Exploring k out of Top ρ Fraction of Arms in Stochastic Bandits

Wenbo Ren[1], Jia Liu[2], Ness Shroff[1]
1: Department of Computer Science, Ohio State University, Columbus, OH 43210
2: Department of Computer Science, Iowa State University, Ames, Iowa 50011
ren.453@osu.edu, jialiu@iastate.edu, shroff.11@osu.edu

October 30, 2018

Abstract

This paper studies the problem of identifying any k distinct arms among the top ρ fraction (e.g., top 5%) of arms from a finite or infinite set with a probably approximately correct (PAC) tolerance ϵ. We consider two cases: (i) when the threshold of the top arms' expected rewards is known and (ii) when it is unknown. We prove lower bounds for the four variants (finite or infinite, and threshold known or unknown), and propose algorithms for each. Two of these algorithms are shown to be sample complexity optimal (up to constant factors) and the other two are optimal up to a log factor. Results in this paper provide up to $\rho n/k$ reductions compared with the "k-exploration" algorithms that focus on finding the (PAC) best k arms out of n arms. We also numerically show improvements over the state-of-the-art.

1 INTRODUCTION

Background. Multi-armed bandit (MAB) problems [9] have been studied for decades, and well abstract the problems of decision making with uncertainty. It has been widely applied to many areas, e.g., online advertising [24], clinical trials [8], adaptive routing [10], and pairwise ranking [1]. In this paper, we focus on stochastic multi-armed bandit. In this setting, each arm of the bandit is assumed to follow a distribution. Whenever the decision maker samples this arm, an independent instance of this distribution is returned. The decision maker adaptively chooses some arms to sample in order to achieve some specific goals. So far, the majority of works in this area has been focused on minimizing the *regret* (deviation from optimum), (e.g., [5; 10; 2; 6; 17]) i.e., how to trade-off between the exploration and exploitation of arms to minimize the regret.

In this paper, instead of regret minimization, we focus on pure exploration problems, which aim either (i) to identify one or multiple arms satisfying specific conditions (e.g., with the highest expected rewards) and try to minimize the number of samples taken (e.g., [25; 20; 21; 12; 1; 22; 18; 14; 7]), or (ii) to identify one or multiple best possible arms according to a given criteria within a fixed number of samples (e.g., [4; 13; 11]). In some applications such as product testing [23; 4; 26], before the products are launched, rewards are insignificant, and it is more interesting to explore the best products with the least cost, which also suggests the pure exploration setting. This paper focuses on (i) above.

We investigate the problem of identifying any k arms that are in the top ρ fraction of the expected rewards of the arm set. This is in contrast to most works in the pure exploration space that have focused on the problem of identifying k best arms of a given arm set. We name the former as the "quantile exploration" (QE) problem, and the latter as the "k-exploration" (KE) problem. The motivations of studying the QE problem are as follows: First, in many applications, it is not necessary to identify the best arms, since it is acceptable to find "good enough" arms. For instance, a company wants to hire 100 employees from more than ten thousand applicants. It may be costly to find the best 100 applicants, and may be good enough to identify 100 within a certain top percentage (e.g., 5%); Second, theoretical analysis [21; 25] shows that the lower bound on the sample complexity (aka, number of samples taken) of the KE problem depends on n. When the number of arms is extremely large or possibly infinite, it is not feasible to find the best arms, but may be feasible to find arms within a certain top quantile; Third, by adopting the QE setting, we replace the sample complexity's dependence on n of the KE problem with k/ρ [14], which can be much smaller, and can greatly reduce the number of samples needed to find "good" arms.

This paper adopts the probably approximately correct (PAC) setting, where an ϵ bounded error is tolerated. This setting can avoid the cases where arms are too close—making the number of samples needed extremely large. The PAC setting has been adopted by numerous previous works [25; 21; 20; 12; 18; 14; 7; 22].

Model and Notations: Let $\mathcal{S}$ be the set of arms. It can be finite or infinite. When $\mathcal{S}$ is finite, let n be its size, and the top ρ fraction arms are simply the top $\lfloor \rho n \rfloor$ arms. If $\mathcal{S}$ is infinite, we assume that the arms' expected rewards follow some unknown prior identified by an unknown cumulative distribution function (CDF) $\mathcal{F}$. $\mathcal{F}$ is not necessarily continuous. In this paper, we assume the rewards of the arms are of the same finite support, and normalize them into $[0,1]$. For an arm a, we use R_a^t to denote the reward of its t-th sample. $(R_a^t, t \in \mathbb{Z}^+)$ are identical and independent. We also assume that the samples are independent across time and arms. For any arm a, let μ_a be its expected reward, i.e. $\mu_a := \mathbb{E}R_a^1$. To formulate the problem, for any $\rho \in (0,1)$, we define the inverse of $\mathcal{F}$ as

$$\mathcal{F}^{-1}(p) := \sup\{x : \mathcal{F}(x) \le p\}. \tag{1}$$

The inverse $\mathcal{F}^{-1}$ has the following two properties (2) and (3), where $X \sim \mathcal{F}$ means that X is a random variable following the distribution defined by $\mathcal{F}$.

$$\mathcal{F}(\mathcal{F}^{-1}(p)) \ge p, \tag{2}$$

$$\mathbb{P}_{X\sim\mathcal{F}}\{X \ge \mathcal{F}^{-1}(p)\} \ge 1-p. \tag{3}$$

To see (2), by contradiction, suppose $\mathcal{F}(\mathcal{F}^{-1}(p)) < p$. Since $\mathcal{F}(x)$ is right continuous, there exists a number x_1 such that $x_1 > \mathcal{F}^{-1}(p)$ and $\mathcal{F}(x_1) < p$. This implies that x_1 is in $\{x : \mathcal{F}(x) \le p\}$, and thus contradicting (1). Define $\mathcal{G}(x) := \mathbb{P}_{X\sim\mathcal{F}}\{X \ge x\}$. Similar to (2), the left continuity of $\mathcal{G}$ implies (3).

In the finite-armed case, an arm a is said to be (ϵ, m)-optimal if $\mu_a + \epsilon \ge \lambda_{[m]}$, where $\lambda_{[m]}$ is defined as the m-th largest expected reward among all arms in $\mathcal{S}$. In other words, the expected reward of an (ϵ, m)-optimal arm plus ϵ is no less than $\lambda_{[m]}$. The QE problem is to find k distinct (ϵ, m)-optimal arms of $\mathcal{S}$. We consider both cases where $\lambda_{[n]}$ is known and unknown.

Given a set $\mathcal{S}$ of size n, $k \in \mathbb{Z}^+$ and $\epsilon, \delta \in (0, \frac{1}{2})$, we define the two finite-armed QE problems Q-FK (Quantile, Finite-armed, $\lambda_{[m]}$ Known) and Q-FU (Quantile, Finite-armed, $\lambda_{[m]}$ Unknown) as follows:

Problem 1 (**Q-FK**). *With known $\lambda_{[m]}$, we want to find k distinct (ϵ, m)-optimal arms with at most δ error probability, and use as few samples as possible.*

Problem 2 (**Q-FU**). *Without knowing $\lambda_{[m]}$, we want to find k distinct (ϵ, m)-optimal arms with at most δ error probability, and use as few samples as possible.*

In the infinite-armed case, an arm is said to be $[\epsilon, \rho]$-optimal if its expected reward is no less than $\mathcal{F}^{-1}(1-\rho) - \epsilon$. Here we use brackets to avoid ambiguity. To simplify notation, we define $\lambda_\rho := \mathcal{F}^{-1}(1-\rho)$. An $[\epsilon, \rho]$-optimal arm is within the top ρ fraction of $\mathcal{S}$ with an at most ϵ error. We consider both cases where λ_ρ is known and unknown. Note that in both cases, we have no knowledge on $\mathcal{F}$ except that λ_ρ is possibly known.

Given a set $\mathcal{S}$ of infinite number of arms, $k \in \mathbb{Z}^+$, and $\rho, \delta, \epsilon \in (0, 1/2)$, we define the two infinite-armed QE problems Q-IK (Quantile, Infinite-armed, λ_ρ Known) and Q-IU (Quantile, Infinite-armed, λ_ρ Unknown).

Problem 3 (**Q-IK**). *Knowing λ_ρ, we want to find k distinct $[\epsilon, \rho]$-optimal arms with error probability no more than δ, and use as few samples as possible.*

Problem 4 (**Q-IU**). *Without knowing λ_ρ, we want to find k distinct $[\epsilon, \rho]$-optimal arms with error probability no more than δ, and use as few samples as possible.*

2 RELATED WORKS

To our best knowledge, Goschin et al. [18] were the first ones who has focused on the QE problems. They derived the tight lower bound $\Omega(\frac{1}{\epsilon^2}(\frac{1}{\rho} + \log\frac{1}{\delta}))$[1] for the Q-IK problem with $k = 1$. They also provided an Q-IK algorithm for $k = 1$, with sample complexity $O(\frac{1}{\rho\epsilon^2}\log\frac{1}{\delta})$, higher than the lower bound roughly by a $\log\frac{1}{\delta}$ factor. In contrast, our Q-IK algorithm works for all k values and matches the lower bound.

Chaudhuri and Kalyanakrishnan [14] studied the Q-IU and Q-FU problems with $k = 1$. They derived the lower bounds for $k = 1$. In this paper, we generalize their lower bounds to cases with $k > 1$. They also proposed algorithms for these two problems with $k = 1$, and the upper bounds ($O(\frac{1}{\rho\epsilon^2}\log^2\frac{1}{\delta})$ for Q-IK, $O(\frac{n}{m\epsilon^2}\log^2\frac{1}{\delta})$ for Q-FK) are the same as ours. For the case $k > 1$, by simply repeating their algorithms k times and setting error probability to $\frac{\delta}{k}$ for each repetition, one can solve the two problems with sample complexity $O(\frac{k}{\rho\epsilon^2}\log^2\frac{k}{\delta})$ and $O(\frac{n}{mk\epsilon^2}\log^2\frac{k}{\delta})$, respectively. In this paper, we propose new algorithms for all k values with $\log\frac{k}{\delta}$ reductions over sample complexity.

