+\ln\delta^{-1}+\ln\frac{\hat{H}_t}{H}\right)\\ &\le 4\sum_{j}\left(H_j\ln\frac{H}{H_j/4}+H_j\ln\delta^{-1}+H_j\ln\frac{\hat{H}_t}{H}\right)\\ &\le 8H\left(\mathsf{Ent}+\ln\delta^{-1}+\ln\frac{\hat{H}_t}{H}\right).\end{aligned}$$

The second part can be bounded similarly.

$$\begin{aligned}2\sum_{i>j}|G_j|\cdot \mathrm{E}[T_{i,j}] &\le 2\sum_{j}\sum_{i=j+1}^{\infty}0.01^{i-j-1}|G_j|\cdot 4^i\left(\ln\frac{H}{|G_j|4^i}+\ln\delta^{-1}+\ln\frac{\hat{H}_t}{H}\right)\\ &\le 4\sum_{j}|G_j|\cdot 4^j\left(\ln\frac{H}{|G_j|4^j}+\ln\delta^{-1}+\ln\frac{\hat{H}_t}{H}\right)\\ &\le 4\sum_{j}\left(H_j\ln\frac{H}{H_j/4}+H_j\ln\delta^{-1}+H_j\ln\frac{\hat{H}_t}{H}\right)\\ &\le 8H\left(\mathsf{Ent}+\ln\delta^{-1}+\ln\frac{\hat{H}_t}{H}\right).\end{aligned}$$

In fact, the crucial observation for both the two inequalities above is that the summation decreases exponentially as i becomes farther away from j. The lemma directly follows. ■

Finally, we prove Lemma B.6, which is restated below. Recall that we abuse the notation a little bit by assuming $A_1 \in G_\infty$ and $\Delta_{[1]}^{-2} = +\infty$.

Lemma B.6. (restated) *Suppose that $s \in \{2, 3, \ldots, n\}$ and $r^* \in \mathbb{N} \cup \{\infty\}$ satisfy $A_{s-1} \in G_{r^*}$. When $\mathsf{Entropy\text{-}Elimination}$ runs on parameter $\hat{H}_t < \Delta_{[s-1]}^{-2}$, the probability that there exists a call of procedure $\mathsf{Elimination}$ that returns incorrectly before round r^* is upper bounded by*

$$3000s\left(\sum_{i=s}^{n}\Delta_{[i]}^{-2}\right)\delta^2/\hat{H}_t.$$

Proof [Proof of Lemma B.6] Recall that $A_{s-1} \in G_{r^*}$. Suppose $A_s \in G_{r'}$. Suppose that we are at the beginning of round r of Entropy-Elimination and the subset of arms that have not been removed is $S_r = S$. Moreover, we assume that the optimal arm, A_1, is still in S_r. Let $P(r, S)$ denote the probability that some call of procedure Elimination returns incorrectly in round $r, r+1, \ldots, r^*-1$.

As in the proof of Lemma B.4, we bound $P(r,S)$ by induction using the potential function method. Define

$$C(r,S) = \sum_{i=r-1}^{r'} |S \cap G_i| \sum_{j=r}^{i+1} \varepsilon_j^{-2} + (s-1)\sum_{j=r}^{r'+2} \varepsilon_j^{-2}$$

and $M(r,S) = |S \cap G_{\le r-2}|$. Then it holds that for $1 \le r \le r'+1$,

$$C(r,S) - C(r+1,S) = \sum_{i=r-1}^{r'} |S \cap G_i|\varepsilon_r^{-2} + (s-1)\varepsilon_r^{-2} \ge (|S \cap G_{\ge r-1}| + 1)\varepsilon_r^{-2}.$$

We prove by induction that

$$P(r,S) \le \left(128C(r,S) + 16M(r,S)\varepsilon_r^{-2}\right)\delta^2/\hat{H}. \tag{7}$$

We first prove the base case at round $r'+2$. If $r'+2 \ge r^*$, the bound holds trivially. Otherwise, we consider the ratio

$$\alpha = |S_{r'+2} \cap \{A_s, A_{s+1}, \ldots, A_n\}|/|S_{r'+2}|,$$

which is the fraction of arms at round $r'+2$ that are strictly worse than A_{s-1}. Let r_0 be the first round after $r'+2$ (inclusive) in which Med-Elim returns correctly. If Frac-Test returns False in round r_0, according to Fact 5.3 and the correctness of Frac-Test conditioning on event $\mathcal{E}_1$, we have $\alpha \le \theta_{r_0}$. Consequently, in each of the following rounds (say, round $r > r_0$), Frac-Test always returns False since $\alpha \le \theta_{r_0} \le \theta_{r-1}$, and Elimination will never be called before round r^*. Note that it is crucial that the threshold interval of Frac-Test in diffrent rounds are disjoint. For the other case, suppose Frac-Test returns True and we call Elimination in round r_0. Then after that, assuming Elimination returns correctly, the fraction of arms worse than A_{s-1} will be smaller than 0.1. It also follows that Elimination will never be called after round r_0. Therefore, Elimination is called at most once between round $r'+2$ and r^*-1, and it can only be called at round r_0. Note that for $r \ge r'+2$, $\Pr[r_0 = r] \le 0.01^{r-r'-2}$. A direct summation on all possible values of r_0 yields

$$\begin{aligned} P(r'+2,S) &\le \sum_{r=r'+2}^{r^*-1} \Pr[r_0 = r] \cdot \delta'_r \\ &= \sum_{r=r'+2}^{r^*-1} 0.01^{r-r'-2} \cdot 4|S|\varepsilon_r^{-2}\delta^2/\hat{H} \\ &\le \left(4|S|\varepsilon_{r'+2}^{-2}\delta^2/\hat{H}\right)\sum_{k=0}^{\infty} 0.01^k 4^k \\ &\le 5|S|\varepsilon_{r'+2}^{-2}\delta^2/\hat{H}. \end{aligned}$$

Note that $C(r'+2,S) = (s-1)\varepsilon_{r'+2}^{-2}$, $M(r'+2,S) = |S \cap G_{\le r'}|$ and $|S| \le |S \cap G_{\le r'}| + (s-1)$. Thus

$$\begin{aligned} P(r'+2,S) &\le 5(|S \cap G_{\le r'}| + s - 1)\varepsilon_{r'+2}^{-2}\delta^2/\hat{H} \\ &\le \left(128C(r'+2,S) + 16M(r'+2,S)\varepsilon_{r'+2}^{-2}\right)\delta^2/\hat{H}, \end{aligned}$$

which proves the base case.

Then we proceed to the induction step. Again, we consider whether Med-Elim returns correctly and whether Frac-Test returns True. Let $N_{\mathsf{cur}} = |S \cap G_{r-1}|$ and $N_{\mathsf{big}} = |S \cap G_{\geq r}|$. Again, we denote $M(r, S)$ by N_{sma} for brevity. Note that S_{r+1} is the set of arms that survive round r.

Case 1: Med-Elim returns correctly and Frac-Test returns True.

In this case, Elimination is called with confidence level δ'_r. Then the conditional probability that some Elimination returns incorrectly in this case is bounded by

$$\begin{aligned}
&P(r+1, S_{r+1}) + \delta'_r \\
\leq& \left[128C(r+1, S_{r+1}) + 16M(r+1, S_{r+1})\varepsilon_{r+1}^{-2} + 4|S|\varepsilon_r^{-2}\right] \delta^2/\hat{H} \\
\leq& \left[128C(r+1, S) + 64(N_{\mathsf{sma}} + N_{\mathsf{cur}})\varepsilon_r^{-2}/10 + 4|S|\varepsilon_r^{-2}\right] \delta^2/\hat{H} \\
=& \left[128C(r, S) - 128(N_{\mathsf{cur}} + N_{\mathsf{big}} + s - 1)\varepsilon_r^{-2} + (6.4N_{\mathsf{sma}} + 6.4N_{\mathsf{cur}} + 4|S|)\varepsilon_r^{-2}\right] \delta^2/\hat{H} \\
\leq& [128C(r, S) + 10.4M(r, S)\varepsilon_r^{-2}]\delta^2/\hat{H}.
\end{aligned}$$

Case 2: Med-Elim returns correctly and Frac-Test returns False.

Since Frac-Test returns False, according to Fact 5.3 and Observation A.4, we have $N_{\mathsf{sma}} \leq |S| - N_{\mathsf{sma}} = N_{\mathsf{cur}} + N_{\mathsf{big}} + 1$. Then the conditional probability in this case is bounded by

$$\begin{aligned}
P(r+1, S) &\leq [128C(r+1, S) + 16M(r+1, S)\varepsilon_{r+1}^{-2}]\delta^2/\hat{H} \\
&\leq [128C(r, S) - 128(N_{\mathsf{cur}} + N_{\mathsf{big}} + s - 1)\varepsilon_r^{-2} + 64(N_{\mathsf{sma}} + N_{\mathsf{cur}})\varepsilon_r^{-2}]\delta^2/\hat{H} \\
&\leq [128C(r, S) + (64N_{\mathsf{sma}} - 64N_{\mathsf{cur}} - 128N_{\mathsf{big}} - 128(s-1))\varepsilon_r^{-2}]\delta^2/\hat{H} \\
&\leq 128C(r, S)\varepsilon_r^{-2}\delta^2/\hat{H}.
\end{aligned}$$

Here the last step follows from $64N_{\mathsf{sma}} - 64N_{\mathsf{cur}} - 128N_{\mathsf{big}} - 128(s-1) \leq 64(N_{\mathsf{sma}} - N_{\mathsf{cur}} - N_{\mathsf{big}} - 1) \leq 0$.

Case 3: Med-Elim returns incorrectly.

In this case, the worst scenario is that we call Elimination with confidence $\delta'_r \leq 4|S|\varepsilon_r^{-2}\delta^2/\hat{H}$, yet no arms are removed. So the conditional probability in this case is bounded by

$$\begin{aligned}
&P(r+1, S) + 4|S|\varepsilon_r^{-2}\delta^2/\hat{H} \\
\leq& \left[128C(r+1, S) + 16M(r+1, S)\varepsilon_{r+1}^{-2} + 4|S|\varepsilon_r^{-2}\right] \delta^2/\hat{H} \\
\leq& [128C(r, S) - 128(N_{\mathsf{cur}} + N_{\mathsf{big}} + s - 1)\varepsilon_r^{-2} + 64(N_{\mathsf{sma}} + N_{\mathsf{cur}})\varepsilon_r^{-2} + 4(N_{\mathsf{sma}} + N_{\mathsf{cur}} + N_{\mathsf{big}} + 1)\varepsilon_r^{-2}]\delta^2/\hat{H} \\
\leq& [128C(r, S) + (68N_{\mathsf{sma}} - 60N_{\mathsf{cur}} - 124N_{\mathsf{big}} - 124)\varepsilon_r^{-2}]\delta^2/\hat{H} \\
\leq& \left[128C(r, S) + 68M(r, S)\varepsilon_r^{-2}\right] \delta^2/\hat{H}.
\end{aligned}$$

Recall that Case 3 happens with probability at most 0.01. Thus we have:

$$\begin{aligned}
P(r, S) &\leq 0.01 \left[128C(r, S) + 68M(r, S)\varepsilon_r^{-2}\right] \delta^2/\hat{H} + 0.99 \left[128C(r, S) + 10.4M(r, S)\varepsilon_r^{-2}\right] \delta^2/\hat{H} \\
&\leq \left[128C(r, S) + 16M(r, S)\varepsilon_r^{-2}\right] \delta^2/\hat{H}.
\end{aligned}$$

The induction is completed. It follows from (7) that

$$\begin{aligned}
P(1,S) &\le 128\left[\sum_{i=0}^{r'}|G_i|\sum_{j=1}^{i+1}\varepsilon_j^{-2}+(s-1)\sum_{j=1}^{r'+2}\varepsilon_j^{-2}\right]\delta^2/\hat{H}\\
&\le 128\left[(4/3)\sum_{i=0}^{r'}|G_i|4^{i+1}+(4/3)(s-1)4^{r'+2}\right]\delta^2/\hat{H}\\
&\le 128\left[(16/3)\sum_{i=s}^{n}\Delta_{[i]}^{-2}+(64/3)(s-1)4^{r'}\right]\delta^2/\hat{H}\\
&\le 3000s\left(\sum_{i=s}^{n}\Delta_{[i]}^{-2}\right)\delta^2/\hat{H}.
\end{aligned}$$

■

B.3. Proof of Lemma B.1

Recall that $t_{\max} = \lfloor \log_{100} H \rfloor - 2$ and $t'_{\max} = \lceil \log_{100}[H(\mathsf{Ent} + \ln \delta^{-1})\delta^{-1}] \rceil + 2$. We restate and prove Lemma B.1 in the following.

Lemma B.1. (restated) *With probability $1 - \delta/3$ conditioning on event $\mathcal{E}_1$, Complexity-Guessing halts before or at iteration $t'_{\max}$ and it never returns a sub-optimal arm between iteration $t_{\max} + 1$ and $t'_{\max}$.*

The high-level idea of the proof is to construct three other "good events" $\mathcal{E}_2$, $\mathcal{E}_3$ and $\mathcal{E}_4$. We show that each event happens with high probability conditioning on $\mathcal{E}_1$. Moreover, events $\mathcal{E}_1$ through $\mathcal{E}_4$ together imply the desired event.