Aziz et al. [7] studied the Q-IU problem. They proposed a Q-IK algorithm which is higher than the lower bound proved in this paper by a $\log\frac{1}{\rho\delta}$ factor in the worst case. Under some "good" priors, its theoretical sample complexity can be lower than ours. However, numerical results in this paper show that our algorithm still obtains improvement under "good" priors.

Although the KE problem is not the focus of this paper, we provide a quick overview for comparative perspective. An early attempt on the KE problem was done by Even-Dar et al. [16], which proposed an algorithm called Median-Elimination that finds an $(\epsilon, 1)$-optimal arm with probability $1-\delta$ by taking

[1] All log, unless explicitly noted, are natural log.

Table 1: Comparison of Previous Works and Ours. All Bounds Are for the Worst Instances.

PROBLEM	WORK	SAMPLE COMPLEXITY
Q-IK	Goschin et al. [18]	$O\left(\frac{1}{\rho\epsilon^2}\log\frac{1}{\delta}\right)$ for $k=1$ $\Omega\left(\frac{1}{\epsilon^2}\left(\frac{1}{\rho}+\log\frac{1}{\delta}\right)\right)$ for $k=1$
	This Paper	$\Theta\left(\frac{k}{\epsilon^2}\left(\frac{1}{\rho}+\log\frac{k}{\delta}\right)\right)$ for $k\in\mathbb{Z}^+$
Q-FK	Goschin et al. [18]	$O\left(\frac{m}{n\epsilon^2}\log\frac{1}{\delta}\right)$ for $k=1$
	This Paper	$O\left(\frac{1}{\epsilon^2}\left(n\log\frac{m+1}{m+1-k}+k\log\frac{k}{\delta}\right)\right)$ for $k\le m\le n/2$ $\Omega\left(\frac{k}{\epsilon^2}\left(\frac{n}{m}+\log\frac{k}{\delta}\right)\right)$ for $k\le m\le n/2$
Q-IU	Chaudhuri et al. [14] and Aziz et al. [7]	$O\left(\frac{1}{\rho\epsilon^2}\log^2\frac{1}{\delta}\right)$ for $k=1$ $\Omega\left(\frac{1}{\rho\epsilon^2}\log\frac{1}{\delta}\right)$ for $k=1$
	This Paper	$O\left(\frac{1}{\epsilon^2}\left(\frac{1}{\rho}\log^2\frac{1}{\delta}+k\left(\frac{1}{\rho}+\log\frac{k}{\delta}\right)\right)\right)$ for $k\in\mathbb{Z}^+$ $\Omega\left(\frac{1}{\epsilon^2}\left(\frac{1}{\rho}\log\frac{1}{\delta}+k\left(\frac{1}{\rho}+\log\frac{k}{\delta}\right)\right)\right)$ for $k\in\mathbb{Z}^+$
Q-FU	Chaudhuri et al. [14]	$O\left(\frac{n}{m\epsilon^2}\log^2\frac{1}{\delta}\right)$ for $k=1$ $\Omega\left(\frac{n}{m\epsilon^2}\log\frac{1}{\delta}\right)$ for $k=1$
	Aziz et al. [7]	$O\left(\frac{n}{m\epsilon^2}\log^2\frac{1}{\delta}\right)$ for $k=1$
	This Paper	$O\left(\frac{1}{\epsilon^2}\left(\frac{n}{m}\log^2\frac{1}{\delta}+n\log\frac{m+2}{m+2-2k}+k\log\frac{k}{\delta}\right)\right)$ for $2k<m\le n/2$ $\Omega\left(\frac{1}{\epsilon^2}\left(\frac{n}{m}\log\frac{1}{\delta}+k\left(\frac{n}{m}+\log\frac{k}{\delta}\right)\right)\right)$ for $k\le m\le n/2$

at most $O(\frac{n}{\epsilon^2}\log\frac{1}{\delta})$ samples. Mannor and Tsitsiklis [25]; Kalyanakrishnan et al. [21]; Kalyanakrishnan and Stone [20]; Agarwal et al. [1]; Cao et al. [12]; Jamieson et al. [19]; Chen et al. [15]; Kaufmann and Kalyanakrishnan [22] studied the KE problem in different settings. Halving algorithm proposed by Kalyanakrishnan and Stone [20] finds k distinct (ϵ, k)-optimal arms with probability $1-\delta$ by using $O(\frac{n}{\epsilon^2}\log\frac{k}{\delta})$ samples. Kalyanakrishnan and Stone [20]; Kalyanakrishnan et al. [21]; Jamieson et al. [19]; Chaudhuri and Kalyanakrishnan [14]; Aziz et al. [7]; Kaufmann and Kalyanakrishnan [22] used confidence bounds to establish algorithms that can exploit the large gaps between the arms. In practice, these algorithms are promising in most situations, while in the worst case, their sample complexities can be higher than the lower bound by log factors.

3 LOWER BOUND ANALYSIS

We first establish the lower bound for the Q-FK problem. The lower bound is stated in Theorem 1.

Theorem 1 (Lower bound for Q-FK)**.** *Given $k\le m\le n/2$, $\epsilon\in(0,\frac{1}{4})$, and $\delta\in(0,e^{-8}/40)$, there is a set such that to find k distinct (ϵ, m)-optimal arms of it with error probability at most δ, any algorithm must take $\Omega(\frac{k}{\epsilon^2}(\frac{n}{m}+\log\frac{k}{\delta}))$ samples in expectation.*

Proof Sketch. Theorem 13 of [25] shows that there is an $\frac{n}{m}$-sized set such that to find an $(\epsilon, 1)$-optimal arm, any algorithm needs to take $\Omega(\frac{1}{\epsilon^2}(\frac{n}{m}+\log\frac{1}{\delta}))$ samples in expectation. We will show that any algorithm that solves the Q-FK problem with $k=1$ can be transformed to solve the above problem, and the lower bound for $k=1$ follows. Then, we construct k problems, each of which requires to find an $(\epsilon, \frac{m}{k})$-optimal arm from an $\frac{n}{k}$-sized set that matches the lower bound proved above. We will show that to solve these k problems with total error probability no more than δ, any algorithm needs $\Omega(\frac{k}{\epsilon^2}(\frac{n}{m}+\log\frac{k}{\delta}))$ samples in expectation. Any algorithm that solves the Q-FK problem with parameter k can be transformed to solve the above k problems. The desired lower bound follows. □

By Theorem 1, we prove Theorem 2, the lower bound for the Q-IK problem.

Theorem 2 (Lower bound for Q-IK)**.** *Given k, $\rho\in(0,\frac{1}{2}]$, $\epsilon\in(0,\frac{1}{4})$, and $\delta\in(0,e^{-8}/40)$, there is an infinite set such that to find k distinct $[\epsilon,\rho]$-optimal arms of it with error probability at most δ, any algorithm must take $\Omega(\frac{k}{\epsilon^2}(\frac{1}{\rho}+\log\frac{k}{\delta}))$ samples in expectation.*

Proof. By contradiction, suppose there is an algorithm $\mathcal{A}$ that solves all instances of the Q-IK problem by using $o(\frac{k}{\epsilon^2}(\frac{1}{\rho}+\log\frac{k}{\delta}))$ samples in expectation. Choosing $m\ge\frac{k(k-1)}{\delta}$ and $n\ge 2m$, we construct an n-sized set $\mathcal{C}$ that meets the lower bound of the Q-FK problem. By drawing arms from $\mathcal{C}$ with replacement, we can apply $\mathcal{A}$ to it with $\rho=\frac{m}{n}$. Now, we use $\mathcal{A}$ to find k possibly duplicated (ϵ, m)-optimal arms of $\mathcal{C}$ with error probability $\delta/2$. The probability that there is no duplication in these k found arms is at least $\prod_{i=1}^{k} \frac{m+1-i}{m} \geq 1-\sum_{i=1}^{k} \frac{i-1}{m} \geq 1-\frac{\delta}{2}$. Thus, with probability at least $1-\delta$, $\mathcal{A}$ finds k distinct (ϵ, m)-optimal arms of $\mathcal{C}$ by $o(\frac{k}{\epsilon^2}(\frac{n}{m}+\log\frac{k}{\delta}))$ samples in expectation, contradicting Theorem 1. The proof is complete. □

The lower bound for the Q-FU problem is stated in Corollary 3, which directly follows Theorem 3.3 given by Chaudhuri and Kalyanakrishnan [14] and Theorem 1. Theorem 3.3 of [14] only works for $k=1$, and the lower bound is $\Omega(\frac{n}{m\epsilon^2}\log\frac{1}{\delta})$. Corollary 3 applies for all k values.

Corollary 3 (Lower bound for Q-FU)**.** *Given $k \leq m \leq n/2$, $\epsilon \in (0, 1/\sqrt{32})$, and $\delta \in (0, e^{-8}/40)$, there is a set such that to find k distinct (ϵ, m)-optimal arms with probability at least $1-\delta$, any algorithm must take $\Omega(\frac{1}{\epsilon^2}(\frac{nk}{m}+k\log\frac{k}{\delta}+\frac{n}{m}\log\frac{1}{\delta}))$ samples in expectation.*

The lower bound for the Q-FU problem is stated in Corollary 4, which directly follows Corollary 3.4 given by Chaudhuri and Kalyanakrishnan [14] and Theorem 2. Corollary 3.4 of [14] only works for $k=1$, and its lower bound is $\Omega(\frac{1}{\rho\epsilon^2}\log\frac{1}{\delta})$. Corollary 4 applies for all k values.

Corollary 4 (Lower bound for Q-IU)**.** *Given k, $\rho \in (0, \frac{1}{2}]$, $\epsilon \in (0, 1/\sqrt{32})$, and $\delta \in (0, e^{-8}/40)$, there is an infinite set such that to find k distinct $[\epsilon, \rho]$-optimal arms with probability at least $1-\delta$, any algorithm must take $\Omega(\frac{1}{\epsilon^2}(\frac{k}{\rho}+k\log\frac{k}{\delta}+\frac{1}{\rho}\log\frac{1}{\delta}))$ samples in expectation.*

4 ALGORITHMS FOR THE Q-IK PROBLEM

In this section, we present two Q-IK algorithms: AL-Q-IK and CB-AL-Q-IK. "AL" stands for "algorithm" and "CB" stands for "confidence bounds".

A worst case order-optimal algorithm. We first introduce AL-Q-IK. It calls the function "Median-Elimination" [16], which can find an $(\epsilon, 1)$-optimal arm with probability at least $1-\delta$ by using $O(\frac{|A|}{\epsilon^2}\log\frac{1}{\delta})$ samples. AL-Q-IK is similar to Iterative Uniform Rejection (IUR) [18]. At each repetition, IUR draws an arm from $\mathcal{S}$, performs $\Theta(\frac{1}{\epsilon^2}\log\frac{1}{\delta})$ samples on it, and returns it if the empirical mean is large enough. It solves the Q-IK problem with $k=1$, and its sample complexity is $O(\frac{1}{\epsilon^2\rho}\log\frac{1}{\delta\rho})$. This is higher than the lower bound roughly by a $\frac{1}{\rho}\log\frac{1}{\rho}$ factor (compared with the $\Omega(\frac{1}{\epsilon^2}\log\frac{1}{\delta})$ term). The $\frac{1}{\rho}\log\frac{1}{\rho}$ factor is because the random arm drawn from $\mathcal{S}$ is $[\epsilon, \rho]$-optimal with probability ρ (in the worst case). Inspired by their work, we add Lines 2 and 3 to ensure that a_t is $[\epsilon_1, \rho]$-optimal with probability at least $\frac{1}{2}$. By doing this, we replace the $\frac{1}{\rho}\log\frac{1}{\rho}$ factor by a constant while adding $O(\frac{1}{\rho\epsilon^2})$ samples for each repetition. Repetitions continue until k arms are found, and the number of repetitions is no more than $4k$ in expectation. The choice of n_2 guarantees that for each arm added to Ans, it is $[\epsilon, \rho]$-optimal with probability at least $1-\frac{\delta}{k}$. We state its theoretical performance in Theorem 5.

Algorithm 1 AL-Q-IK$(\mathcal{S}, k, \rho, \epsilon, \delta, \lambda)$

Input: $\mathcal{S}, k, \rho, \epsilon, \delta$, and $\lambda \leq \mathcal{F}^{-1}(1-\rho)$;
Initialize: Choose $\epsilon_1, \epsilon_2 > 0$ with $\epsilon_1 + 2\epsilon_2 = \epsilon$; $t \leftarrow 0$; $Ans \leftarrow \emptyset$; $n_1 \leftarrow \lceil \frac{1}{\rho}\log 3 \rceil$; $n_2 \leftarrow \lceil \frac{1}{2\epsilon_2^2}\log\frac{k}{\delta} \rceil$;
▷ $\epsilon_1, \epsilon_2 = \Omega(\epsilon)$, Ans stores the chosen arms;

```
1:  repeat t ← t + 1;
2:      Draw n_1 arms from S, and form set A_t;
3:      arm a_t ← Median-Elimination(A_t, ε_1, 1/4);
4:      Sample a_t for n_2 times;
5:      μ̂_t ← the empirical mean;
6:      if μ̂_t ≥ λ_ρ − ε_1 − ε_2 then
7:          Ans ← Ans ∪ {a_t};
8:      end if
9:  until |Ans| ≥ k
10: return Ans;
```

Theorem 5 (Theoretical performance of AL-Q-IK)**.** *With probability at least $1-\delta$, AL-Q-IK returns k distinct arms having expected rewards no less than $\lambda-\epsilon$. The expected sample complexity is $O(\frac{k}{\epsilon^2}(\frac{1}{\rho}+\log\frac{k}{\delta}))$.*

Proof Sketch. Correctness: Here we note that $\lambda_\rho \geq \lambda$. At each repetition, n_1 arms are drawn from $\mathcal{S}$ to guarantee that with probability at least $2/3$, the set A_t contains an arm of the top ρ fraction. Then in Line 3, the algorithm calls Median-Elimination$(A_t, \epsilon_1, \frac{1}{4})$ to get a_t, and with probability at least $\frac{2}{3}(1-\frac{3}{4}) = \frac{1}{2}$, a_t is an $[\epsilon_1, \rho]$-optimal arm. At Line 5, by Hoeffding's Inequality, we can prove that if a_t is $[\epsilon_1, \rho]$-optimal, $\hat{\mu}_t$ is greater than $\lambda-\epsilon_1-\epsilon_2$ with probability at least $1-\frac{\delta}{k}$, and if $\mu_{a_t} \leq \lambda-\epsilon$, $\hat{\mu}_t$ is less than $\lambda-\epsilon_1-\epsilon_2$ with probability at least $1-\frac{\delta}{k}$. By some computation, we can show that if a_t is added to Ans, a_t has $\mu_{a_t} \geq \lambda-\epsilon$ with probability at least $1-\frac{\delta}{k}$. Thus, with probability at least $1-\delta$, all arms in Ans having expected rewards $\geq \lambda-\epsilon$. *Sample Complexity:* For each t, a_t is $[\epsilon_1, \rho]$-optimal with probability at least $\frac{1}{2}$, and if a_t is $[\epsilon_1, \rho]$-optimal, then with probability at least $1-\frac{\delta}{k}$, it will be added to Ans. Thus, in the t-th repetition, with probability at least $(1-\frac{\delta}{k})\cdot\frac{1}{2} \geq \frac{1}{4}$, one arm is added to Ans. Thus, the algorithm returns after average $4k$ repetitions. In each repetition, Line 3 takes $O(\frac{n_1}{\epsilon^2}\log 4) = O(\frac{1}{\rho\epsilon^2})$ samples, and Line 4 takes $n_2 = O(\frac{1}{\epsilon^2}\log\frac{k}{\delta})$ samples. The desired sample complexity follows. □

Remark: The expected sample complexity of Algorithm 1 matches the lower bound proved in Theorem 2. Even for $k=1$, this result is better than the previous works $O(\frac{1}{\rho\epsilon^2}\log\frac{1}{\delta})$ [18].

Alternative Version Using Confidence Bounds AL-Q-IK is order-optimal for the worst instances, and provides theoretical insights on the Q-IK problem, but in practice, it does not exploit the large gaps between the arms' expected rewards. In this part, we use confidence bounds to establish an algorithm that is not order-optimal for the worst instance but has better practical performance for most instances. Many previous works [20; 21; 19; 14; 7] have shown that this kind of confidence-bound-based (CBB) algorithms can dramatically reduce the actual number of samples taken in practice. Given an arbitrary arm a with expected reward μ_a, we let $\hat{X}^N(a)$ be its empirical mean after N samples. A function $u(\cdot)$ ($l(\cdot)$) is said to be an upper (lower) δ-confidence bound if it satisfies

$$\mathbb{P}\{u(\hat{X}^N(a), N, \delta) \geq \mu_a\} \geq 1 - \delta, \quad (4)$$

$$\mathbb{P}\{l(\hat{X}^N(a), N, \delta) \leq \mu_a\} \geq 1 - \delta. \quad (5)$$

There are many choices of confidence bounds, e.g., the confidence bounds using Hoeffding's Inequality can be

$$u(\hat{X}^N(a), N, \delta) = \hat{X}^N(a) + \sqrt{\log\delta^{-1}/(2N)}, \quad (6)$$

$$l(\hat{X}^N(a), N, \delta) = \hat{X}^N(a) - \sqrt{\log\delta^{-1}/(2N)}. \quad (7)$$

In this paper, we propose a general algorithm that works for all confidence bounds satisfying (4) and (5). We first introduce PACMaxing (Algorithm 2), an algorithm to find one (ϵ, m)-optimal arm. The idea follows KL-LUCB [22], except that it is designed for all confidence bounds and has a budget to bound the number of samples taken. Adding *budget* prevents the number of samples from blowing up to infinity, and helps establish Algorithm 3.

In PACMaxing, we let $U^t(a) := u(\hat{\mu}^t(a), N^t(a), \delta^{N^t(a)})$ and $L^t(a) := l(\hat{\mu}^t(a), N^t(a), \delta^{N^t(a)})$. For every arm a, PACMaxing guarantees that during the execution of algorithm, with probability at least $1-\frac{\delta}{n}$, its expected reward is always between the lower and upper confidence bounds, and thus, is correct with probability at least $1-\delta$ (see Lemma 6). Lemma 6's proof is similar to that of KL-LUCB [22], and is provided in supplementary materials.

Lemma 6 (Correctness of PACMaxing)**.** *Given sufficiently large budget, PACMaxing returns an* $(\epsilon, 1)$*-optimal arm with probability at least* $1-\delta$*.*

Lemma 6 does not provide any insight about PACMaxing's sample complexity because it depends on the confidence bounds we choose. For Hoeffding bounds defined by (6) and (7), we compute the sample complexity of PACMaxing, stated in Lemma 7. Here we define $\Delta_b := \frac{1}{2}\max\{\epsilon, \max_{a\in A}\mu_a - \mu_b\}$ for all arms b.