Proof Recall that $t_{\max} = \lfloor \log_{100} H \rfloor - 2$ and $t'_{\max} = \lceil \log_{100}[H(\mathsf{Ent} + \ln \delta^{-1})\delta^{-1}] \rceil + 2$. Let $\mathcal{E}_2$ denote the following event: for all t such that $t \ge t_{\max} + 1$ and $\hat{H}_t < 100^3 H$, Entropy-Elimination either rejects or outputs the optimal arm. Since $\hat{H}_{t_{\max}+1} = 100^{t_{\max}+1} \ge H/10000$, there are at most $\log_{100}[100^3 H/(H/10000)] + 1 = 6$ different values of such $\hat{H}_t$. For each $\hat{H}_t$, the probability of returning a sub-optimal arm is bounded by the probability that the optimal arm is deleted, which is in turn upper bounded by δ^2 as a corollary of Lemma B.9 proved in the following section.

Thus, by a union bound,

$$\Pr[\mathcal{E}_2|\mathcal{E}_1] \ge 1 - 6\delta^2.$$

Let $\mathcal{E}_3$ be the event that for all $\hat{H}_t$ such that $t \le t'_{\max}$ and $\hat{H}_t \ge 100^3 H$ (or equivalently, $\lceil \log_{100} H \rceil + 3 \le t \le t'_{\max}$), Entropy-Elimination never returns an incorrect answer. In fact, in order for Entropy-Elimination to return incorrectly, some call of Elimination must be wrong. Thus we may apply Lemma B.6 to bound the probability of $\mathcal{E}_3$. Specifically, we apply Lemma B.6 with

$s = 2$. Then we have

$$\begin{aligned}
\Pr[\mathcal{E}_3|\mathcal{E}_1] &\ge 1 - \sum_{t=\lceil \log_{100} H\rceil+3}^{t'_{\max}} \frac{3000s\left(\sum_{i=s}^{n}\Delta_{[i]}^{-2}\right)\delta^2}{\hat{H}_t} \\
&\ge 1 - \sum_{t=\lceil \log_{100} H\rceil+3}^{\infty} \frac{6000H}{100^t}\delta^2 \\
&\ge 1 - \sum_{k=3}^{\infty} \frac{6000}{100^k}\delta^2 \ge 1 - \delta^2/100.
\end{aligned}$$

Here the third step is due to the simple fact that $100^{\lceil \log_{100} H\rceil} \ge H$.

Finally, let $\mathcal{E}_4$ denote the event that when Entropy-Elimination runs on $\hat{H}_{t'_{\max}}$, no Elimination is wrong and the algorithm finally accepts. In order to bound the probability of the last event, we simply apply Markov inequality based on Lemma B.4 and Lemma B.5. Let $\mathcal{E}_0$ be the event that no Elimination is wrong when Entropy-Elimination runs on $\hat{H}_{t'_{\max}}$. Then we have

$$\begin{aligned}
\Pr[\mathcal{E}_4|\mathcal{E}_1] &\ge \Pr[\mathcal{E}_0|\mathcal{E}_1] - \frac{\mathrm{E}[H_\infty|\mathcal{E}_0]}{\hat{H}_{t'_{\max}}} - \frac{\mathrm{E}[T_\infty|\mathcal{E}_0]}{100\hat{H}_{t'_{\max}}} \\
&\ge 1 - \delta^2 - \frac{256H}{100^2 H(\mathsf{Ent} + \ln\delta^{-1})\delta^{-2}} - \frac{16H\left[\mathsf{Ent} + \ln\delta^{-1} + \ln(\hat{H}_{t'_{\max}}/H)\right]}{100^3 H(\mathsf{Ent} + \ln\delta^{-1})\delta^{-2}} \\
&\ge 1 - \delta^2 - \frac{256}{100^2}\delta^2 - \frac{16\left[\mathsf{Ent} + 3\ln\delta^{-1} + \ln(100^2(\mathsf{Ent} + \ln\delta^{-1}))\right]}{100^3(\mathsf{Ent} + \ln\delta^{-1})}\delta^2 \\
&\ge 1 - 2\delta^2.
\end{aligned}$$

Note that conditioning on events $\mathcal{E}_1$ through $\mathcal{E}_4$, Entropy-Elimination never outputs an incorrect answer between iteration $t_{\max} + 1$ and $t'_{\max}$. Moreover, our algorithm terminates before or at iteration $t'_{\max}$. The lemma directly follows from a union bound and the observation that for all $\delta \in (0, 0.01)$,

$$6\delta^2 + \delta^2/100 + 2\delta^2 \le \delta/3.$$

■

Remark B.7 *The last part of the proof implies a more general fact: for fixed $\hat{H}_t$, Entropy-Elimination accepts with probability at least*

$$1 - \delta^2 - \frac{256H}{\hat{H}_t} - \frac{16H(\mathsf{Ent} + \ln\delta^{-1} + \ln(\hat{H}_t/H))}{100\hat{H}_t}.$$

B.4. Mis-deletion of Arms

We prove Lemma B.2 in the following. Again, our analysis in this subsection conditions on event $\mathcal{E}_1$, which guarantees that all calls of Frac-Test and Unif-Sampl in Entropy-Elimination are correct. The high-level idea of the proof is to show that a large proportion of arms will not be accidentally removed before they have contributed a considerable amount to the total complexity. Formally, we define the mis-deletion of arms as follows.

Definition B.8 *An arm $A \in G_r$ is **mis-deleted** in a particular run of Entropy-Elimination, if A is deleted before or at round $r-1$. In particular, the optimal arm is **mis-deleted** if it is deleted in any round.*

The following lemma bounds the probability that a certain collection of arms are all mis-deleted.

Lemma B.9 *For a fixed collection of k arms, the probability that all of them are mis-deleted is at most δ^{2k}.*

Proof Let $S = \{A_1, A_2, \ldots, A_k\}$ be a fixed set of k arms. (Here we temporarily drop the convention that A_i denotes the arm with the i-th largest mean.) For each A_i, let $\mathcal{E}_i^{\text{bad}}$ denote the event that A_i is mis-deleted, and let r_i denote the group that contains A_i (i.e., $A_i \in G_{r_i}$). By definition, $\mu_{A_i} \ge \mu_{[1]} - \varepsilon_{r_i}$.

We start by proving the following fact: suppose Elimination is called with confidence level δ'_r in round r. Then the probability that all arms in S are mis-deleted in round r simultaneously is bounded by ${\delta'}_r^k$.

We assume that $r < r_i$ for all $i = 1, 2, \ldots, k$. Otherwise, if $r \ge r_i$ for some i, then A_i cannot be *mis-deleted* in round r, since the definition of mis-deletion requires that $r < r_i$. To analyze the behaviour of Elimination, we recall that each run of Elimination consists of several stages. (Here we use the term “stage” for an iteration of Elimination, while the term for Entropy-Elimination is “round”.) In each stage, procedure Unif-Sampl is called at line 6 to estimate the means of the arms that have not been eliminated. Let r_i^{bad} denote the stage in which A_i gets deleted.

Recall that d_r^{high} is the upper threshold used in Elimination in round r. According to Observation A.5,

$$d_r^{\text{high}} \le \mu_{[1]}(S_r) - 0.5\varepsilon_r = \mu_{[1]}(S_r) - 2^{-(r+1)} \le \mu_{[1]} - \varepsilon_{r_i} \le \mu_{A_i}.$$

Here the third step follows from our assumption that $r < r_i$. In order for Elimination to eliminate an arm A_i with mean greater than d^{high} in stage r_i^{bad}, the Unif-Sampl subroutine must return an incorrect estimation for A_i (i.e., $|\hat{\mu}_{A_i} - \mu_{A_i}| > (d^{\text{high}} - d^{\text{mid}})/2$), which happens with probability at most $\delta'_r / \left(10 \cdot 2^{r_i^{\text{bad}}}\right)$. Since the samples taken on different arms are independent, the events that Unif-Sampl returns incorrect estimates for different arms are also independent, and it follows that the probability that each arm A_i is removed at stage r_i^{bad} is bounded by $\prod_{i=1}^{k} \left(\delta'_r / \left(10 \cdot 2^{r_i^{\text{bad}}}\right)\right)$.

Therefore, the probability that all the k arms in S are mis-deleted in Elimination is upper bounded by

$$\begin{aligned}
&\sum_{r_1^{\text{bad}}=1}^{\infty} \sum_{r_2^{\text{bad}}=1}^{\infty} \cdots \sum_{r_k^{\text{bad}}=1}^{\infty} \prod_{i=1}^{k} \left(\delta'_r / (10 \cdot 2^{r_i^{\text{bad}}})\right) \\
=&\prod_{i=1}^{k} \left[\sum_{r_i^{\text{bad}}=1}^{\infty} \left(\delta'_r / \left(10 \cdot 2^{r_i^{\text{bad}}}\right)\right) \right] \\
\le&\prod_{i=1}^{k} \delta'_r = {\delta'}_r^k.
\end{aligned}$$

Then we start with the proof of the lemma. Suppose that we are at the beginning of round r. m arms among S are still in S_r, while the sum of confidence levels allocated in the previous rounds is δ' (i.e., $\delta' = \sum_{i=1}^{r-1} \delta'_i$). Let $P(r, \delta', m)$ denote the probability that all the m remaining arms are mis-deleted in the future. We prove by induction that

$$P(r, \delta', m) \le (\delta^2 - \delta')^m. \tag{8}$$

Recall that the number of rounds that Entropy-Elimination lasts is bounded by ct according to Observation A.1. Thus when $r = \lceil ct \rceil + 1$, we have $P(r, \delta', m) = 0$. Observe that δ' never exceeds δ^2 according to the behaviour of Entropy-Elimination. Therefore (8) holds for the base case. Now we proceed to the induction step. If Elimination is not called in round r, by induction hypothesis we have

$$P(r, \delta', m) \le P(r+1, \delta', m) \le (\delta^2 - \delta')^m,$$

which proves inequality (8). If, on the other hand, Elimination is called with confidence δ'_r. We observe that by the claim we proved above, the probability that exactly j arms among the m arms are mis-deleted is at most $\binom{m}{j}\delta'^j_r$. Thus by a simple summation,

$$P(r, \delta', m) \le \sum_{j=0}^{m} \binom{m}{j} \delta'^j_r \cdot P(r+1, \delta' + \delta'_r, m-j) \le \sum_{j=0}^{m} \binom{m}{j} \delta'^j_r (\delta^2 - \delta' - \delta'_r)^{m-j} = (\delta^2 - \delta')^m,$$

which completes the induction step.

Finally, the lemma directly follows from (8) by plugging in $r = 1$, $\delta' = 0$ and $m = k$. ■

Remark B.10 *Let $\mathcal{E}_i^{\mathrm{bad}}$ denote the event that A_i is mis-deleted. Note that although the events $\{\mathcal{E}_i^{\mathrm{bad}}\}$ are not independent, we can still obtain an exponential bound (i.e., δ^{2k}) on the probability that k such events happen simultaneously. We call such events* ***quasi-independent*** *to reflect this property. Formally, a collection of n events $\{\mathcal{E}_i\}_{i=1}^n$ are δ-quasi-independent, if for all $1 \le k \le n$ and $1 \le a_1 < a_2 < \cdots < a_k \le n$, we have*

$$\Pr[\mathcal{E}_{a_1} \cap \mathcal{E}_{a_2} \cap \cdots \cap \mathcal{E}_{a_k}] \le \delta^k.$$

Then the collection of events $\{\mathcal{E}_i^{\mathrm{bad}}\}$ are δ^2-quasi-independent.

The following lemma proves a generalized Chernoff bound for quasi-independent events.

Lemma B.11 *Suppose $v_1, v_2, \ldots, v_n > 0$. $\{Y_i\}_{i=1}^n$ is a collection of random variables, where the support of Y_i is $\{0, v_i\}$. Moreover, the collection of events $\{Y_i = v_i\}$ are δ-quasi-independent. Let $(S_1, S_2, \ldots, S_m)$ be a partition of $\{1, 2, \ldots, n\}$ such that $\sum_{j \in S_i} v_j \le 1$ for all i. Define $X_i = \sum_{j \in S_i} Y_j$. Let $X = \frac{1}{m}\sum_{i=1}^m X_i$ and $p = \frac{\delta}{m}\sum_{i=1}^n v_i$. Then for all $q \in (p, 1)$,*

$$\Pr[X \ge q] \le e^{-mD(q||p)},$$

where

$$D(x||y) = x\ln(x/y) + (1-x)\ln[(1-x)/(1-y)]$$

is the relative entropy function.

Proof Let $p_i = \delta \sum_{j \in S_i} v_j$. Then $p = \frac{1}{m}\sum_{i=1}^{m} p_i$. For $t > 0$, we have

$$\Pr[X \ge q] = \Pr[e^{tmX} \ge e^{tmq}] \le \frac{\mathrm{E}[e^{tmX}]}{e^{tmq}}.$$

To bound $\mathrm{E}[e^{tmX}]$, we consider a collection of *independent* random variables $\tilde{Y}_1, \tilde{Y}_2, \ldots, \tilde{Y}_n$ defined by $\Pr[\tilde{Y}_i = v_i] = \delta$ and $\Pr[\tilde{Y}_i = 0] = 1 - \delta$. Define $\tilde{X}_i = \sum_{j \in S_i} \tilde{Y}_j$ for $i = 1, 2, \ldots, m$, and $\tilde{X} = \frac{1}{m}\sum_{i=1}^{m} \tilde{X}_i$. Note that each term in the Taylor expansion of e^{tmX} can be written as $\alpha \prod_{i=1}^{l} Y_{n_i}$, where $l \ge 0$, $(n_1, n_2, \ldots, n_l) \in \{1, 2, \ldots, n\}^l$, and $\alpha = t^l/(l!) > 0$. The corresponding term in $e^{tm\tilde{X}}$ is then $\alpha \prod_{i=1}^{l} \tilde{Y}_{n_l}$. Let $U = |\{n_i : i \in \{1, 2, \ldots, l\}\}|$ denote the set of distinct numbers among $n_1, n_2, \ldots, n_l$. We have

$$\mathrm{E}\left[\prod_{i=1}^{l} Y_{n_l}\right] = \Pr[Y_i = v_i \text{ for all } i \in U] \cdot \prod_{i=1}^{l} v_{n_l} \le \delta^{|U|} \prod_{i=1}^{l} v_{n_l} = \mathrm{E}\left[\prod_{i=1}^{l} \tilde{Y}_{n_l}\right].$$

Summing over all terms in the expansion yields

$$\mathrm{E}\left[e^{tmX}\right] \le \mathrm{E}\left[e^{tm\tilde{X}}\right] = \prod_{i=1}^{m} \mathrm{E}\left[e^{t\tilde{X}_i}\right].$$

Here the last step holds since $\{\tilde{X}_i\}$ are independent. Note that since $\tilde{X}_i \in [0, 1]$, it follows from Jensen's inequality that

$$\mathrm{E}\left[e^{t\tilde{X}_i}\right] \le \mathrm{E}\left[e^t \tilde{X}_i + 1 - \tilde{X}_i\right] = p_i e^t + 1 - p_i.$$

Then

$$\mathrm{E}\left[e^{tmX}\right] \le \prod_{i=1}^{m} (p_i e^t + 1 - p_i) \le (pe^t + 1 - p)^m.$$

Recall that $p = \frac{1}{m}\sum_{i=1}^{m} p_i$. Here the last step follows from Jensen's inequality and the concavity of $\ln(e^t x + 1 - x)$ for $t > 0$.

By setting $t = \ln \frac{q(1-p)}{p(1-q)}$, we have

$$\Pr[X \ge q] \le \frac{\mathrm{E}[e^{tmX}]}{e^{tmq}} \le \left[\frac{pe^t + 1 - p}{e^{tq}}\right]^m = e^{-mD(q||p)}.$$

■

The following lemma states that if a collection of arms with a considerable amount of total complexity are not mis-deleted, Entropy-Elimination rejects $\hat{H}$.

Lemma B.12 *S is a set of sub-optimal arms with complexity $H(S) > \hat{H}$. Let $r^* = \max_{A \in S} \left\lfloor \log_2 \Delta_A^{-1} \right\rfloor$. If in a particular run of Entropy-Elimination, no arm in S is mis-deleted and there exists an arm A^* outside S with $\mu_{A^*} \ge \max_{A \in S} \mu_A$ such that A^* is not deleted in the first $r^* - 1$ rounds, then $\hat{H}$ is rejected in that run.*

Proof Suppose $S = \{A_1, A_2, \ldots, A_k\}$ and $A_i \in G_{r_i}$. Without loss of generality, $\mu_{A_1} \le \mu_{A_2} \le \cdots \le \mu_{A_k}$. By definition of r^*, we have $r^* = \max_{1 \le i \le k} r_i = r_k$. According to our assumption, both A_k and A^* are not deleted in the first $r^* - 1$ rounds. Thus Entropy-Elimination does not accept in the first r^* rounds.

Suppose for contradiction that $\hat{H}$ is not rejected by Entropy-Elimination in a particular run. Define $\mathcal{R} = \{r \in [1, r^* - 1] : \text{Elimination is called in round } r\}$. Let $N_1 = \{i \in [k] : \exists r \in \mathcal{R}, r \ge r_i\}$ and $N_2 = [k] \setminus N_1$. For each $i \in N_1$, since A_i is not mis-deleted, $A_i \in S_{r_i}$. Define $r'_i = \min\{r \in \mathcal{R} : r \ge r_i\}$ as the first round after r_i (inclusive) in which Elimination is called. It follows that $A_i \in S_{r'_i}$. At round r'_i of Entropy-Elimination, $H_{r'_i+1}$ is set to $H_{r'_i} + 4|S_{r'_i}|\varepsilon_{r'_i}^{-2}$. Therefore we can "charge" A_i a cost of $4\varepsilon_{r'_i}^{-2} = \varepsilon_{r'_i+1}^{-2}$. It follows that H_{r^*} is at least the total cost that arms in N_1 are charged, $\sum_{i \in N_1} \varepsilon_{r'_i+1}^{-2}$.

For each $i \in N_2$, we have $A_i \in S_{r_i}$ and $S_{r_i} = S_{r^*}$. Thus it holds that $|S_{r^*}| \ge |N_2|$. When the if-statement in Entropy-Elimination is checked in round r^*, we have

$$H_{r^*} + 4|S_{r^*}|\varepsilon_{r^*}^{-2} \ge \sum_{i \in N_1} \varepsilon_{r'_i+1}^{-2} + N_2 \varepsilon_{r^*+1}^{-2} \ge \sum_{i=1}^{k} \varepsilon_{r_i+1}^{-2} \ge \sum_{i=1}^{k} \Delta_{A_i}^{-2} = H(S) > \hat{H}.$$

Here the second step follows from $r'_i \ge r_i$ and $r^* \ge r_i$, while the third step follows from $\Delta_{A_i} \ge 2^{-(r_i+1)}$. Therefore Entropy-Elimination rejects in round r^*, a contradiction. ■

B.5. Proof of Lemma B.2

Lemma B.2 is restated below. Recall that $t_{\max} = \lfloor \log_{100} H \rfloor - 2$.

Lemma B.2. (restated) *With probability $1 - \delta/3$ conditioning on event $\mathcal{E}_1$, Complexity-Guessing never returns a sub-optimal arm in the first $t_{\max}$ iterations.*

The high-level idea of the proof is simple. For each $\hat{H}_t$, we identify a collection of near-optimal "crucial arms". By Lemma B.9, the probability that all "crucial arms" are mis-deleted is small, thus we may assume that at least one crucial arm survives. This crucial arm serves as A^* in Lemma B.12. Then according to Lemma B.12, in order for Entropy-Elimination to accept $\hat{H}_t$, it must mis-delete a collection of "non-crucial" arms with a significant fraction of complexity. The probability of this event can also be bounded by using the generalized Chernoff bound proved in Lemma B.11.

The major technical difficulty is the choice of "crucial arms". We deal the following three cases separately: (1) $\hat{H}_t$ is greater than $\Delta_{[2]}^{-2}$, the complexity of the arm with the second largest mean; (2) $\hat{H}_t$ is between $\Delta_{[s]}^{-2}$ and $\Delta_{[s-1]}^{-2}$ for some $3 \le s \le n$; and (3) $\hat{H}_t$ is smaller than $\Delta_{[n]}^{-2}$. We bound the probability that the lemma is violated in each case, and sum them up using a union bound.

Proof [Proof of Lemma B.2]

Case 1: $\Delta_{[2]}^{-2} \le \hat{H}_t \le \hat{H}_{t_{\max}}$.

We first deal with the case that $\hat{H}_t$ is relatively large. We partition the sequence of sub-optimal arms $A_2, A_3, \ldots, A_n$ into contiguous blocks $B_1, B_2, \ldots, B_m$ such that the total complexity in each block B_i, denoted by $H(B_i) = \sum_{A \in B_i} \Delta_A^{-2}$, is between $\Delta_{[2]}^{-2}$ and $3\Delta_{[2]}^{-2}$. To construct such a partition, we append arms to the current block one by one from A_2 to A_n. When the complexity of the current block exceeds $\Delta_{[2]}^{-2}$, we start with another block. Clearly, the complexity of each resulting block is upper bounded by $2\Delta_{[2]}^{-2}$. Note that the last block may have a complexity less than $\Delta_{[2]}^{-2}$. In that case, we simply merge it into the second last block. As a result, the total complexity of every block is in $\left[\Delta_{[2]}^{-2}, 3\Delta_{[2]}^{-2}\right]$. It follows that $H \in \left[m\Delta_{[2]}^{-2}, 3m\Delta_{[2]}^{-2}\right]$.

For brevity, let $B_{\le i}$ denote $B_1 \cup B_2 \cup \cdots \cup B_i$ and $B_{<i} = B_{\le i-1}$. Since $H(B_1) = \Delta_{[2]}^{-2} \le \hat{H}_t < H = H(B_{\le m})$, there exists a unique integer $k \in [2, m]$ that satisfies $H(B_{<k}) \le \hat{H}_t < H(B_{\le k})$. Then we have $\hat{H}_t \in \left[(k-1)\Delta_{[2]}^{-2}, 3k\Delta_{[2]}^{-2}\right]$. Since $B_{\le k}$ contains at least k arms, it follows from Lemma B.9 that with probability $1 - \delta^{2k}$, at least one arm in $B_{\le k}$ is not mis-deleted. Recall that by Lemma B.12, Entropy-Elimination accepts $\hat{H}_t$ only if either of the following two events happens: (a) no arm in $B_{\le k}$ survives, which happens with probability δ^{2k}; (b) a collection of arms among $B_{>k}$ with total complexity of at least $H(B_{>k}) - \hat{H}$ are mis-deleted.

For $i = 2, 3, \ldots, n$, define $v_i = \Delta_{[i]}^{-2}/(3\Delta_{[2]}^{-2})$ and $Y_i = v_i \cdot \mathbb{I}[A_i \text{ is mis-deleted}]$. For $i = 1, 2, \ldots, m$, X_i is defined as

$$X_i = \sum_{A_j \in B_i} Y_j = \sum_{A \in B_i} \frac{\Delta_A^{-2}}{3\Delta_{[2]}^{-2}} \cdot \mathbb{I}[A \text{ is mis-deleted}].$$

In other words, X_i is the total complexity of the arms in block B_i that are mis-deleted, divided by a constant $3\Delta_{[2]}^{-2}$. Recall that $H(B_i) \le 3\Delta_{[2]}^{-2}$, so X_i is between 0 and 1. Let

$$X = \frac{1}{m}\sum_{i=1}^{m} X_i = \frac{1}{3m\Delta_{[2]}^{-2}} \sum_{i=2}^{n} \Delta_{[i]}^{-2} \cdot \mathbb{I}[A_i \text{ is mis-deleted}]$$

denote the mean of these random variables. Since the events of mis-deletion of arms are δ^2-quasi-independent, we may apply Lemma B.11. Note that

$$p = \frac{\delta^2}{m}\sum_{i=2}^{n} v_i = \frac{\delta^2}{m}\sum_{i=2}^{n} \frac{\Delta_{[i]}^{-2}}{3\Delta_{[2]}^{-2}} = \frac{H\delta^2}{3m\Delta_{[2]}^{-2}} \le \delta^2.$$

Here the last step applies $H \le 3m\Delta_{[2]}^{-2}$. On the other hand, conditioning on event (b) (i.e., a collection of arms with total complexity $H(B_{>k}) - \hat{H}$ are mis-deleted), we have

$$\begin{aligned} X &= \frac{1}{3m\Delta_{[2]}^{-2}} \sum_{i=2}^{n} \Delta_{[i]}^{-2} \cdot \mathbb{I}[A_i \text{ is mis-deleted}] \\ &\ge \frac{H(B_{>k}) - \hat{H}}{3m\Delta_{[2]}^{-2}} \ge \frac{(m-k)\Delta_{[2]}^{-2} - 3k\Delta_{[2]}^{-2}}{3m\Delta_{[2]}^{-2}} \\ &\ge \frac{m-4k}{3m} \ge \frac{m - 12m/10000}{3m} \ge \frac{1}{6}. \end{aligned}$$

Here the third step follows from $H(B_{>k}) \ge (m-k)\Delta_{[2]}^{-2}$ and $\hat{H} \le 3k\Delta_{[2]}^{-2}$. The last line holds since

$$k\Delta_{[2]}^{-2} \le \hat{H} \le \hat{H}_{t_{\max}} \le H/10000 \le 3m\Delta_{[2]}^{-2}/10000,$$

which implies $k \le 3m/10000$.

According to Lemma B.11, we have

$$\begin{aligned}\Pr[X \ge 1/6] &\le \exp\left(-mD\left(1/6||\delta^2\right)\right)\\ &= \exp\left(-\frac{m}{6}\ln\frac{1}{6\delta^2} - \frac{5m}{6}\ln\frac{5}{6(1-\delta^2)}\right)\\ &\le (6\delta^2)^{m/6}\cdot(6/5)^{5m/6} \le \delta^{m/6}.\end{aligned}$$

Recall that $D(x||y)$ stands for the relative entropy function. The last step follows from $6\delta\cdot(6/5)^5 \le 1$.