Algorithm 2 PACMaxing($A, \epsilon, \delta, budegt$)

Input: A an n-sized set of arms; $\delta, \epsilon \in (0, 1)$;
1: $\forall s, \delta^s := \frac{\delta}{k_1 n s^\gamma}$, where $\gamma > 1$ and $k_1 \geq 2(1+\frac{1}{\gamma-1})$;
2: $t \leftarrow 0$ (number of sample taken);
3: $B(t) \leftarrow \infty$ (stopping index);
4: Sample every arm of A once; $t \leftarrow n$;
5: $N^t(a) \leftarrow 1, \forall a \in A$;(number of times a is sampled)
6: Let $\hat{\mu}^t(a)$ be the empirical mean of a;
7: $a^t \leftarrow \arg\max_a \hat{\mu}^t(a)$;
8: $b^t \leftarrow \arg\max_{a\neq a^t} U^t(a)$;
9: **while** $B(t) > \epsilon \wedge t \leq budget$ **do**
10: Sample a^t and b^t once; $t \leftarrow t+2$;
11: Update $\hat{\mu}^t(a), \hat{\mu}^t(b), N^t(a), N^t(b)$;
12: Update a^t and b^t as Lines 7 and 8;
13: $B(t) \leftarrow U^t(b^t) - L^t(a^t)$;
14: **end while**
15: **if** $B(t) \leq \epsilon$ **then** **return** a^t
16: **else** **return** a random arm
17: **end if**

Lemma 7 (Sample complexity of PACMaxing)**.** *Using confidence bounds (6) (7), and for budget no less than* $3n + \max\left\{\frac{8n}{\epsilon^2}\log\frac{k_1 n}{\delta}, \frac{8(1+e^{-1})\gamma n}{\epsilon^2}\log\frac{4(1+e^{-1})\gamma}{\epsilon^2}\right\}$*, with probability at least* $1-\delta$*, PACMaxing returns a correct result after* $O(\sum_{a\in A}\frac{1}{\Delta_a^2}\log\frac{n}{\delta\Delta_a})$ *samples.*

Its proof is similar to that of KL-LUCB [22], and is relegated to supplementary materials due to space limitation.

Using PACMaxing, we establish the CBB version of AL-Q-IK, presented in Algorithm 3. In the algorithm, we choose g_0, g_1 be the corresponding *budget* lower bounds as in Lemma 7. CB-AL-Q-IK is almost the same as AL-Q-IK, except that it replaces Median-Elimination and the sampling of a_t by PACMaxing.

Algorithm 3 CB-AL-Q-IK($\mathcal{S}, k, \rho, \epsilon, \delta, \lambda$)

Input: $\mathcal{S}, k, \rho, \epsilon, \delta$, and $\lambda \leq \mathcal{F}^{-1}(1-\rho)$;
Initialize: $t \leftarrow 0$; $Ans \leftarrow \emptyset$; $n_1 \leftarrow \lceil\frac{1}{\rho}\log 3\rceil$;
repeat $t \leftarrow t+1$;
  Draw n_1 arms from $\mathcal{S}$, and form set A_t;
  arm $a_t \leftarrow$PACMaxing($A_t, \frac{3}{4}\epsilon, \frac{1}{4}, g_0$);
  Let c be an arm with constant rewards $\lambda - \frac{7}{8}\epsilon$;
  $b_t \leftarrow$PACMaxing($\{a_t, c\}, \frac{\epsilon}{8}, \frac{\delta}{k}, g_1$);
  if $b_t = a_t$ **then**
    $Ans \leftarrow Ans \cup \{a_t\}$;
  end if
until $|Ans| \geq k$
return Ans;

Theorem 8 states the theoretical performance of CB-AL-Q-IK. Its worst case sample complexity is higher than the lower bound and that of AL-Q-IK roughly by a $\log \frac{1}{\rho\epsilon}$ factor. However, since it can exploit the large gaps between the arms, its empirical performance can be much better (See Section 7 for numerical evidences).

Theorem 8 (Theoretical performance of CB-AL-Q-IK)**.** *With probability at least $1-\delta$, CB-AL-Q-IK returns k distinct arms having expected rewards no less than $\lambda - \epsilon$. When using confidence bounds (6) and (7), it terminates after at most $O(\frac{k}{\epsilon^2}(\frac{1}{\rho}\log\frac{1}{\rho\epsilon} + \log\frac{k}{\delta\epsilon}))$ samples in expectation.*

Proof. The correctness follows by directly using the same steps as in the proof of Theorem 5. In each repetition, by Lemma 7, the sample complexity of Line 3 is at most $O(\frac{n_1}{\epsilon^2}\log\frac{n_1}{\rho\epsilon}) = O(\frac{1}{\rho\epsilon^2}\log\frac{1}{\rho\epsilon})$, and that of Line 5 is at most $O(\frac{1}{\epsilon^2}\log\frac{k}{\delta\epsilon})$. The "at most" comes from the choice of *budget* in Lemma 7. The algorithm returns after at most $4k$ repetitions in expectation. The desired sample complexity follows. □

5 ALGORITHMS FOR THE Q-IU PROBLEM

Chaudhuri and Kalyanakrishnan [14] proposed an $O(\frac{1}{\rho\epsilon^2}\log^2\frac{1}{\delta})$ sample complexity algorithm for the $k=1$ case. Obviously, performing it for k times with $\frac{\delta}{k}$ error probability for each can solve the problem for all k values. However, this method will yield unnecessary dependency on $\log^2 k$. If we can first estimate the value of λ_ρ, we can use (CB-)AL-Q-IK to solve this problem and replace the quadratic log dependency by $\log k$. We first use LambdaEstimation to get a "good" estimation of λ_ρ, and then use AL-Q-IK to solve the Q-IU problem.

We first present the algorithm to estimate λ_ρ: Algorithm 4 LambdaEstimation. In this algorithm, we will call Halving [20], which finds k distinct (ϵ, k)-optimal arms of an n-sized set with probability at least $1-\delta$ by taking $O(\frac{n}{\epsilon^2}\log\frac{k}{\delta})$ samples. Halving$_2$ is an algorithm similar to Halving that finds (PAC) worst arms.

Algorithm 4 LambdaEstimation($\mathcal{S}, \rho, \epsilon, \delta$)

Input: $\mathcal{S}$ an infinite set of arms; $\rho, \delta, \epsilon \in (0, 1/2)$;
1: Choose $\epsilon_1, \epsilon_2, \epsilon_3 = \Omega(\epsilon)$ with $\epsilon_1 + \epsilon_2 + 2\epsilon_3 = \epsilon$;
2: $n_3 \leftarrow \lceil \frac{32}{\rho}\log\frac{5}{\delta}\rceil$; $n_4 \leftarrow \lceil \frac{1}{2\epsilon_3^2}\log\frac{10}{\delta}\rceil$; $m \leftarrow \lfloor 1 + \frac{3}{4}\rho n_3 \rfloor$;
3: Draw n_3 arms from $\mathcal{S}$, and form A_1;
4: $A_2 \leftarrow$ Halving($A_1, m, \epsilon_1, \frac{\delta}{5}$);
5: $\hat{a} \leftarrow$ Halving$_2$($A_2, 1, \epsilon_2, \frac{\delta}{5}$);
6: Sample $\hat{a}$ for n_4 times, $\hat{\mu}_0 \leftarrow$the empirical mean;
7: **return** $\hat{\lambda} \leftarrow \hat{\mu}_0 - \epsilon_2 - \epsilon_3$;

In LambdaEstimation, we ensure that with probability at least $1 - \frac{2}{5}\delta$, the m-th most rewarding arm of A_1 is in $M := \{a \in \mathcal{S} : \lambda_\rho \le \mu_a \le \lambda_{\rho/2}\}$. After calling Halving and Halving$_2$, we get $\hat{a}$ whose expected reward is in $[\lambda_\rho - \epsilon_1, \lambda_{\rho/2} + \epsilon_2]$ with probability at least $1 - \frac{4\delta}{5}$. Finally, $\hat{a}$ is sampled for n_4 times, and its empirical mean is in $[\lambda_\rho - \epsilon_1 - \epsilon_3, \lambda_{\rho/2} + \epsilon_2 + \epsilon_3]$ with probability at least $1-\delta$. Thus, the returned value $\hat{\lambda}$ is in $[\lambda_\rho - \epsilon, \lambda_{\rho/2}]$ with probability at least $1-\delta$. Detailed proof of Lemma 9 is provided in supplementary materials.

Lemma 9 (Theoretical performance of LambdaEstimation)**.** *After at most $O(\frac{1}{\rho\epsilon^2}\log^2\frac{1}{\delta})$ samples, LambdaEstimation returns $\hat{\lambda}$ that is in $[\lambda_\rho - \epsilon, \lambda_{\rho/2}]$ with probability at least $1-\delta$.*

Now, we use LambdaEstimation to establish the ALgorithm for the Q-IU problem (AL-Q-IU) (Algorithm 5). Its theoretical performance is stated in Theorem 10.

Algorithm 5 AL-Q-IU($\mathcal{S}, k, \rho, \epsilon, \delta$)

Input: $\mathcal{S}$ infinite; $k \in \mathbb{Z}^+$; $\rho, \delta, \epsilon \in (0, 1/2)$;
1: $\hat{\lambda} \leftarrow$ LambdaEstimation($\mathcal{S}, \rho, \frac{\epsilon}{2}, \frac{\delta}{2}$);
2: **return** AL-Q-IK($\mathcal{S}, k, \frac{\rho}{2}, \frac{\epsilon}{2}, \frac{\delta}{2}, \hat{\lambda}$);

Theorem 10 (Theoretical performance of AL-Q-IU)**.** *With probability at least $1-\delta$, AL-Q-IU returns k distinct $[\epsilon, \rho]$-optimal arms. With probability at least $1-\frac{\delta}{2}$, it terminates after $O(\frac{1}{\epsilon^2}(\frac{1}{\rho}\log^2\frac{1}{\delta} + k(\frac{1}{\rho} + \log\frac{k}{\delta})))$ samples in expectation.*

Proof. With probability at least $1-\frac{\delta}{2}$, $\hat{\lambda}$ is in $[\lambda_\rho - \frac{\epsilon}{2}, \lambda_{\rho/2}]$. When $\hat{\lambda}$ is in $[\lambda_\rho - \frac{\epsilon}{2}, \lambda_{\rho/2}]$, by Theorem 5, Line 2 takes $O(\frac{k}{\epsilon^2}(\frac{1}{\rho} + \log\frac{1}{\delta}))$ samples in expectation, and, with probability at least $1-\frac{\delta}{2}$, all returned arms are $[\epsilon, \rho]$-optimal. The correctness of AL-Q-IU follows. The desired sample complexity follows by summing up $O(\frac{k}{\epsilon^2}(\frac{1}{\rho} + \log\frac{1}{\delta}))$ and $O(\frac{1}{\epsilon^2}\log^2\frac{1}{\delta})$ (Lemma 9). □

Remark: By Corollary 4, AL-Q-IU is sample complexity optimal up to a $\log\frac{1}{\delta}$ factor. When $\log\frac{1}{\delta} = O(k)$, i.e., $\delta \ge e^{-ck}$ for some constant $c > 0$, AL-Q-IU is sample complexity optimal up to a constant factor.