Therefore,

$$\Pr\left[\textsf{Entropy-Elimination}\text{ accepts }\hat{H}_t\right] \le \delta^{2k} + \delta^{m/6}. \tag{9}$$

It remains to apply a union bound to (9) for all values of $\hat{H}_t$ in $\left[\Delta_{[2]}^{-2}, \hat{H}_{t_{\max}}\right]$. Recall that $k \ge 2$, and the ratio between different guesses $\hat{H}_t$ is at least 100. It follows that the values of k are distinct for different values of $\hat{H}_t$, and thus the sum of the first term, δ^{2k}, can be bounded by

$$\sum_{k=2}^{\infty}\delta^{2k} = \frac{\delta^4}{1-\delta^2} \le 2\delta^4.$$

For the second term, we note that the number of guesses $\hat{H}_t$ between $\Delta_{[2]}^{-2}$ and $\hat{H}_{t_{\max}}$ is at most

$$t_{\max} - \left\lceil\log_{100}\Delta_{[2]}^{-2}\right\rceil + 1 \le \log_{100} H - 2 - \log_{100}\Delta_{[2]}^{-2} + 1 = \log_{100}\frac{H}{\Delta_{[2]}^{-2}} - 1 \le \log_{100}(3m) - 1.$$

In particular, if $m < 100^2/3$, no $\hat{H}_t$ will fall into $[\Delta_{[2]}^{-2}, t_{\max}]$. Thus we focus on the nontrivial case $m \ge 100^2/3$. Then the sum of the second term $\delta^{m/6}$ can be bounded by

$$\delta^{m/6}\cdot(\log_{100}(3m) - 1) \le \delta^{100^2/18},$$

since $\delta^{m/6}\cdot(\log_{100}(3m)-1)$ decreases on $[100^2/3, +\infty)$ for $\delta \in (0, 0.01)$. Finally, we have

$$\Pr\left[\textsf{Entropy-Elimination}\text{ accepts }\hat{H}_t\text{ for some }\hat{H}_t \in [\Delta_{[2]}^{-2}, H_{t_{\max}}]\right] \le 2\delta^4 + \delta^{100^2/18} \le 3\delta^4.$$

Case 2: $\Delta_{[s]}^{-2} \le \hat{H} < \Delta_{[s-1]}^{-2}$ for some $3 \le s \le n$.

In this case, $\hat{H}$ is between the complexity of A_{s-1} and A_s. Our goal is to prove an upper bound of $\delta^{\Omega(s)}$ on the probability of returning a sub-optimal arm for each specific s. Summing over all s yields a bound on the total probability. Our analysis depends on the ratio between $\hat{H}$ and $\sum_{i=s}^{n}\Delta_{[i]}^{-2}$, the complexity of arms that are worse than A_s. Intuitively, when $\hat{H}$ is greater than the sum (Case 2-1), the contribution of the arms worse than A_s to the complexity is negligible. Thus we have to rely on the fact that the $s-1$ arms with the largest means will not be mis-deleted simultaneously with high probability. On the other hand, when $\hat{H}$ is significantly smaller than the sum (Case 2-2), we may apply the same analysis as in Case 1. Finally, if the value of $\hat{H}$ is between the two cases (Case 2-3), it suffices to prove a relatively loose bound, since the number of possible values is small.

Case 2-1: $\hat{H} > 300000s\sum_{i=s}^{n}\Delta_{[i]}^{-2}$.

In this case, our guess $\hat{H}$ is significantly larger than the total complexity of $A_s, A_{s+1}, \ldots, A_n$, yet $\hat{H}$ is smaller than the complexity of any one among the remaining arms. Thus intuitively, in order to reject $\hat{H}$, Entropy-Elimination should not mis-delete all the first $s-1$ arms. More formally, we have the following fact: in order for Entropy-Elimination to return a sub-optimal arm, it must delete A_1 along with at least $s-3$ arms among $A_2, A_3, \ldots, A_{s-1}$ before round r^*, where r^* is the group that contains A_{s-1}. In fact, since $4\varepsilon_{r^*}^{-2} = 4^{r^*+1} \ge \Delta_{[s-1]}^{-2} \ge \hat{H}_t$, Entropy-Elimination terminates before or at round r^*. If A_1 is not deleted before round r^*, Entropy-Elimination can only return A_1 as the optimal arm, which is correct. If less than $s-3$ arms among $A_2, A_3, \ldots, A_{s-1}$ are deleted before round r^*, for example A_i and A_j are not deleted ($2 \le i < j \le s-1$), then both of them are contained in S_{r^*}. It follows that Entropy-Elimination does not return before round r^*.

We first bound the probability that A_1 is deleted before round r^*. In order for this to happen, some Elimination must return incorrectly. By Lemma B.6, the probability of this event is upper bounded by

$$3000s\left(\sum_{i=s}^{n}\Delta_{[i]}^{-2}\right)\delta^2/\hat{H}_t.$$

In fact, we have a more general fact: the probability that a fixed set of k arms among $\{A_2, A_3, \ldots, A_{s-1}\}$ together with A_1 are deleted before round r^* is bounded by

$$3000s\left(\sum_{i=s}^{n}\Delta_{[i]}^{-2}\right)\delta^2/\hat{H}_t \cdot \delta^{2k} = 3000s\left(\sum_{i=s}^{n}\Delta_{[i]}^{-2}\right)\delta^{2(k+1)}/\hat{H}_t.$$

The proof follows from combining the two inductions in the proof of Lemma B.6 and Lemma B.9, and we omit it here. Since $\{A_2, A_3, \ldots, A_{s-1}\}$ contains $s-2$ subsets of size $s-3$, the probability that Entropy-Elimination returns an incorrect answer on a particular guess $\hat{H}_t$ is at most

$$(s-2)\cdot 3000s\left(\sum_{i=s}^{n}\Delta_{[i]}^{-2}\right)\delta^{2(s-2)}/\hat{H}_t.$$

It remains to apply a union bound on all $\hat{H}_t$ that fall into this case. Recall that $\hat{H}_t > 300000s\sum_{i=s}^{n}\Delta_{[i]}^{-2}$ and $\hat{H}_t$ grows exponentially in t at a rate of 100. Thus the total probability is upper bounded by

$$\sum_{k=0}^{\infty}\frac{3000s\left(\sum_{i=s}^{n}\Delta_{[i]}^{-2}\right)\delta^{2(s-2)}(s-2)}{100^k\cdot 300000s\sum_{i=s}^{n}\Delta_{[i]}^{-2}} = \sum_{k=0}^{\infty}\frac{\delta^{2(s-2)}(s-2)}{100^{k+1}} = \frac{1}{99}\delta^{2(s-2)}(s-2).$$

Case 2-2: $\hat{H} < \sum_{i=s}^{n}\Delta_{[i]}^{-2}/(78s)$.

In this case, we apply the technique in the proof of Case 1. We partition the sequence $A_s, A_{s+1}, \ldots, A_n$ into m consecutive blocks $B_1, B_2, \ldots, B_m$ such that $H(B_i) \in \left[\Delta_{[s]}^{-2}, 3\Delta_{[s]}^{-2}\right]$. Let $B_{\le i}$ denote $B_1 \cup B_2 \cup \cdots \cup B_i$. Since $H(B_1) = \Delta_{[s]}^{-2} \le \hat{H} < \sum_{i=s}^{n}\Delta_{[i]}^{-2}/(78s) < H(B_{\le m})$, there exists a unique integer $k \in [2, m]$ such that $H(B_{<k}) \le \hat{H} < H(B_{\le k})$. It follows that $\hat{H} \in \left[(k-1)\Delta_{[s]}^{-2}, 3k\Delta_{[s]}^{-2}\right]$.

By Lemma B.12, in order for Entropy-Elimination to accept $\hat{H}$, one of the following two events happens: (a) Entropy-Elimination mis-deletes all arms in $B_{\le k} \cup \{A_1, A_2, \ldots, A_{s-1}\}$; (b) the total complexity of mis-deleted arms among $B_{>k}$ is greater than $H(B_{>k}) - \hat{H}$. Since $B_{\le k}$ contains at least k arms, by Lemma B.9, the probability of event (a) is bounded by $\delta^{2(s+k-1)}$.

Again, we bound the probability of event (b) using the generalized Chernoff bound in Lemma B.11. For each $i = s, s+1, \ldots, n$, define $v_i = \Delta_{[i]}^{-2}/(3\Delta_{[s]}^{-2})$ and $Y_i = v_i \cdot \mathbb{I}[A_i \text{ is mis-deleted}]$. Define random variables $\{X_i : i \in \{1, 2, \ldots, m\}\}$ as

$$X_i = \sum_{A_j \in B_i} Y_j = \frac{1}{3\Delta_{[s]}^{-2}} \sum_{A \in B_i} \Delta_A^{-2} \cdot \mathbb{I}[A \text{ is mis-deleted}].$$

Since $H(B_i) \le 3\Delta_{[s]}^{-2}$, X_i is between 0 and 1. Let

$$X = \frac{1}{m}\sum_{i=1}^{m} X_i = \frac{1}{3m\Delta_{[s]}^{-2}} \sum_{i=s}^{n} \Delta_{[i]}^{-2} \cdot \mathbb{I}[A_i \text{ is mis-deleted}]$$

denote the mean of these random variables. Since the events $\{Y_i = v_i\}$ are δ^2-quasi-independent, we may apply Lemma B.11. We have

$$p = \frac{\delta^2}{m}\sum_{i=s}^{n} v_i = \frac{H(B_{\le m})\delta^2}{3m\Delta_{[s]}^{-2}} \le \delta^2.$$

Here the last step applies $H(B_{\le m}) \le 3m\Delta_{[s]}^{-2}$. On the other hand, conditioning on event (b) (i.e., a collection of arms in $B_{>k}$ with total complexity $H(B_{>k}) - \hat{H}$ are mis-deleted), we have

$$\begin{aligned}
X &= \frac{1}{3m\Delta_{[s]}^{-2}} \sum_{i=s}^{n} \Delta_{[i]}^{-2} \cdot \mathbb{I}[A_i \text{ is mis-deleted}] \\
&\ge \frac{H(B_{>k}) - \hat{H}}{3m\Delta_{[s]}^{-2}} \ge \frac{(m-k)\Delta_{[s]}^{-2} - 3k\Delta_{[s]}^{-2}}{3m\Delta_{[s]}^{-2}} \\
&\ge \frac{m-4k}{3m} \ge \frac{m - 4m/(26s)}{3m} \ge \frac{1}{6}.
\end{aligned}$$

Here the third step follows from $H(B_{>k}) \ge (m-k)\Delta_{[s]}^{-2}$ and $\hat{H} \le 3k\Delta_{[s]}^{-2}$. The last line holds since

$$k\Delta_{[s]}^{-2} \le \hat{H} \le H(B_{\le m})/(78s) \le m\Delta_{[s]}^{-2}/(26s),$$

which implies $k \le m/(26s)$. By Lemma B.11, we have

$$\Pr[X \ge 1/6] \le \delta^{m/6},$$

and thus the probability that Entropy-Elimination return an incorrect answer on $\hat{H}_t$ is bounded by $\delta^{2(s+k-1)} + \delta^{m/6}$.

It remains to apply a union bound on all valus of $\hat{H}_t$ that fall into this case. Since $k \ge 2$ and the values of k are distinct, the sum of the first term is bounded by

$$\sum_{k=2}^{\infty} \delta^{2(s+k-1)} = \frac{\delta^{2s+2}}{1-\delta^2} \le 2\delta^{2s+2}.$$

For the second term, note that the number of different values of $\hat{H}_t$ between $\Delta_{[s]}^{-2}$ and $\sum_{i=s}^{n}\Delta_{[i]}^{-2}/(78s) = H(B_{\le m})/(78s)$ is bounded by

$$\log_{100}\left[H(B_{\le m})/(78s)/\Delta_{[s]}^{-2}\right] + 1 \le \log_{100}[m/(26s)] + 1.$$

In particular, if $m < 26s$, no $\hat{H}_t$ will fall into this case. So in the following we focus on the nontrivial case that $m \ge 26s$. Since The sum of the second term is at most

$$\delta^{m/6}(\log_{100}[m/(26s)] + 1) \le \delta^{13s/3} \le \delta^{2s+2}.$$

Here the first step follows from the fact that $\delta^{m/6}(\log_{100}[m/(26s)] + 1)$ decreases on $[26s, +\infty)$ for all $\delta \in (0, 0.01)$ and $s \ge 3$. The second step follows from $s \le 3$.

Therefore, the total probability that Entropy-Elimination returns incorrectly in this sub-case is bounded by

$$2\delta^{2s+2} + \delta^{2s+2} \le 3\delta^{2s+2}.$$

Case 2-3: $\hat{H} \in [\sum_{i=s}^{n}\Delta_{[i]}^{-2}/(78s), 300000s\sum_{i=s}^{n}\Delta_{[i]}^{-2}]$.

In this case, we simply bound the probability of returning an incorrect answer by the probability that at least $s-2$ arms in $\{A_1, A_2, \ldots, A_{s-1}\}$ are mis-deleted, which is in turn bounded by $(s-1)\delta^{2(s-2)}$ according to Lemma B.9. As in the argument of Case 2-1, suppose that two arms A_i and A_j $(1 \le i < j \le s-1)$ are not mis-deleted. Let r^* be the group that contain A_{s-1}. Then both A_i and A_j are contained in S_{r^*}. However, as $4\varepsilon_{r^*}^{-2} = 4^{r^*+1} \ge \Delta_{[s-1]}^{-2} \ge \hat{H}_t$, Entropy-Elimination will reject in round r^*, which implies that Entropy-Elimination will never return a sub-optimal arm.

Note that at most

$$\log_{100}\frac{300000s}{1/(78s)} + 1 \le 2\log_{100}s + 5 = \log_{10}s + 5$$

different values of $\hat{H}$ fall into this case. Therefore, the total probability is bounded by

$$\delta^{2(s-2)}(\log_{10}s + 5)(s-1).$$

Combining Case 2-1 through Case 2-3 yields the following bound: the probability that Entropy-Elimination outputs an incorrect answer for some $3 \le s \le n$ and $\hat{H} \in [\Delta_{[s]}^{-2}, \Delta_{[s-1]}^{-2})$ is at most

$$\begin{aligned}
&\sum_{s=3}^{n}\left[\frac{1}{99}\delta^{2(s-2)}(s-2) + 3\delta^{2s+2} + \delta^{2(s-2)}(\log_{10}s+5)(s-1)\right]\\
=&\sum_{s=3}^{n}\delta^{2(s-2)}\left[\frac{s-2}{99} + 3\delta^{6} + (\log_{10}s+5)(s-1)\right]\\
\le&\sum_{s=3}^{\infty}\delta^{2(s-2)}(\log_{10}s+6)(s-1)\\
\le&\delta^2\sum_{s=3}^{\infty}0.01^{2(s-3)}(\log_{10}s+6)(s-1) \le 20\delta^2.
\end{aligned}$$

Case 3: $\hat{H}_t < \Delta_{[n]}^{-2}$.

Finally, we turn to the case that $\hat{H}$ is smaller than $\Delta_{[n]}^{-2}$. In this case, Complexity-Guessing always rejects. Suppose $A_n \in G_{r^*}$. Then in the first $r^* - 1$ rounds of Entropy-Elimination, Frac-Test always returns False. Thus no elimination is done before round r^*. Since $\hat{H}_t < \Delta_{[n]}^{-2} \le 4\varepsilon_{r^*}^{-2}$, Entropy-Elimination directly rejects when checking the if-statement at round r^*.

Case 1 through Case 3 together directly imply the lemma, as $3\delta^4 + 20\delta^2 < \delta/3$ for all $\delta \in (0, 0.01)$. ■

Appendix C. Analysis of Sample Complexity

Recall that $\mathcal{E}_1$ is the event that all calls of Frac-Test and Unif-Sampl in Entropy-Elimination return correctly. We bound the sample complexity of our algorithm using the following two lemmas.

Lemma C.1 *Conditioning on $\mathcal{E}_1$, the expected number of samples taken by Med-Elim and Elimination in Complexity-Guessing is*

$$O(H(\mathsf{Ent} + \ln \delta^{-1})).$$

Lemma C.2 *Conditioning on $\mathcal{E}_1$, the expected number of samples taken by Unif-Sampl and Frac-Test in Complexity-Guessing is*

$$O(\Delta_{[2]}^{-2} \ln\ln \Delta_{[2]}^{-1} \operatorname{polylog}(n, \delta^{-1})).$$

The two lemmas above directly imply the following theorem.

Theorem C.3 *Conditioning on $\mathcal{E}_1$, the expected sample complexity of Complexity-Guessing is*

$$O\left(H(\ln \delta^{-1} + \mathsf{Ent}) + \Delta_{[2]}^{-2} \ln\ln \Delta_{[2]}^{-1} \operatorname{polylog}(n, \delta^{-1})\right).$$

Theorems B.3 and C.3 together imply that Complexity-Guessing is a δ-correct algorithm for Best-1-Arm, and its expected sample complexity is

$$O\left(H(\ln \delta^{-1} + \mathsf{Ent}) + \Delta_{[2]}^{-2} \ln\ln \Delta_{[2]}^{-1} \operatorname{polylog}(n, \delta^{-1})\right)$$

conditioning on an event which happens with probability at least $1 - \delta$. However, to prove Theorem 1.11, we need a δ-correct algorithm with the desired sample complexity in expectation (not conditioning on another event). In the following, we prove Theorem 1.11 using a parallel simulation trick developed in Chen and Li (2015).

Proof [Proof of Theorem 1.11] Given an instance I of Best-1-Arm and a confidence level δ, we define a collection of algorithms $\{\mathbb{A}_k : k \in \mathbb{N}\}$, where $\mathbb{A}_k$ simulates Complexity-Guessing on instance I and confidence level $\delta_k = \delta/2^k$. Then we construct the following algorithm $\mathbb{A}$:

- $\mathbb{A}$ runs in iterations. In iteration t, for each k such that 2^{k-1} divides t, $\mathbb{A}$ simulates $\mathbb{A}_k$ until $\mathbb{A}_k$ requires a sample from some arm A. $\mathbb{A}$ draws a sample from A, feeds it to $\mathbb{A}_k$, and continue simulating $\mathbb{A}_k$ until it requires another sample. After that, $\mathbb{A}$ temporarily suspends $\mathbb{A}_k$.

- When some algorithm $\mathbb{A}_k$ terminates, $\mathbb{A}$ also terminates and returns the same answer.

We first note that if all algorithms in $\{\mathbb{A}_k\}$ are correct, $\mathbb{A}$ eventually returns the correct answer. Recall that $\mathbb{A}_k$ is a $\delta/2^k$-correct algorithm for Best-1-Arm. Thus by a simple union bound, $\mathbb{A}$ is correct with probability $1-\sum_{k=1}^{\infty}\delta/2^k = 1-\delta$, thus proving that $\mathbb{A}$ is δ-correct.

It remains to bound the sample complexity of $\mathbb{A}$. According to Theorem C.3, there exist constants C and m, along with a collection of events $\{\mathcal{E}_k\}$, such that for each k, $\Pr[\mathcal{E}_k] \ge 1-\delta_k$, and the expected number of samples taken by $\mathbb{A}_k$ conditioning on $\mathcal{E}_k$ is at most

$$\begin{aligned}
&C\left[H\cdot(\ln\delta_k^{-1}+\mathsf{Ent})+\Delta_{[2]}^{-2}\ln\ln\Delta_{[2]}^{-1}(\ln^m n+\ln^m\delta_k^{-1})\right]\\
\le& C\left[H\cdot(k\ln\delta+\mathsf{Ent})+\Delta_{[2]}^{-2}\ln\ln\Delta_{[2]}^{-1}(\ln^m n+(k\ln\delta^{-1})^m)\right]\\
\le& k^m\cdot T(I).
\end{aligned}$$

Here $T(I)$ denotes $C\left[H\cdot(\ln\delta^{-1}+\mathsf{Ent})+\Delta_{[2]}^{-2}\ln\ln\Delta_{[2]}^{-1}(\ln^m n+\ln^m\delta^{-1})\right]$, the desired sample complexity. The first step follows from the fact that $\ln\delta_k^{-1} = \ln\delta^{-1}+k \le k\ln\delta^{-1}$ for $\delta < 0.01$.

Since different algorithms in $\{\mathbb{A}_k\}$ take independent samples, the events $\{\mathcal{E}_k\}$ are independent. Define random variable σ as the minimum number such that event $\mathcal{E}_\sigma$ happens. Then it follows that

$$\Pr[\sigma=k]\le\Pr[\overline{\mathcal{E}}_1\cap\overline{\mathcal{E}}_2\cap\cdots\cap\overline{\mathcal{E}}_{k-1}]\le\prod_{i=1}^{k-1}\delta_i\le 0.01^{k-1}.$$

Let T_k denote the number of samples taken by $\mathbb{A}_k$ if it is allowed to run indefinitely (i.e., $\mathbb{A}$ does not terminate). Conditioning on $\sigma=k$, we have $\mathrm{E}[T_k]\le k^m\cdot T(I)$. Moreover, $\mathbb{A}$ terminates before or at iteration $2^{k-1}k^m\cdot T(I)$. It follows that the number of samples taken by $\mathbb{A}$ is bounded by

$$\sum_{i=1}^{\infty}\lfloor 2^{k-1}k^m\cdot T(I)/2^{i-1}\rfloor\le 2^{k-1}k^m\cdot T(I)\sum_{i=1}^{\infty}2^{-(i-1)}\le 2^k k^m\cdot T(I).$$

Thus the expected sample complexity of $\mathbb{A}$ is bounded by

$$\begin{aligned}
&\sum_{k=1}^{\infty}\Pr[\sigma=k]\cdot 2^k k^m\cdot T(I)\\
\le&\sum_{k=1}^{\infty}0.01^{k-1}\cdot 2^k k^m\cdot T(I)\\
\le&100T(I)\sum_{k=1}^{\infty}0.02^k k^m = O(T(I)).
\end{aligned}$$

Therefore, $\mathbb{A}$ is a δ-correct algorithm for Best-1-Arm with expected sample complexity of

$$O(H\cdot(\ln\delta^{-1}+\mathsf{Ent})+\Delta_{[2]}^{-2}\ln\ln\Delta_{[2]}^{-1}\operatorname{polylog}(n,\delta^{-1})).$$

■

We conclude the section with the proofs of Lemmas C.1 and C.2.

Proof [Proof of Lemma C.1] Suppose that $\textsf{Complexity-Guessing}$ terminates after iteration t_0. According to $\textsf{Entropy-Elimination}$, for each $1 \le t \le t_0$, the algorithm takes $O(\hat{H}_t) = O(100^t)$ samples in $\textsf{Med-Elim}$ and $\textsf{Elimination}$ when it runs on $\hat{H}_t$. As 100^t grows exponentially in t, it suffices to bound the expectation of the last term, namely 100^{t_0}.

Let $t^* = \lceil \log_{100} H(\mathsf{Ent} + \ln \delta^{-1}) + 3 \rceil$. We first show that when $t \ge t^*$, $\textsf{Entropy-Elimination}$ accepts $\hat{H}_t$ with constant probability. According to Remark B.7, the probability that $\textsf{Entropy-Elimination}$ rejects $\hat{H}_t$ is upper bounded by

$$\frac{256H}{\hat{H}_t} + \frac{H(\mathsf{Ent} + \ln \delta^{-1} + \ln(\hat{H}_t/H))}{100\hat{H}_t} \le \frac{256H}{100^3 H} + \frac{\hat{H}_t/20}{100\hat{H}_t} \le 1/200.$$

The first step follows from the following two observations. First, as $\hat{H}_t \ge \hat{H}_{t^*} \ge 100^3 H(\mathsf{Ent} + \ln \delta^{-1})$, we have $H(\mathsf{Ent} + \ln \delta^{-1}) \le 100^{-3}\hat{H}_t$. Second, since $\hat{H}_t/H \ge 100^3$ and $x \ge 100 \ln x$ holds for all $x \ge 10^6$, we have $H \ln(\hat{H}_t/H) \le H \cdot \frac{1}{100}(\hat{H}_t/H) = \hat{H}_t/100$.

Therefore, the probability that t_0 equals $t^* + k$ is bounded by 200^{-k} for all $k \ge 1$. It follows from a simple summation on all possible t_0 that

$$\begin{aligned}
\mathrm{E}\left[100^{t_0}\right] &= \sum_{t=1}^{\infty} 100^t \Pr[t_0 = t] \\
&\le \sum_{t=1}^{t^*} 100^t \cdot 1 + \sum_{k=1}^{\infty} 100^{t^*+k} \cdot 200^{-k} \\
&= O\left(100^{t^*}\right) = O\left(H(\mathsf{Ent} + \ln \delta^{-1})\right).
\end{aligned}$$

■

Proof [Proof of Lemma C.2] When $\textsf{Entropy-Elimination}$ runs on guess $\hat{H}_t$, $\textsf{Unif-Sampl}$ takes $O(\varepsilon_r^{-2} \ln \delta_r^{-1})$ samples in the r-th round, while the number of samples taken by $\textsf{Frac-Test}$ is

$$O\left(\varepsilon_r^{-2} \ln \delta_r^{-1} (\theta_r - \theta_{r-1})^{-2} \ln(\theta_r - \theta_{r-1})^{-1}\right) = O\left(\varepsilon_r^{-2} \ln \delta_r^{-1} (\theta_r - \theta_{r-1})^{-3}\right).$$

As the second term dominates the first, we focus on the complexity of $\textsf{Frac-Test}$ in the following analysis.