6 ALGORITHMS FOR THE FINITE CASES

In this section, we let $\mathcal{S}$ be a finite-sized set of arms. By drawing arms from it with replacement, these arms can be regarded as drawn from an infinite-sized set. We use $\mathcal{T}(\mathcal{S})$ to denote the corresponding infinite-sized set, and call it the *infinite extension* of $\mathcal{S}$.

Q-FK. When $k=1$, obviously, calling AL-Q-IK($\mathcal{T}(\mathcal{S}), 1, \frac{m}{n}, \epsilon, \delta, \lambda_\rho$) can solve the Q-FK problem. When $k > 1$, we can solve the Q-FK problem by repeatedly calling AL-Q-IK($\mathcal{T}(\mathcal{S}), 1, \rho_t, \epsilon, \delta/k, \lambda_\rho$) and updating $\mathcal{S}$ by deleting the chosen arm, where $\rho_t = \frac{m+1-t}{n+1-t}$. We present the algorithm AL-Q-FK (ALgorithm for Q-FK) in Algorithm 6, and state the theoretical performance in Theorem 11. The proof is relegated to supplementary materials.

Algorithm 6 AL-Q-FK($\mathcal{S}, m, k, \epsilon, \delta, \lambda$)

Require: $\mathcal{S}$ n-sized, $k \le m \le n/2$, $\lambda \le \lambda_{[m]}$;
Initialize: $Ans \leftarrow \emptyset$; ▷ stores the chosen arms;
1: **repeat**
2: $\mathcal{S}' \leftarrow \mathcal{T}(\mathcal{S} - Ans)$; $\rho \leftarrow \frac{m-|Ans|}{n-|Ans|}$;
3: $a_t \leftarrow$AL-Q-IK($\mathcal{S}', 1, \rho, \epsilon, \frac{\delta}{k}, \lambda$);
4: $Ans \leftarrow Ans \cup \{a_t\}$;
5: **until** $|Ans| \ge k$
6: **return** Ans;

Theorem 11 (Theoretical performance of AL-Q-FK). *With probability at least $1-\delta$, AL-Q-FK returns k distinct arms having mean rewards at least $\lambda-\epsilon$. Its takes $O(\frac{1}{\epsilon^2}(n \log \frac{m+1}{m+1-k} + k \log \frac{k}{\delta}))$ samples in expectation.*

Remark: If $k \le cm$ for some constant $c < 1$, $\log \frac{m+1}{m+1-k} \le \frac{k}{m+1-k} = O(\frac{k}{m})$, and thus, the expected sample complexity becomes $O(\frac{k}{\epsilon^2}(\frac{k}{m} + \log \frac{k}{\delta}))$, meeting the lower bound (Theorem 1). When k is arbitrarily close to m, the Q-FK problem (almost) reduces to the KE problem. The tightest upper bound for the KE problem (with the knowledge of $\lambda_{[k]}$) is $O(\frac{n}{\epsilon^2} \log \frac{k}{\delta})$ [21] to our best knowledge. When k is arbitrary close to m, as $O(\frac{1}{\epsilon^2}(n \log \frac{m+1}{m+1-k} + k \log \frac{k}{\delta})) = O(\frac{1}{\epsilon^2}(n \log k + k \log \frac{k}{\delta}))$, AL-Q-FK is still better than the literature asymptotically.

Q-FU. Algorithm 7 AL-Q-FU (ALgorithm for Q-FU) solves the Q-FU problem. Its idea follows AL-Q-IU and AL-Q-FK. We only consider the case $k < \frac{m}{2}$. For $k \ge \frac{m}{2}$, it is better to use KE algorithms instead. Theorem 12 states its theoretical performance.

Algorithm 7 AL-Q-FU($\mathcal{S}, m, k, \epsilon, \delta$)

Require: $\mathcal{S}$ n-sized; $2k < m \le n/2$;
1: $\hat{\lambda} \leftarrow$LambdaEstimation($\mathcal{T}(\mathcal{S}), \frac{m}{n}, \frac{\epsilon}{2}, \frac{\delta}{2}$);
2: **return** AL-Q-FU($\mathcal{S}, \lfloor \frac{m}{2} \rfloor, k, \frac{\epsilon}{2}, \frac{\delta}{2}$);

Theorem 12 (Theoretical performance of AL-Q-FU). *With probability at least $1-\delta$, AL-Q-FU returns k distinct (ϵ, m)-optimal arms. With probability at least $1-\frac{\delta}{2}$, It returns after $O(\frac{1}{\epsilon^2}(\frac{n}{m} \log^2 \frac{1}{\delta} + n \log \frac{m+2}{m+2-2k} + k \log \frac{k}{\delta}))$ samples in expectation.*

Proof. The proof follows immediately from that of Theorem 11 and Theorem 10. □

Remark: By Corollary 3, when $k \le cm$ for some constant $c \in (0, \frac{1}{2})$, AL-Q-FU is sample complexity optimal up to a $\log \frac{1}{\delta}$ factor. If $\log \frac{1}{\delta} = O(k)$ also holds, i.e., $\delta \ge e^{-ck}$ for some constant $c > 0$, AL-Q-FU is sample complexity optimal in order sense.

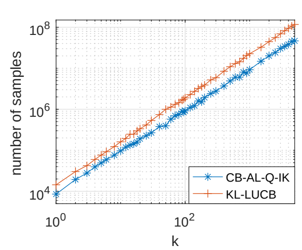
7 NUMERICAL RESULTS

In this section, we illustrate the improvements of our algorithms by running numerical experiments. Due to space limitation, only the results for the Q-IK problem are presented. In the supplementary materials, we present additional numerical results about the Q-IU problem and the finite case.

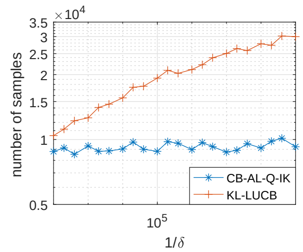
As has been shown above, the worst case performance of non-CBB algorithms are better than CBB ones and provide better theoretical insights, but their practical performance may not be better under most instances and parameters. For fair comparisons, we first compare the CBB algorithms, and then compare the non-CBB algorithms. In the end, we will discuss the non-CBB algorithms and CBB ones.

In the simulations, we adopt Bernoulli rewards for all the arms. For fair comparisons, for all CBB-algorithms or versions, we use the KL-Divergence based confidence bounds given by Aziz et al. [7]. Every point in every figure is averaged over 100 independent trials. The priors $\mathcal{F}$ of all experiments are Uniform([0,1]). Previous works only considered the case where $k = 1$. In the implementations, for $k > 1$, we repeat them for k times, each of which is with error probability $\frac{\delta}{k}$.

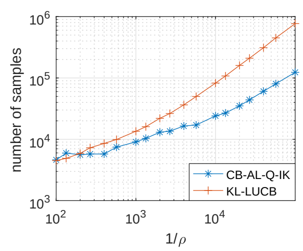
First, we compare CBB algorithms: CB-AL-Q-IK (choose $\epsilon_1 = 0.8\epsilon$) and (α, ϵ)-KL-LUCB [7] (we name it KL-LUCB in this section). KL-LUCB is almost equivalent to $\mathcal{P}_2$ [14] with a large enough batch size. The only difference is that they choose different confidence bounds. Here we note that KL-LUCB does not require the knowledge of λ_ρ, but we want to show that our algorithm along with this information can significantly reduce the actual number of samples needed. The results are summarized in Figure 1 (a)-(d). It can be seen from Figure 1 that CB-AL-Q-IK performs better than KL-LUCB except two or three points where ρ is large. According to (a), the number of samples CB-AL-Q-IK takes increases slightly slower than KL-LUCB, consistent with the theory that CB-AL-Q-IK depends on $k \log k$ while KL-LUCB depends on $k \log^2 k$. According to (b), we can see that KL-LUCB's number of samples increases obviously with $\frac{1}{\delta}$, while that of CB-AL-Q-IK is almost independent of δ. The reason is that CB-AL-Q-IK depends on $(\frac{1}{\rho} \log \frac{1}{\rho} + \log \frac{1}{\delta})$ term, and when ρ is small enough, $\log \frac{1}{\delta}$ can be dominated by $\frac{1}{\rho} \log \frac{1}{\rho}$. According to (c), CB-AL-Q-IK takes less samples than KL-LUCB for $\rho < 0.005$, and the gap increases with $\frac{1}{\rho}$. According to (d), CB-AL-Q-IK per-



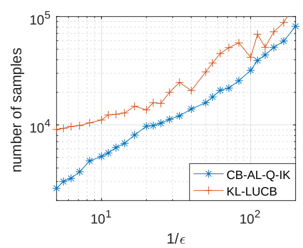
(a) Vary k, $\rho = 0.001$,$\epsilon = 0.05$, and $\delta = 0.001$.



(b) Vary δ, $k = 1$, $\rho = 0.001$, and $\epsilon = 0.05$.

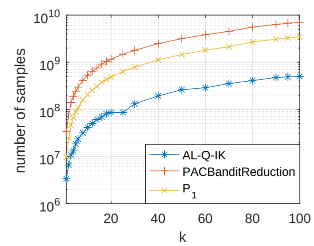


(c) Vary ρ, $k = 1$, $\epsilon = 0.05$, and $\delta = 0.001$.