Recall that $\varepsilon_r = 2^{-r}$, $\delta_r = \delta/(50r^2t^2) \ge \delta^2/(r^2t^2)$ and $\theta_r - \theta_{r-1} = (ct - r)^{-2}/10$. For each t, suppose r ranges from 1 to $r_{\max}$, then the complexity at iteration t is bounded by

$$\begin{aligned}
&\sum_{r=1}^{r_{\max}} \varepsilon_r^{-2} \ln \delta_r^{-1} (\theta_r - \theta_{r-1})^{-3} \\
\le& 2 \sum_{r=1}^{r_{\max}} 4^r (\ln \delta^{-1} + \ln r + \ln t)[(ct - r)^{-2}/10]^{-3} \\
\le& 2000 \sum_{r=1}^{r_{\max}} 4^r (\ln \delta^{-1} + \ln r + \ln t)(ct - r)^6 \\
=& O(4^{r_{\max}} (\ln \delta^{-1} + \ln t)(ct - r_{\max})^6)
\end{aligned}$$

The last step follows from the observation that the last term dominates the summation, and the fact $\ln r_{\max} = O(\ln t)$ due to Observation A.1.

Let random variable t_0 denote the last t in the execution of $\textsf{Complexity-Guessing}$. As in the proof of Lemma C.1, we define $t^* = \lceil \log_{100} H(\textsf{Ent} + \ln \delta^{-1}) + 3 \rceil$. We have also shown that $\Pr[t \ge t_0 + k] \le 200^{-k}$ for all $k \ge 1$. Thus, the expected complexity incurred after iteration t^* can be bounded by the complexity at iteration t^*.

When $t < \log_{100} \Delta_{[2]}^{-2}$, it follows from $r_{\max} \le ct - 1$ that the complexity is

$$O(4^{ct}(\ln \delta^{-1} + \ln t)) = O(100^t(\ln \delta^{-1} + \ln t)).$$

Summing over $t = 1, 2, \ldots, \log_{100} \Delta_{[2]}^{-2}$ yields

$$\sum_{t=1}^{\log_{100} \Delta_{[2]}^{-2}} 100^t(\ln \delta^{-1} + \ln t) = O(\Delta_{[2]}^{-2}(\ln \delta^{-1} + \ln \ln \Delta_{[2]}^{-1})).$$

Clearly this term is bounded by the desired complexity.

When $\log_{100} \Delta_{[2]}^{-2} \le t \le t^*$, we choose $r_{\max} = \log_2 \Delta_{[2]}^{-1} = \log_4 \Delta_{[2]}^{-2}$. Note that in fact the algorithm may not always terminate before or at round $r_{\max}$. However, since the probability that the algorithm lasts $r_{\max} + k$ rounds is bounded by $O(100^{-k})$, the contribution of those rounds to total complexity is also dominated. Thus we have

$$\begin{aligned}
&\sum_{t=\log_{100} \Delta_{[2]}^{-2}}^{t_0} O(4^{r_{\max}}(\ln \delta^{-1} + \ln t)(ct - r_{\max})^6) \\
=&\sum_{t=\log_{100} \Delta_{[2]}^{-2}}^{t_0} O(\Delta_{[2]}^{-2}(\ln \delta^{-1} + \ln t)(ct - \log_4 \Delta_{[2]}^{-2})^6) \\
=&O\left(t^* - \log_{100} \Delta_{[2]}^{-2}\right) \cdot O\left(\Delta_{[2]}^{-2}(\ln \delta^{-1} + \ln t^*)(ct^* - \log_4 \Delta_{[2]}^{-2})^6\right) \\
=&O\left(\Delta_{[2]}^{-2}(\ln \delta^{-1} + \ln t^*)(ct^* - \log_4 \Delta_{[2]}^{-2})^7\right) \\
=&O\left(\Delta_{[2]}^{-2}(\ln \delta^{-1} + \ln \ln H)(\ln(H/\Delta_{[2]}^{-2}) + \ln \textsf{Ent} + \ln \delta^{-1})^7\right) \\
=&O\left(\Delta_{[2]}^{-2} \ln \ln \Delta_{[2]}^{-1}(\ln \delta^{-1} + \ln \ln n)(\ln n + \ln \delta^{-1})^7\right) \\
=&O\left(\Delta_{[2]}^{-2} \ln \ln \Delta_{[2]}^{-1} \operatorname{polylog}(n, \delta^{-1})\right).
\end{aligned}$$

The fourth step follows from

$$O(\ln t^*) = O(\ln \ln(H(\textsf{Ent} + \ln \delta^{-1}))) = O(\ln \ln H + \ln \ln \textsf{Ent} + \ln \ln \ln \delta^{-1}),$$

while the last two terms are dominated by $\ln \delta^{-1} + \ln \ln H$. The fifth step follows from the simple observation that $H/\Delta_{[2]}^{-2} \le n$ and $\textsf{Ent} = O(\ln \ln n)$. ■

Appendix D. Lower Bound

In this section, we prove Lemma 4.1. We restate it here for convenience.

Lemma 4.1. (restated) *Suppose $\delta \in (0, 0.04)$, $m \in \mathbb{N}$ and $\mathbb{A}$ is a δ-correct algorithm for SIGN-ξ. P is a probability distribution on $\{2^{-1}, 2^{-2}, \dots, 2^{-m}\}$ defined by $P(2^{-k}) = p_k$. $\mathsf{Ent}(P)$ denotes the Shannon entropy of distribution P. Let $T_{\mathbb{A}}(\mu)$ denote the expected number of samples taken by $\mathbb{A}$ when it runs on an arm with distribution $\mathcal{N}(\mu, 1)$ and $\xi = 0$. Define $\alpha_k = T_{\mathbb{A}}(2^{-k})/4^k$. Then,*

$$\sum_{k=1}^{m} p_k \alpha_k = \Omega(\mathsf{Ent}(P) + \ln \delta^{-1}).$$

D.1. Change of Distribution

We introduce a lemma that is essential for proving the lower bound for SIGN-ξ in Lemma 4.1, which is a special case of (Kaufmann et al., 2015, Lemma 1). In the following, KL stands for the Kullback-Leibler divergence, while $D(x||y) = x \ln(x/y) + (1-x) \ln[(1-x)/(1-y)]$ is the relative entropy function.

Lemma D.1 (Change of Distribution) *Let $\mathbb{A}$ be an algorithm for SIGN-ξ. Let A and A' be two instances of SIGN-ξ (i.e., two arms). $\Pr_A$ and $\Pr_{A'}$ (E_A and $\mathrm{E}_{A'}$) denote the probability law (expectation) when $\mathbb{A}$ runs on instance A and A' respectively. Random variable τ denotes the number of samples taken by the algorithm. For all event $\mathcal{E}$ in $\mathcal{F}_\sigma$, where σ is a stopping time with respect to the filtration $\{\mathcal{F}_t\}$, we have*

$$\mathrm{E}_A[\tau]\mathrm{KL}(A, A') \ge D\left(\Pr_A[\mathcal{E}] \middle|\middle| \Pr_{A'}[\mathcal{E}]\right).$$

D.2. Proof of Lemma 4.1

We start with an overview of our proof of Lemma 4.1. For each k, we consider the number of samples taken by Algorithm $\mathbb{A}$ when it runs on an arm with mean 2^{-k}. We first show that with high probability, this number is between $\Omega(4^k)$ and $O(4^k\alpha_k)$. Then we apply Lemma D.1 to show that the same event happens with probability at least $e^{-\alpha_k}$ when the input is an arm with mean zero.

Since the probability of an event is at most 1, we would like to bound the sum of $e^{-\alpha_k}$ by 1, yet the problem is that the events for different k may not be disjoint. To avoid this difficulty, we carefully select a collection of disjoint events denoted by S. We bound $\sum_{k=1}^{m} e^{-d\alpha_k}$ (for appropriate constant d) by $\sum_{k \in S} e^{-\alpha_k}$ based on the way we construct S. After that, we use the "change of distribution" argument (Lemma D.1) to bound $\sum_{k \in S} e^{-\alpha_k}$ by 1. As a result, we have the following inequality for appropriate constant M, which is reminiscent of Kraft's inequality in coding theory.

$$\sum_{k=1}^{m} e^{-d\alpha_k} \le M. \tag{10}$$

Once we obtain (10), the desired bound directly follows from a simple calculation.

Proof [Proof of Lemma 4.1] Fix $m \in \mathbb{N}$. Recall that all arms are normal distributions with a standard deviation of 1 and ξ is always equal to zero. $4^k\alpha_k$ is the expected number of samples taken by $\mathbb{A}$ on an arm A with distribution $\mathcal{N}(2^{-k}, 1)$. It is well-known that to distinguish $\mathcal{N}(2^{-k}, 1)$ from $\mathcal{N}(-2^{-k}, 1)$ with confidence level δ, $\Omega(4^k \ln \delta^{-1})$ samples are required in expectation. Therefore, we have $\alpha_k = \Omega(\ln \delta^{-1})$ for all k. It follows that $\sum_{k=1}^{m} p_k \alpha_k = \Omega(\ln \delta^{-1})$.

It remains to prove that $\sum_{k=1}^{m} p_k \alpha_k = \Omega(\mathsf{Ent}(P))$ for all 0.04-correct algorithms. For each $\mu \in \mathbb{R}$, let $\Pr_\mu$ and E_μ denote the probability and expectation when $\mathbb{A}$ runs on an arm with mean μ (i.e., $\mathcal{N}(\mu, 1)$). Define random variable $\tau_{\mathbb{A}}$ as the number of samples taken by $\mathbb{A}$. Let $c = 1/64$. Let $\mathcal{E}_k$ denote the event that $\mathbb{A}$ outputs "$\mu > 0$" and $\tau_{\mathbb{A}} \in [4^k c, 16 \cdot 4^k \alpha_k]$. The following lemma gives a lower bound of $\Pr_0[\mathcal{E}_k]$.

Lemma D.2

$$\Pr_0[\mathcal{E}_k] \ge \frac{1}{4} e^{-\alpha_k}.$$

Our second step is to choose a collection of disjoint events from $\{\mathcal{E}_k : 1 \le k \le m\}$. We have the following lemma.

Lemma D.3 *There exists a set $S \subseteq \{1, 2, \ldots, m\}$ such that:*

- *$\{\mathcal{E}_k : k \in S\}$ is a collection of disjoint events.*
- *$\sum_{k=1}^{m} e^{-d\alpha_k} \le M \sum_{k \in S} e^{-\alpha_k}$ for universal constants d and M independent of m and $\mathbb{A}$.*

It follows that

$$\sum_{k=1}^{m} e^{-d\alpha_k} \le M \sum_{k \in S} e^{-\alpha_k} \le 4M \sum_{k \in S} \Pr_0[\mathcal{E}_k] = 4M.$$

Here the first two steps follow from Lemma D.3 and Lemma D.2, respectively. The last step follows from the fact that $\{\mathcal{E}_k : k \in S\}$ is a disjoint collection of events.

Finally, for a distribution P on $\{2^{-1}, 2^{-2}, \ldots, 2^{-m}\}$ defined by $P(2^{-k}) = p_k$, we consider the following optimization problem with variables $\alpha_1, \alpha_2, \ldots, \alpha_m$:

$$\begin{aligned} \text{minimize} \quad & \sum_{k=1}^{m} p_k \alpha_k \\ \text{subject to} \quad & \sum_{k=1}^{m} e^{-d\alpha_k} \le 4M \end{aligned}$$

The method of Lagrange multipliers yields that the minimum value is obtained when $\sum_{k=1}^{m} e^{-d\alpha_k} = 4M$ and $e^{-d\alpha_k}$ is proportional to p_k. It follows that $\alpha_k = -\dfrac{1}{d} \ln(4Mp_k)$ and consequently

$$\sum_{k=1}^{m} p_k \alpha_k \ge \frac{1}{d} \sum_{k=1}^{m} p_k (\ln(4M)^{-1} + \ln p_k^{-1}) = \frac{1}{d} \left(\mathsf{Ent}(P) - \ln(4M)\right).$$

Note that d and M are constants independent of m, distribution P and algorithm $\mathbb{A}$. This completes the proof.

■

D.3. Proofs of Lemma D.2 and Lemma D.3

Finally, we prove the two technical lemmas.

Proof [Proof of Lemma D.2] Recall that our goal is to lower bound $\Pr_0[\mathcal{E}_k]$. We first show that $\Pr_{2^{-k}}[\mathcal{E}_k] \ge 1/2$ and then prove the desired lower bound by applying change of distribution. Recall that $\mathcal{E}_k = (\mathbb{A} \text{ outputs } \mu > 0) \wedge (\tau_{\mathbb{A}} \in [4^k c, 16 \cdot 4^k \alpha_k])$. We have

$$\begin{aligned}
\Pr_{2^{-k}}[\mathcal{E}_k] &\ge \Pr_{2^{-k}}[\mathbb{A} \text{ outputs } \mu > 0] - \Pr_{2^{-k}}\left[\tau_{\mathbb{A}} > 16 \cdot 4^k \alpha_k\right] - \Pr_{2^{-k}}\left[\tau_{\mathbb{A}} < 4^k c\right] \\
&\ge 1 - 0.04 - 1/16 - \Pr_{2^{-k}}\left[\tau_{\mathbb{A}} < 4^k c\right] \\
&\ge 0.8 - \Pr_{2^{-k}}\left[\tau_{\mathbb{A}} < 4^k c\right].
\end{aligned}$$

Here the second step follows from Markov's inequality and the fact that $\mathrm{E}_{2^{-k}}[\tau_{\mathbb{A}}] = 4^k \alpha_k$.

It remains to show that $\Pr_{2^{-k}}\left[\tau_{\mathbb{A}} < 4^k c\right] \le 0.3$. Suppose towards a contradiction this does not hold. Then we consider the algorithm $\mathbb{A}'$ that simulates $\mathbb{A}$ in the following way: if $\mathbb{A}$ terminates within $4^k c$ samples, $\mathbb{A}'$ outputs the same answer; otherwise $\mathbb{A}'$ outputs nothing. Let $\Pr_{\mathbb{A}',\mu}$ denote the probability when $\mathbb{A}'$ runs on an arm of mean μ. Moreover, let $\mathcal{E}_k^{\text{bad}}$ denote the event that the output is "$\mu > 0$". Then we have

$$\Pr_{\mathbb{A}',2^{-k}}[\mathcal{E}_k^{\text{bad}}] = \Pr_{2^{-k}}\left[\mathcal{E}_k^{\text{bad}} \wedge \tau_{\mathbb{A}} < 4^k c\right] \ge \Pr_{2^{-k}}\left[\tau_{\mathbb{A}} < 4^k c\right] - 0.04 > 0.26.$$

On the other hand, when we run $\mathbb{A}'$ on an arm with mean -2^{-k}, we have

$$\Pr_{\mathbb{A}',-2^{-k}}\left[\mathcal{E}_k^{\text{bad}}\right] \le \Pr_{-2^{-k}}\left[\mathcal{E}_k^{\text{bad}}\right] \le 0.04.$$

Since $\mathbb{A}'$ never takes more than $4^k c$ samples, it follows from Lemma D.1 that

$$\begin{aligned}
2c &= 4^k c \cdot \mathrm{KL}(\mathcal{N}(2^{-k}, 1), \mathcal{N}(-2^{-k}, 1)) \\
&\ge \mathrm{E}_{\mathbb{A}',2^{-k}}[\tau_{\mathbb{A}'}] \cdot \mathrm{KL}(\mathcal{N}(2^{-k}, 1), \mathcal{N}(-2^{-k}, 1)) \\
&\ge D\left(\Pr_{\mathbb{A}',2^{-k}}[\mathcal{E}_k^{\text{bad}}] \middle|\middle| \Pr_{\mathbb{A}',-2^{-k}}[\mathcal{E}_k^{\text{bad}}]\right) \\
&\ge D(0.26||0.04) \ge 0.2,
\end{aligned}$$

which leads to a contradiction as $c = 1/64$.

In the following, we lower bound $\Pr_0[\mathcal{E}_k]$ using change of distribution. Note that

$$\begin{aligned}
D\left(\Pr_{2^{-k}}[\mathcal{E}_k] \middle|\middle| \Pr_0[\mathcal{E}_k]\right) &\le 4^k \alpha_k \cdot \mathrm{KL}(\mathcal{N}(2^{-k}, 1), \mathcal{N}(0, 1)) \\
&\le 4^k \alpha_k \cdot \frac{1}{2}\left(2^{-k}\right)^2 = \alpha_k/2.
\end{aligned}$$

Let $\theta_k = e^{-\alpha_k}/4$. We have

$$D(1/2||\theta_k) = \frac{1}{2}\ln\frac{1}{4\theta_k(1-\theta_k)} \ge \frac{1}{2}\ln\frac{1}{4\theta_k} = \alpha_k/2.$$

Since we have shown $\Pr_{2^{-k}}[\mathcal{E}_k] \ge 1/2$, the two inequalities above imply

$$\Pr_0[\mathcal{E}_k] \ge \theta_k = \frac{1}{4}e^{-\alpha_k}.$$

■

Proof [Proof of Lemma D.3] We map each event $\mathcal{E}_k$ to an interval

$$\mathcal{I}_k = [\log_4(4^k c) + 3, \log_4(16 \cdot 4^k \alpha_k) + 3] = [k, k + \log_4 \alpha_k + 5].$$

By construction, two events $\mathcal{E}_i$ and $\mathcal{E}_j$ are disjoint if and only if their corresponding intervals, $\mathcal{I}_i$ and $\mathcal{I}_j$, are disjoint.

We construct a subset of $\{1, 2, \ldots, m\}$ using the following greedy algorithm:

- Sort $(1, 2, \ldots, m)$ into a list $(l_1, l_2, \ldots, l_m)$ such that $\alpha_{l_1} \le \alpha_{l_2} \le \cdots \le \alpha_{l_m}$.
- While the list is not empty, we add the first element x in the list into set S. Let $S_x = \{y : y \text{ is in the current list, and } \mathcal{I}_x \cap \mathcal{I}_y \ne \emptyset\}$. We remove all elements in S_x from the list.

Note that the way we construct S ensures that $\{\mathcal{E}_k : k \in S\}$ is indeed a disjoint collection of events, which proves the first part of the lemma. Moreover, $\{S_k : k \in S\}$ is a partition of $\{1, 2, \ldots, m\}$. Thus we have

$$\sum_{k=1}^{m} e^{-d\alpha_k} = \sum_{k \in S} \sum_{j \in S_k} e^{-d\alpha_j}. \tag{11}$$

It suffices to bound $\sum_{j \in S_k} e^{-d\alpha_j}$ by $Me^{-\alpha_k}$ for appropriate constants d and M. Summing over all k yields the desired bound

$$\sum_{k=1}^{m} e^{-d\alpha_k} \le \sum_{k \in S} e^{-\alpha_k}.$$

According to our construction of S, for all $j \in S_k$ we have $\alpha_j \ge \alpha_k$. For each integer $l \ge \lfloor \log_4 \alpha_k \rfloor$, we consider the values of j such that $\log_4 \alpha_j \in [l, l+1)$. Recall that the interval corresponding to event $\mathcal{E}_k$ is $\mathcal{I}_k = [k, k + \log_4 \alpha_k + 5]$. In order for $\mathcal{I}_j$ to intersect $\mathcal{I}_k$, we must have $j \in [k - \log_4 \alpha_j - 5, k + \log_4 \alpha_k + 5]$. Since $\log_4 \alpha_j < l + 1$, j must be contained in $[k - l - 6, k + \log_4 \alpha_k + 5]$, and thus there are at most $(\log_4 \alpha_k + l + 12)$ such values of j.

Recall that since $\mathcal{I}_k = [k, k + \log_4 \alpha_k + 5]$ is nonempty, we have $\alpha_k \ge 4^{-5}$. In the following calculation, we assume for simplicity that $\alpha_k \ge 1$ for all k, since it can be easily verified that the contribution of the terms with $\alpha_k < 1$ (i.e., $l = -5, -4, \ldots, -1$) is a constant, and thus can be covered by a sufficiently large constant M in the end. Then we have

$$\begin{aligned}
\sum_{j\in S_k} e^{-d\alpha_j} &\le \sum_{l=\lfloor \log_4 \alpha_k \rfloor}^{\infty} \sum_{j\in S_k} \exp(-d\alpha_j)\mathbb{I}[\log_4 \alpha_j \in [l, l+1)] \\
&\le \sum_{l=\lfloor \log_4 \alpha_k \rfloor}^{\infty} \exp(-d4^l)(\log_4 \alpha_k + l + 12) \\
&= (\log_4 \alpha_k + 12) \sum_{l=\lfloor \log_4 \alpha_k \rfloor}^{\infty} \exp\left(-d4^l\right) + \sum_{l=\lfloor \log_4 \alpha_k \rfloor}^{\infty} l \exp\left(-d4^l\right) \\
&= (\log_4 \alpha_k + 12) \cdot O(\exp(-d\alpha_k)) + O(\exp(-d\alpha_k) \cdot \log_4 \alpha_k) \\
&\le M(\log_4 \alpha_k + 12) \exp(-d\alpha_k) \\
&= M \exp(-d\alpha_k + \ln \log_4 \alpha_k + \ln 12) \le M e^{-\alpha_k}.
\end{aligned} \tag{12}$$

The first step rearranges the summation based on the value of l. The second step follows from the observation that S_k contains at most $\log_4 \alpha_k + l + 12$ values of j corresponding to each l. The fourth step holds since both summations decrease double-exponentially, and thus can be bounded by their respective first terms. Then we find a sufficiently large constant M (which depends on d) to cover the hidden constant in the big-O notation. Finally, the last step holds for sufficiently large d. In fact, we first choose d according to the last step, and then find the appropriate constant M. Clearly the choice of M and d is independent of the value of m and the algorithm $\mathbb{A}$. ■

Remark D.4 *Recall that all distributions are assumed to be Gaussian distributions with a fixed variance of* 1. *In fact, our proof of Lemma 4.1 only uses the following property: the KL-divergence between two distributions with mean μ_1 and μ_2 is $\Theta((\mu_1 - \mu_2)^2)$. Note that this property is indeed essential to the "change of distribution" argument in the proof of Lemma D.2.*

In general, suppose U is a set of real numbers and $\mathcal{D} = \{D_\mu : \mu \in U\}$ is a family of distributions with the following two properties: (1) the mean of distribution D_μ is μ; (2) $\mathrm{KL}(D_{\mu_1}, D_{\mu_2}) \le C(\mu_1 - \mu_2)^2$ for fixed constant $C > 0$. Then Lemma 4.1 also holds for distributions from $\mathcal{D}$.

For instance, suppose $\mathcal{D} = \{B(1, \mu) : \mu \in [1/2 - \varepsilon, 1/2 + \varepsilon]\}$, where $\varepsilon \in (0, 1/2)$ is a constant and $B(1, \mu)$ denotes the Bernoulli distribution with mean μ. Since

$$\mathrm{KL}(B(1,p), B(1,q)) \le \frac{(p-q)^2}{q(1-q)} \le \frac{(p-q)^2}{1/4 - \varepsilon^2},$$

distribution family D satisfies the condition above with $C = \dfrac{4}{1 - 4\varepsilon^2}$. It follows that Lemma 4.1 also holds for Bernoulli distributions with means sufficiently away from 0 *and* 1.

Appendix E. Missing Proofs in Section 5

In this section, we present the technical details in the proofs of Lemma 5.5 and Lemma 5.6. These are essentially identical to the proofs of Lemmas B.4 and B.5, which either use a potential function or apply a charging argument.

E.1. Proof of Lemma 5.5

Proof [Proof of Lemma 5.5 (continued)] Recall that $P(r, S_r)$ is defined as the probability that, given the value of S_r at the beginning of round r, at least one call to Elimination returns incorrectly at round r or later rounds, while Unif-Sampl and Frac-Test always return correctly. We prove inequality (2) by induction: for any S_r that contains the optimal arm A_1,

$$P(r, S_r) \le \frac{\delta}{\hat{H}}\left(128C(r, S_r) + 16M(r, S_r)\varepsilon_r^{-2}\right),$$

where

$$C(r, S_r) := \sum_{i=r-1}^{\infty} |S_r \cap G_i| \sum_{j=r}^{i+1} \varepsilon_j^{-2} + \sum_{i=r}^{r_{\max}+1} \varepsilon_i^{-2},$$

and

$$M(r, S_r) := |S_r \cap G_{\le r-2}|.$$

Note that if $|S_r| = 1$, the algorithm directly terminates at round r, and the inequality clearly holds. Thus, we assume $|S_r| \ge 2$ in the following.

Base case. We prove the base case $r = r_{\max} + 2$, where $r_{\max} = \max_{G_r \neq \emptyset} r$. Note that $C(r, S) = 0$ and $M(r, S) = |S| - 1$ for $r = r_{\max} + 2$ and any $S \subseteq I$ with $A_1 \in S$.

Let random variable r^* be the smallest integer greater than or equal to r, such that Med-Elim is correct at round r^*. Note that for $k \ge r$, $\Pr\left[r^* = k\right] \le 0.01^{k-r}$. We claim that conditioning on $r^* = k$, if Elimination is correct between round r and round k, the algorithm will terminate at round $k+1$. Consequently, the probability that Elimination fails in some round is bounded by the probability that it fails between round r and k. This allows us to upper bound the conditional probability by $\sum_{i=r}^{k} \delta'_i$.

Now we prove the claim. By Observation 5.8, the lower threshold used in Frac-Test at round k, denoted by c_k^{low}, is greater than or equal to $\mu_{[1]} - \varepsilon_k$. Since $k \ge r = r_{\max} + 2$,

$$|\{A \in S : \mu_A < c_k^{\text{low}}\}| \ge |\{A \in S : \mu_A < \mu_{[1]} - \varepsilon_k\}| = |S \cap G_{\le k-1}| \ge |S \cap G_{\le r_{\max}+1}| = |S| - 1 \ge 0.5|S|.$$

Thus by Fact 5.3, Frac-Test is guaranteed to return True in round k, and the algorithm calls Elimination. Then, by Observation 5.9, it holds that $d_k^{\text{low}} \ge \mu_{[1]} - 0.5\varepsilon_k$. Assuming that Elimination returns correctly at round k, the set returned by Elimination, denoted by S_{k+1}, satisfies $|\{A \in S_{k+1} : \mu_A < d_k^{\text{low}}\}| < 0.1|S_{k+1}|$, which implies

$$|S_{k+1}| - 1 \le |S_{k+1} \cap G_{\le k}| = |\{A \in S_{k+1} : \mu_A < \mu_{[1]} - 0.5\varepsilon_k\}| < 0.1|S_{k+1}|.$$

Thus we have $|S_{k+1}| = 1$, which proves the claim.