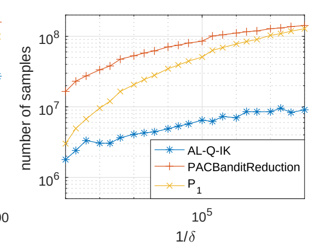


(d) Vary ϵ, $k = 1$, $\rho = 0.001$, and $\delta = 0.001$.

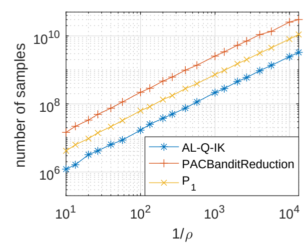
Figure 1: Comparison of CB-AL-Q-IK and KL-LUCB



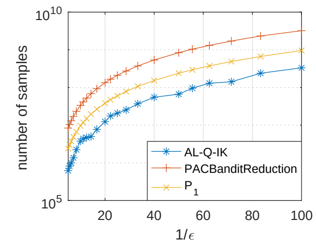
(a) Vary k, $\rho = 0.05$,$\epsilon = 0.1$, and $\delta = 0.01$.



(b) Vary δ, $k = 1$, $\rho = 0.05$, and $\epsilon = 0.1$.



(c) Vary ρ, $k = 1$, $\epsilon = 0.1$, and $\delta = 0.01$.



(d) Vary ϵ, $k = 1$, $\rho = 0.05$, and $\delta = 0.01$.

Figure 2: Comparison of Non-CBB Algorithms.

forms better than KL-LUCB given under the given ϵ values.

Second, we compare non-CBB algorithms: AL-Q-IK, PACBanditReduction [18], and $\mathcal{P}_1$ [14]. Here, again, we note that $\mathcal{P}_1$ does not require the knowledge of λ_ρ, but we want to illustrate how our algorithm along with this knowledge can improve the efficiency. The results are summarized in Figure 2 (a)-(d). The theoretical sample complexities of these three algorithms are: AL-Q-IK, $O(\frac{k}{\epsilon^2}(\frac{1}{\rho} + \log\frac{k}{\delta}))$; PACBanditReduction, $O(\frac{k}{\rho\epsilon^2}\log\frac{k}{\delta})$; $\mathcal{P}_1$, $O(\frac{k}{\rho\epsilon^2}\log^2\frac{k}{\delta})$. The numerical results confirm that AL-Q-IK performs better than the other two significantly. Figure 2 (b) shows that AL-Q-IK's sample complexity increase slowly with $\frac{1}{\delta}$, consistent with the theory and numerical results on CB-AL-Q-IK.

According to Figure 1 (c) and Figure 2 (c), the CB-AL-Q-IK's number of samples increases super-linearly with $\frac{1}{\rho}$ while that of AL-Q-IK increases linearly, consistent with the theory that the former depends on $\frac{1}{\rho}\log\frac{1}{\rho}$ while the latter depends on $\frac{1}{\rho}$. When $\frac{1}{\rho}$ is large enough, asymptotically AL-Q-IK will outperform CB-AL-Q-IK. However, in practice, under such small ρ values, the sample complexity of both algorithms will be extremely large.

8 CONCLUSION

In this paper, we studied the problems of finding k top ρ fraction arms with an ϵ bounded error from a finite or infinite arm set. We considered both cases where the thresholds (i.e., λ_ρ and $\lambda_{[m]}$) are priorly known and unknown. We derived lower bounds on the sample complexity for all four settings, and proposed algorithms for them. For the Q-IK and Q-FK problems, our algorithms match the lower bounds. For the Q-IU and Q-FU problems, our algorithms are sample complexity optimal up to a log factor. Our simulations also confirm these improvements numerically.

References

[1] Agarwal, A., Agarwal, S., Assadi, S., and Khanna, S. (2017). Learning with limited rounds of adaptivity: Coin tossing, multi-armed bandits, and ranking from pairwise comparisons. In *Conference on Learning Theory*.

[2] Agrawal, S. and Goyal, N. (2012). Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, pages 39–1.

[3] Arratia, R. and Gordon, L. (1989). Tutorial on large deviations for the binomial distribution. *Bulletin of mathematical biology*, pages 125–131.

[4] Audibert, J. and Bubeck, S. (2010). Best arm identification in multi-armed bandits. In *COLT-23th Conference on Learning Theory-2010*.

[5] Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine learning*.

[6] Auer, P. and Ortner, R. (2010). Ucb revisited: Im-

proved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*.

[7] Aziz, M., Anderton, J., Kaufmann, E., and Aslam, J. (2018). Pure exploration in infinitely-armed bandit models with fixed-confidence.

[8] Berry, D. A. and Eick, S. G. (1995). Adaptive assignment versus balanced randomization in clinical trials: a decision analysis. *Statistics in medicine*.

[9] Berry, D. A. and Fristedt, B. (1985). Bandit problems: sequential allocation of experiments (monographs on statistics and applied probability). *London: Chapman and Hall*, 5:71–87.

[10] Bubeck, S. and Cesa-Bianchi, N. (2012). Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*.

[11] Bubecka, S., Munosa, R., and Stoltzb, G. (2011). Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412:1832–1852.

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

[13] Carpentier, A. and Valko, M. (2015). Simple regret for infinitely many armed bandits. In *International Conference on Machine Learning*.

[14] Chaudhuri, A. R. and Kalyanakrishnan, S. (2017). Pac identification of a bandit arm relative to a reward quantile. In *AAAI*.

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

[16] Even-Dar, E., Mannor, S., and Mansour, Y. (2002). Pac bounds for multi-armed bandit and markov decision processes. In *International Conference on Computational Learning Theory*. Springer.

[17] Garivier, A. and Cappé, O. (2011). The kl-ucb algorithm for bounded stochastic bandits and beyond. In *Proceedings of the 24th annual Conference On Learning Theory*, pages 359–376.

[18] Goschin, S., Weinstein, A., Littman, M. L., and Chastain, E. (2013). Planning in reward-rich domains via pac bandits. In *European Workshop on Reinforcement Learning*.

[19] Jamieson, K., Malloy, M., Nowak, R., and Bubeck, S. (2014). lilucb: An optimal exploration algorithm for multi-armed bandits. In *Conference on Learning Theory*.

[20] Kalyanakrishnan, S. and Stone, P. (2010). Efficient selection of multiple bandit arms: Theory and practice. In *ICML*.

[21] Kalyanakrishnan, S., Tewari, A., Auer, P., and Stone, P. (2012). Pac subset selection in stochastic multi-armed bandits. In *ICML*.

[22] Kaufmann, E. and Kalyanakrishnan, S. (2013). Information complexity in bandit subset selection. In *Conference on Learning Theory*.

[23] Kohavi, R., Longbotham, R., Sommerfield, D., and Henne, R. M. (2009). Controlled experiments on the web: survey and practical guide. *Data mining and knowledge discovery*.

[24] Li, L., Chu, W., Langford, J., and Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*. ACM.

[25] Mannor, S. and Tsitsiklis, J. N. (2004). The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*.

[26] Scott, S. L. (2010). A modern bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*.

Supplementary Materials

1 PROOF OF THEOREM 1

Proof. **For $\mathbf{k=1}$**. We first prove the lower bound for $k = 1$.

Claim 1 (Lower bound for Q-FK with $k = 1$)**.** *There is an instance such that to find an (ϵ, m)-optimal arm of it, any algorithm must use $\Omega(\frac{1}{\epsilon^2}(\frac{n}{m}+\log\frac{1}{\delta}))$ samples in expectation.*

Proof. Let parameters n, m, ϵ, and δ be given. For these parameters, suppose there is an algorithm $\mathcal{A}_1$ which solves every Q-FK instance with average sample complexity $o(\frac{1}{\epsilon^2}(\frac{n}{m}+\log\frac{1}{\delta}))$. We introduce the following problem $\mathcal{P}_1$.

Problem $\mathcal{P}_1$: Given $\lfloor n/m \rfloor$ coins, where a toss of coin i has an unknown probability p_i to produce a head, and produce a tail otherwise. We name p_i the "head probability" of coin i. Let p_{max} be the largest one among all p_i's. Knowing the value of p_{max}, we want to find a coin whose head probability is no less than $p_{max} - \epsilon$, and the error probability is no more than δ.

Theorem 13 proved by Mannor and Tsitsiklis [25] proves that the worst case sample complexity lower bound of $\mathcal{P}_1$ is $\Omega(\frac{1}{\epsilon^2}(\frac{n}{m}+\log\frac{1}{\delta}))$. Here we will show that $\mathcal{A}_1$ can solve $\mathcal{P}_1$ with average sample complexity $o(\frac{1}{\epsilon^2}(\frac{n}{m}+\log\frac{1}{\delta}))$, implying a contradiction.

Now, we make "duplications" of these coins. Let $\mathcal{C}_1$ be the set of the coins in $\mathcal{P}_1$. For each coin i, we "duplicate" it for $m-1$ times and construct $m-1$ "duplicated" coins. Whenever one wants to sample a duplication of coin i, coin i will be tossed but the result is regarded as that of the duplication. Thus, we guarantee that all the duplications of coin i have the same head probability as coin i.

With these duplications, we construct a new set $\mathcal{C}_2$ of coins with size n. $\mathcal{C}_2$ consists of all the coins of $\mathcal{C}_1$, all the duplications of all coins in set $\mathcal{C}_1$, and $(n - m\lfloor n/m \rfloor)$ "fake" coins with head probability 0. Set $\mathcal{C}_2$ consist of n coins. For each p_i defined in $\mathcal{P}_1$, there are m coins with head probability p_i in $\mathcal{C}_2$. The fake coins are used to make the size of $\mathcal{C}_2$ be n.