Summing over all possible k yields that the probability that Elimination returns incorrectly is upper bounded by

$$\begin{aligned}
\sum_{k=r}^{\infty} \Pr\left[r^* = k\right] \sum_{j=r}^{k} \delta'_j \le & \sum_{k=r}^{\infty} 0.01^{k-r} \sum_{j=r}^{k} \frac{|S_j|\varepsilon_j^{-2}}{\hat{H}}\delta \\
\le & \frac{4}{3} \cdot \frac{|S_r|\varepsilon_r^{-2}\delta}{\hat{H}} \sum_{k=r}^{\infty} 0.01^{k-r} \cdot 4^{k-r} \\
\le & \frac{2|S_r|\varepsilon_r^{-2}\delta}{\hat{H}} \le \frac{\delta}{\hat{H}}\left(128C(r, S_r) + 16M(r, S_r)\varepsilon_r^{-2}\right).
\end{aligned}$$

Inductive step. Assuming that the inequality holds for $r+1$ and all S_{r+1} that contains A_1, we bound the probability $P(r, S_r)$. We first note that both C and M are monotone in the following sense: $C(r,S) \le C(r,S')$ and $M(r,S) \le M(r,S')$ for $S \subseteq S'$. Moreover, we have

$$C(r,S_r) - C(r+1,S_r) = \sum_{i=r-1}^{\infty} |S_r \cap G_i|\varepsilon_r^{-2} + \varepsilon_r^{-2} = \varepsilon_r^{-2}(|S_r \cap G_{\ge r-1}| + 1). \tag{13}$$

We consider the following three cases separately:

- Case 1. Med-Elim is correct and Frac-Test returns True.
- Case 2. Med-Elim is correct and Frac-Test returns False.
- Case 3. Med-Elim is incorrect.

Let P_1 through P_3 denote the conditional probability of the event that Elimination fails at some round while Unif-Sampl and Frac-Test are correct in Case 1 through Case 3.

Upper bound P_1. Assuming that Frac-Test returns True, procedure Elimination will be called at round r. By a union bound, we have $P_1 \le P(r+1, S_{r+1}) + \delta_r'$, where S_{r+1} is the set of arms returned by Elimination. According to the inductive hypothesis, the monotonicity of C, and identity (13),

$$\begin{aligned} P(r+1,S_{r+1}) &\le \frac{\delta}{\hat{H}}\left(128C(r+1,S_{r+1}) + 16M(r+1,S_{r+1})\varepsilon_{r+1}^{-2}\right) \\ &\le \frac{\delta}{\hat{H}}\left(128C(r+1,S_r) + 64M(r+1,S_{r+1})\varepsilon_r^{-2}\right) \\ &= \frac{\delta}{\hat{H}}\left[128C(r,S_r) + \varepsilon_r^{-2}(64M(r+1,S_{r+1}) - 128|S_r \cap G_{\ge r-1}| - 128)\right]. \end{aligned} \tag{14}$$

By Observation 5.9, $d_r^{\mathrm{low}} \le \mu_{[1]} - \varepsilon_r$. If Elimination returns correctly at round r, we have

$$M(r+1,S_{r+1}) = |\{A \in S_{r+1} : \mu_A < \mu_{[1]} - \varepsilon_r\}| \le |\{A \in S_{r+1} : \mu_A < d_r^{\mathrm{low}}\}| < 0.1|S_{r+1}| \le 0.1|S_r|.$$

For brevity, let N_{sma}, N_{cur} and N_{big} denote $|S_r \cap G_{\le r-2}|$, $|S_r \cap G_{r-1}|$ and $|S_r \cap G_{\ge r}|$, respectively. Note that $|S_r| = N_{\mathsf{sma}} + N_{\mathsf{cur}} + N_{\mathsf{sma}} + 1$. Then we have

$$\begin{aligned} P_1 &\le P(r+1,S_{r+1}) + \delta_r' \\ &\le \frac{\delta}{\hat{H}}\left[128C(r,S_r) + \varepsilon_r^{-2}(|S_r| + 64M(r+1,S_{r+1}) - 128|S_r \cap G_{\ge r-1}| - 128)\right] \\ &\le \frac{\delta}{\hat{H}}\left[128C(r,S_r) + \varepsilon_r^{-2}(7.4(N_{\mathsf{sma}} + N_{\mathsf{cur}} + N_{\mathsf{big}} + 1) - 128(N_{\mathsf{cur}} + N_{\mathsf{big}} + 1))\right] \\ &\le \frac{\delta}{\hat{H}}\left[128C(r,S_r) + 7.4\varepsilon_r^{-2}N_{\mathsf{sma}}\right]. \end{aligned}$$

Upper bound P_2. Since Frac-Test returns True, procedure Elimination is not called. Then $P_2 \le P(r+1, S_{r+1}) = P(r+1, S_r)$. By inequality (14),

$$\begin{aligned} P_2 &\le \frac{\delta}{\hat{H}} \left[128C(r, S_r) + \varepsilon_r^{-2}(64M(r+1, S_r) - 128|S_r \cap G_{\ge r-1}| - 128)\right] \\ &\le \frac{\delta}{\hat{H}} \left[128C(r, S_r) + \varepsilon_r^{-2}(64(N_{\mathsf{sma}} + N_{\mathsf{cur}}) - 128(N_{\mathsf{cur}} + N_{\mathsf{big}} + 1))\right] \\ &\le \frac{\delta}{\hat{H}} \left[128C(r, S_r) + 64\varepsilon_r^{-2}(N_{\mathsf{sma}} - N_{\mathsf{cur}} - N_{\mathsf{big}} - 1)\right] \le \frac{\delta}{\hat{H}} \cdot 128C(r, S_r). \end{aligned}$$

Here the last step holds since by Observation 5.8, $c_r^{\text{low}} \ge \mu_{[1]} - 2\varepsilon_r$, and thus Frac-Test returns False implies that

$$N_{\mathsf{sma}} = |S_r \cap G_{\le r-2}| = |\{A \in S_r : \mu_A < \varepsilon_{r-1}\}| < 0.5|S_r| = (N_{\mathsf{sma}} + N_{\mathsf{cur}} + N_{\mathsf{big}} + 1)/2.$$

Upper bound P_3. By (14) and $M(r+1, S_{r+1}) \le M(r+1, S_r)$, we have

$$\begin{aligned} P_3 &\le P(r+1, S_{r+1}) + \delta_r' \\ &\le \frac{\delta}{\hat{H}} \left[128C(r, S_r) + \varepsilon_r^{-2}(64M(r+1, S_r) - 128|S_r \cap G_{\ge r-1}| - 128 + |S_r|)\right] \\ &= \frac{\delta}{\hat{H}} \left[128C(r, S_r) + \varepsilon_r^{-2}(64(N_{\mathsf{sma}} + N_{\mathsf{cur}}) - 128(N_{\mathsf{cur}} + N_{\mathsf{big}}) - 128 + (N_{\mathsf{sma}} + N_{\mathsf{cur}} + N_{\mathsf{big}} + 1))\right] \\ &\le \frac{\delta}{\hat{H}} \left[128C(r, S_r) + 65\varepsilon_r^{-2} N_{\mathsf{sma}}\right]. \end{aligned}$$

Recall that Case 3 happens with probability at most 0.01, and $N_{\mathsf{sma}} = |S_r \cap G_{\le r-2}| = M(r, S_r)$. Therefore, we obtain the following bound on $P(r, S_r)$, which finishes the proof.

$$\begin{aligned} P(r, S_r) &\le 0.01 \cdot \frac{\delta}{\hat{H}} \left[128C(r, S_r) + 65\varepsilon_r^{-2} N_{\mathsf{sma}}\right] + 0.99 \cdot \frac{\delta}{\hat{H}} \left[128C(r, S_r) + 7.4\varepsilon_r^{-2} N_{\mathsf{sma}}\right] \\ &\le \frac{\delta}{\hat{H}} \left(128C(r, S_r) + 16\varepsilon_r^{-2} N_{\mathsf{sma}}\right) \\ &\le \frac{\delta}{\hat{H}} \left(128C(r, S_r) + 16M(r, S_r)\varepsilon_r^{-2}\right). \end{aligned}$$

■

E.2. Proof of Lemma 5.6

Proof [Proof of Lemma 5.6 (continued)] Recall that for each round i, r_i is defined as the largest integer r such that $|G_{\ge r}| \ge 0.5|S_i|$, and

$$T_{i,j} = \begin{cases} 0, & j < r_i, \\ \varepsilon_i^{-2}\left(\ln \delta^{-1} + \ln \dfrac{H}{|G_j|\varepsilon_i^{-2}}\right), & j \ge r_i \end{cases}$$

is the number of samples that each arm in G_j is charged at round i.

We first show that $\sum_j |G_j|T_{i,j}$ is an upper bound on $|S_i|\varepsilon_i^{-2}\left(\ln\delta^{-1}+\ln\frac{H}{|S_i|\varepsilon_i^{-2}}\right)$, the number of samples taken by Med-Elim and Elimination at round i. Recall that $|G_{\ge r_i}| \ge 0.5|S_i|$. By definition of $T_{i,j}$,

$$\begin{aligned}\sum_j |G_j|T_{i,j} &= \sum_{j\ge r_i}|G_j|\varepsilon_i^{-2}\left(\ln\delta^{-1}+\ln\frac{H}{|G_j|\varepsilon_i^{-2}}\right)\\ &\ge |G_{\ge r_i}|\varepsilon_i^{-2}\left(\ln\delta^{-1}+\ln\frac{H}{|S_i|\varepsilon_i^{-2}}\right)\\ &\ge \frac{1}{2}|S_i|\varepsilon_i^{-2}\left(\ln\delta^{-1}+\ln\frac{H}{|S_i|\varepsilon_i^{-2}}\right).\end{aligned}$$

Then we prove the upper bound on $\sum_i \mathrm{E}[T_{i,j}]$, the expected number of samples that each arm in G_j is charged. For $i \le j+1$, we have the straightforward bound

$$\mathrm{E}[T_{i,j}] \le \varepsilon_i^{-2}\left(\ln\delta^{-1}+\ln\frac{H}{|G_j|\varepsilon_i^{-2}}\right). \tag{15}$$

For $i \ge j+2$, we note that $T_{i,j}$ is non-zero only if $j \ge r_i$, which implies that $|G_{\ge j+1}| < 0.5|S_i|$. We claim that this happens only if Med-Elim fails between round $j+2$ and round $i-1$, which happens with probability at most 0.01^{i-j-1}. In fact, suppose Med-Elim is correct at some round k, where $j+2 \le k \le i-1$. By Observations 5.8 and 5.9, $c_k^{\mathrm{low}} \ge \mu_{[1]} - 2\varepsilon_k$ and $d_k^{\mathrm{low}} \ge \mu_{[1]} - \varepsilon_k$, where c^{low} and d^{low} are the two lower thresholds used in Frac-Test and Elimination. If Frac-Test returns False, by Fact 5.3, we have

$$|S_k \cap G_{<k-1}| = \{A \in S_k : \mu_A < \mu_{[1]} - 2\varepsilon_k\} \le \{A \in S_k : \mu_A < c_k^{\mathrm{low}}\} < 0.5|S_k|.$$

Since $S_{k+1} = S_k$ in this case, it follows that $|S_{k+1} \cap G_{<k-1}| < 0.5|S_{k+1}|$. If Frac-Test returns True and the algorithm calls Elimination, by Fact 5.4,

$$|S_{k+1} \cap G_{<k}| = |\{A \in S_{k+1} : \mu_A < \mu_{[1]} - \varepsilon_k\}| \le |\{A \in S_{k+1} : \mu_A < d_k^{\mathrm{low}}\}| < 0.1|S_{k+1}|.$$

In either case, we have $|S_{k+1} \cap G_{\ge k-1}| > 0.5|S_{k+1}|$, and thus,

$$|G_{\ge j+1}| \ge |G_{\ge k-1}| \ge |S_{k+1} \cap G_{\ge k-1}| > 0.5|S_{k+1}| \ge 0.5|S_i|,$$

which contradicts $|G_{\ge j+1}| < 0.5|S_i|$. Therefore, for $i \ge j+2$, we have

$$\begin{aligned}\mathrm{E}[T_{i,j}] &= \Pr\left[T_{i,j} > 0\right] \cdot \varepsilon_i^{-2}\left(\ln\delta^{-1}+\ln\frac{H}{|G_j|\varepsilon_i^{-2}}\right)\\ &\le 0.01^{i-j-1} \cdot \varepsilon_i^{-2}\left(\ln\delta^{-1}+\ln\frac{H}{|G_j|\varepsilon_i^{-2}}\right).\end{aligned} \tag{16}$$

By (15) and (16), a direct summation gives

$$\sum_i \mathrm{E}[T_{i,j}] = O\left(\varepsilon_j^{-2}\left(\ln\delta^{-1}+\ln\frac{H}{|G_j|\varepsilon_j^{-2}}\right)\right).$$

■