Then, we perform $\mathcal{A}_1$ on the set $\mathcal{C}_2$. It returns an (ϵ, m)-optimal coin (coins can be regarded as arms with Bernoulli(p_i) rewards) of $\mathcal{C}_2$ with probability at least $1-\delta$, and uses $o(\frac{1}{\epsilon^2}(\frac{1}{\rho}+\log\frac{1}{\delta}))$ samples in expectation. We use c_r to denote the returned coin. Let coin i^* be one of the coins whose head probability are p_{max} (i.e., one of the most biased coins of $\mathcal{C}_1$). Since coin i^* is duplicated for $m-1$ times, there are at least m coins in $\mathcal{C}_2$ having head probability p_{max}. This implies that if c_r is an (ϵ, m)-optimal coin of $\mathcal{C}_2$, then its head probability is at least $p_{max} - \epsilon$. If c_r is a fake coin, we return a random coin of $\mathcal{C}_1$ as the solution of $\mathcal{P}_1$. If c_r is coin i or one of its duplications, we return coin i as the solution of $\mathcal{P}_1$. Noting that the "fake" coins are not (ϵ, m)-optimal, so if c_r is an (ϵ, m)-optimal coin of $\mathcal{C}_2$, there is a corresponding coin in $\mathcal{C}_1$ having the same probability as c_r. Thus, if $\mathcal{A}_1$ finds an (ϵ, m)-coin of $\mathcal{C}_2$, it finds a coin of $\mathcal{C}_1$ whose head probability is at least $p_{max} - \epsilon$, which gives a correct solution of $\mathcal{P}_1$. To conclude, $\mathcal{A}_1$ solves $\mathcal{P}_1$ with average sample complexity $o(\frac{1}{\epsilon^2}(\frac{n}{m}+\log\frac{1}{\delta}))$, contradicting Theorem 13 [25]. This completes the proof of Claim 1. □

For $\mathbf{k>1}$. Now we consider the case where $k > 1$. By contradiction, suppose there is an algorithm $\mathcal{A}_2$ which solves all the instances of the Q-FK problem by $o(\frac{k}{\epsilon^2}(\frac{n}{m}+\log\frac{k}{\delta}))$ samples in expectation.

Let $\mathcal{C}_3$ be an $\lfloor n/k \rfloor$-sized set such that no algorithm can find one $(\epsilon, \lfloor m/k \rfloor)$-optimal arm of it with probability $1-\delta$ by $o(\frac{1}{\epsilon^2}(\frac{n}{m}+\log\frac{1}{\delta}))$ samples in expectation. Claim 1 guarantees that this set must exist. Choose $L = lk$ (for some integer l) large enough to satisfy

$$L > k \log L + k + m. \tag{8}$$

By randomly reordering the indexes of arms in $\mathcal{C}_3$, we can construct another $L-1$ sets of arms that also meet the lower bound stated in Claim 1. We refer to these L sets as *hard* sets. Now we define problem $\mathcal{P}_2$ by these L *hard* sets.

Problem $\mathcal{P}_2$: Given the above L *hard* sets, we want to find one $(\epsilon, \lfloor m/k \rfloor)$-optimal arm for each set, and the total error probability is no more than δ.

Claim 2 (Lower bound of $\mathcal{P}_2$)**.** *To solve $\mathcal{P}_2$, at least $\Omega(\frac{L}{\epsilon^2}(\frac{n}{m}+\log\frac{L}{\delta}))$ samples are needed in expectation.*

Proof. Let δ_i be the error probability for the i-th *hard* set. We have that $\prod_{i=1}^{L}(1-\delta_i) \geq 1-\delta$. Besides, by the definition of *hard* sets, we have that for the i-th *hard* set, to find an $(\epsilon, \lceil m/k \rceil)$-optimal arm of it with probability $1-\delta_i$, at least $\Omega(\frac{1}{\epsilon^2}(\frac{n}{m}+\log\frac{1}{\delta_i}))$ samples are needed in expectation. Thus, to solve $\mathcal{P}_2$, at least $\Omega(\frac{1}{\epsilon^2}(\frac{nL}{m}+\sum_{i=1}^{L}\log\frac{1}{\delta_i}))$ samples are needed in expectation. By the convexity of $\sum_{i=1}^{L}\log\frac{1}{\delta_i}$ and its symmetry property, under the constraint $\prod_{i=1}^{L}(1-\delta_i) \geq 1-\delta$, as $\delta \downarrow 0$, we have that

$$\sum_{i=1}^{L}\log\frac{1}{\delta_i} = \Omega\left(L\log\frac{L}{\delta}\right). \tag{9}$$

Thus, to solve $\mathcal{P}_2$, any algorithm must use

$$\Omega\left(\frac{1}{\epsilon^2}\left(\frac{nL}{m}+\sum_{i=1}^{L}\log\frac{1}{\delta_i}\right)\right)=\Omega\left(\frac{L}{\epsilon^2}\left(\frac{n}{m}+\log\frac{L}{\delta}\right)\right), \tag{10}$$

samples in expectation. This completes the proof of Claim 2. □

Claim 3. *If there exists an algorithm $\mathcal{A}_2$ that can use $o(\frac{k}{\epsilon^2}(\frac{N}{M}+\log\frac{k}{\delta_1}))$ samples in expectation to find k distinct (ϵ, M)-optimal arms of an N-sized set with probability $1-\delta_1$ for $\delta_1 \in (0,\delta]$, then we can construct another algorithm $\mathcal{A}_3$ that solves $\mathcal{P}_2$ by $o(\frac{L}{\epsilon^2}(\frac{n}{m}+\log\frac{L}{\delta}))$ samples in expectation.*

Proof. We use $\mathcal{A}_2$ to construct a new algorithm $\mathcal{A}_3$ that consists of phases and solves $\mathcal{P}_2$. We will show that $\mathcal{A}_3$ solves $\mathcal{P}_2$ by $o(\frac{L}{\epsilon^2}(\frac{n}{m}+\log\frac{L}{\delta}))$ samples in expectation to lead to a contradiction against Claim 2. Construct a new set $\mathcal{C}_4$ consisting of all these L *hard* sets mentioned above. In each phase, we repeatedly call $\mathcal{A}_2$ on $\mathcal{C}_4$, and each call is with error probability $\frac{\delta}{2l\log\frac{2L}{\delta}}$. By the definition of $\mathcal{A}_2$, each call of it in $\mathcal{A}_3$ takes

$$o\left(\frac{k}{\epsilon^2}\left(\frac{Ln/k}{Lm/k}+\log\frac{k\cdot 2l\log\frac{2L}{\delta}}{\delta}\right)\right) = o\left(\frac{k}{\epsilon^2}\left(\frac{n}{m}+\log\frac{L}{\delta}\right)\right) \tag{11}$$

samples in expectation and returns k arms. We call all the returned arms *found* arms. For each *found* arm, if it belongs to the i-th *hard* set, we remove all arms of the i-th *hard* set from $\mathcal{C}_4$. If k *hard* sets have been removed in this phase, we immediately end this phase. Repeat the phases until all *hard* sets are removed from $\mathcal{C}_4$. When an *hard* set is removed, we assign the corresponding *found* arm as one of its $(\epsilon, \lfloor m/k\rfloor)$-optimal arm. When all *hard* sets are removed, we get a solution for $\mathcal{P}_2$.

First, we prove that $\mathcal{A}_3$ correctly solves $\mathcal{P}_2$ with probability at least $1-\delta$. After $l\log\frac{2L}{\delta}$ calls of $\mathcal{A}_2$, there are $L\log\frac{2L}{\delta}$ *found* arms. For every i, the probability that none of these *found* arms belongs to the i-th *hard* set is at most

$$(1-\frac{1}{L})^{L\log\frac{2L}{\delta}} \le \frac{\delta}{2L}. \tag{12}$$

Thus, with probability at least $1-\delta/2$, after $l\log\frac{2L}{\delta}$ calls of $\mathcal{A}_2$, $\mathcal{A}_3$ identifies a *found* arm for every *hard* set, and, in result, terminates. Also, with probability at least $1-\delta/2$, all these $l\log\frac{2L}{\delta}$ calls of $\mathcal{A}_2$ are correct, i.e., every *found* arm is an $(\epsilon, \lfloor m/k\rfloor)$-optimal arm of the corresponding *hard* set. Thus, the solution obtained by $\mathcal{A}_3$ is correct with probability at least $1-\delta$.

Next, we show that the expected sample complexity of $\mathcal{A}_3$ is $o(\frac{L}{\epsilon^2}(\frac{n}{m}+\log\frac{L}{\delta}))$. For $1\le t\le l$, let T_t be the number of *found* arms obtained in phase t. Let S^t be the set of *found* arms obtained in phase t. We observe the arms of S^t one by one, and mark an *hard* set immediately after one arm of its is observed. Let τ_i^t be the number of arms observed between the marking of the $(i-1)$-th marked set and the i-th marked set. For each observation during this period, since there are $(l+1-t)k$ *hard* sets in total, and $i-1$ of them are already marked, with probability $\frac{(l+1-t)k+1-i}{(l+1-t)k}$, the observed arm belongs to an unmarked *hard* set, and cause it to be marked. Thus, we have that

$$\mathbb{E}\tau_i^t = \frac{(l+1-t)k}{(l+1-t)k+1-i}. \tag{13}$$

Thus, in phase t ($t<l$), we have that

$$\begin{aligned}\mathbb{E}\sum_{i=1}^{k}\tau_i^t &= \sum_{i=1}^{k}\frac{(l+1-t)k}{(l+1-t)k+1-i}\\ &\le (l+1-t)k\int_{(l-t)k}^{(l+1-t)k}\frac{1}{x}\,\mathrm{d}x\\ &= (l+1-t)k\cdot\log\frac{l+1-t}{l-t}.\end{aligned} \tag{14}$$

In phase l, we have that

$$\begin{aligned}\mathbb{E}\sum_{i=1}^{k}\tau_i^t &= \sum_{i=1}^{k}\frac{k}{k+1-i}\\ &\le k+k\int_{1}^{k}\frac{1}{x}\,\mathrm{d}x = k+k\log k.\end{aligned} \tag{15}$$

For each phase t, after k sets are marked, only the *found* arms obtained by the last call of $\mathcal{A}_2$ are not observed, as the next call of $\mathcal{A}_2$ belongs to phase $(t+1)$. We recall that T_t is the number of samples taken in phase t, and $\mathcal{A}_3$ terminates after l phases. By (12), (14), and (15), it follows that

$$\begin{aligned}\mathbb{E}\sum_{t=1}^{l}T_t \le& \mathbb{E}\sum_{t=1}^{l}\left[k+\sum_{i=1}^{k}\tau_i^t\right]\\ \le& lk+k+k\log k\\ &+k\sum_{t=1}^{l-1}(l+1-t)\log\frac{l+1-t}{l-t}/\end{aligned} \tag{16}$$

By (12), we have that

$$lk+k+k\log k \le 2L. \tag{17}$$

It also holds that

$$\begin{aligned}
&k\sum_{t=1}^{l-1}(l+1-t)\log\frac{l+1-t}{l-t}\\
\overset{(a)}{=}&k\sum_{s=2}^{l}s\log\frac{s}{s-1}\\
=&k\sum_{s=2}^{l}s\log\left(1+\frac{1}{s-1}\right)\\
\leq&k\sum_{s=2}^{l}\frac{s}{s-1}=k\left(l-1+\sum_{s=2}^{l}\frac{1}{s-1}\right) && (18)\\
\leq&k\left(\log(l-1)+l\right)\leq 2L, && (19)
\end{aligned}$$

where (a) is letting $s=l+1-t$. Thus, recalling that $L=lk$, after average $4l$ calls of $\mathcal{A}_2$, problem $\mathcal{P}_2$ is solved. By (11), each call of $\mathcal{A}_2$ takes $o(\frac{k}{\epsilon^2}(\frac{n}{m}+\log\frac{L}{\delta}))$ samples in expectation. Thus, the expected number of samples $\mathcal{A}_3$ takes is $o(l\frac{k}{\epsilon^2}(\frac{n}{m}+\log\frac{L}{\delta}))=o(\frac{L}{\epsilon^2}(\frac{n}{m}+\log\frac{L}{\delta}))$. This completes the proof of Claim 3. □

If the $\mathcal{A}_2$ assumed in Claim 3 exists, it will lead to a contradiction against Claim 2. This completes the proof of Theorem 1. □.

2 PROOF OF THEOREM 5

Let $k\in\mathbb{Z}^+$,$\rho,\epsilon,\delta\in(0,\frac{1}{2})$,$\lambda\leq\lambda_\rho$ be given. For $p,x\in(0,1)$, we define $U_p:=\{a\in\mathcal{S}:\mu_a\geq\lambda_p\}$, $E_x:=\{a\in\mathcal{S}:\mu_a\geq\lambda-x\}$, and $F_x:=\mathcal{S}-E_x=\{a\in\mathcal{S}:\mu_a<\lambda-x\}$.

In the t-th loop, by (2) and the choice of n_1 in AL-Q-IK, we have that

$$\begin{aligned}\mathbb{P}\{|A_t\cap U_\rho|=0\}&\leq(1-\rho)^{n_1}\\&=e^{n_1\log(1-\rho)}\leq e^{-n_1\rho}\leq\frac{1}{3}.\end{aligned}\tag{20}$$

Given the condition $|A_t\cap U_\rho|>0$, since a_t is the returned value of Median-Elimination($A_t,\epsilon_1,\frac{1}{4}$), by Theorem 4 [16], a_t is with probability at least $\frac{3}{4}$ in E_{ϵ_1}. Thus, we can conclude that

$$\mathbb{P}\{a_t\in E_{\epsilon_1}\}\geq(1-\frac{1}{3})\frac{3}{4}=\frac{1}{2}.\tag{21}$$

In Line 4, we sample a_t for n_2 times, and its empirical mean is $\hat{\mu}_t$. Define $\mathcal{E}_t:=$ the event that a_t is included in the returned value Ans. Since $\mathcal{E}_t$ happens if and only if $\hat{\mu}_t\geq\lambda-\epsilon_1-\epsilon_2$, by Hoeffding's inequality and $n_2=\lceil\frac{1}{2\epsilon_2^2}\log\frac{k}{\delta}\rceil$, it holds that

$$\mathbb{P}\left\{\mathcal{E}_t^{\complement}\mid a_t\in E_{\epsilon_1}\right\}\leq\exp\left\{-2n_2\left(\epsilon_2^2\right)\right\}\leq\frac{\delta}{k},\tag{22}$$

$$\mathbb{P}\left\{\mathcal{E}_t\mid a_t\in F_{\epsilon}\right\}\leq\exp\left\{-2n_2\left(\epsilon_2^2\right)\right\}\leq\frac{\delta}{k}.\tag{23}$$

Since $\{a_t\in E_{\epsilon_1}\}\cap\{\hat{\mu}_t\geq\lambda-\epsilon_1-\epsilon_2\}\subset\mathcal{E}_t$, by (21) and (22), we have

$$\mathbb{P}\{\mathcal{E}_t\}\geq\frac{1}{2}(1-\frac{\delta}{k})\geq\frac{1}{4}.\tag{24}$$

Besides, by (21), (22), and (23), we have

$$\begin{aligned}\frac{\mathbb{P}\left\{a_t\in E_\epsilon\mid\mathcal{E}_t\right\}}{\mathbb{P}\left\{a_t\in F_\epsilon\mid\mathcal{E}_t\right\}}&\geq\frac{\mathbb{P}\left\{a_t\in E_{\epsilon_1}\mid\mathcal{E}_t\right\}}{\mathbb{P}\left\{a_t\in F_\epsilon\mid\mathcal{E}_t\right\}}\\&=\frac{\mathbb{P}\left\{a_t\in E_{\epsilon_1}\right\}\mathbb{P}\left\{\mathcal{E}_t\mid a_t\in E_{\epsilon_1}\right\}}{\mathbb{P}\left\{a_t\in F_\epsilon\right\}\mathbb{P}\left\{\mathcal{E}_t\mid a_t\in F_\epsilon\right\}}\\&\geq\frac{\frac{1}{2}\cdot(1-\frac{\delta}{k})}{\frac{1}{2}\cdot\frac{\delta}{k}}=\frac{k}{\delta}-1.\end{aligned}\tag{25}$$

Since $\mathbb{P}\{a_t\in E_\epsilon\mid\mathcal{E}_t\}+\mathbb{P}\{a_t\in F_\epsilon\mid\mathcal{E}_t\}=1$, we can conclude that

$$\mathbb{P}\left\{a_t\in E_\epsilon\mid\mathcal{E}_t\right\}\geq1-\frac{\delta}{k}.\tag{26}$$

This shows that when an arm a_t is added to Ans, with probability at least $1-\frac{\delta}{k}$, a_t is in E_ϵ. Thus, we have

$$\mathbb{P}\{\forall a_t\in Ans,a_t\in E_\epsilon\}\geq1-\delta.\tag{27}$$

Thus, the returned arms of AL-Q-IK all have expected rewards no less than $\lambda-\epsilon$ with probability at least $1-\delta$. This completes the proof of correctness.

It remains to derive the sample complexity. In each repetition, the algorithm calls Median-Elimination($A_t,\epsilon_1,\frac{1}{4}$) for once, and sample a_t for n_2 times. Each call of Median-Elimination takes at most $O(\frac{n_1}{\epsilon^2})=O(\frac{1}{\rho\epsilon^2})$ samples [16], and $n_2=O(\frac{1}{\epsilon^2}\log\frac{k}{\delta})$. Thus, each repetition takes $O(\frac{1}{\epsilon^2}(\frac{1}{\rho}+\log\frac{k}{\delta}))$ samples. By (24), in each repetition, with probability at least $\frac{1}{4}$, one arm is added to Ans, and the algorithm terminates after k arms are added to Ans. Obviously, after at most $4k$ repetitions in expectation, the algorithm returns. Thus, the expected sample complexity is $O(\frac{k}{\epsilon^2}(\frac{1}{\rho}+\log\frac{k}{\delta}))$. This completes the proof. □

3 PROOF OF LEMMA 6

Proof. Let a_r be the returned arm. For arm a, define

$$\mathcal{E}_a^N:=\{\exists t,N^t(a)=N,\mu_a<L^t(a)\vee\mu_a>L^t(a)\},\tag{28}$$

i.e., the event that when $N^t(a)=N$, μ_a is not within the interval $[L^t(a),U^t(a)]$. Define the bad event $\mathcal{E}_{out}:=\bigcup_{a,N}\mathcal{E}_a^N$. By (4) and (5), we have that

$$\mathbb{P}\left\{\mathcal{E}_a^N\right\}\leq2\delta^N.\tag{29}$$

Thus, by $k_1\geq2\sum_t t^\gamma$ and the union bound, we have that

$$\mathbb{P}\{\mathcal{E}_{out}\}\leq\sum_{a,N}\mathbb{P}\left\{\mathcal{E}_a^N\right\}\leq n\sum_{N=1}^{\infty}2\delta^N\leq\delta.\tag{30}$$

Since *budget* is large enough, when returning, $B(t) \leq \epsilon$. Let t_0 be the time when the algorithm returns. We have that for all $a \neq a_r$, $U^{t_0}(a) \leq L^{t_0}(a_r) + \epsilon$. By the definition of $\mathcal{E}_{out}$, when it does not happen, for all arms a, $\mu_a \in [L^t(a), U^t(a)]$ for all t, implying that

$$\mu_a \leq U^{t_0}(a) \leq L^{t_0}(a_r) + \epsilon \leq \mu_{a_r} + \epsilon. \tag{31}$$

Thus, the returned arm a_r is $(\epsilon, 1)$-optimal with probability at least $1 - \delta$. □

4 PROOF OF LEMMA 7

Proof. In the proof, we assume $\mathcal{E}_{out}$ does not happen. This event is defined in the proof of Lemma 6, and does not happen with probability at least $1 - \delta$.

Let τ be the number of samples taken till termination. Define the set $T := \{n + 2i : i \in \mathbb{N}, n + 2i < \tau\}$. T is the set of t such that a^t and b^t are computed. For each arm a, define $X_a := \sum_{t \in T} \mathbb{1}_{b^t = a}$, the number of times that b^t is a. Define $\mu^* := \max_{a \in A} \mu_a$, $\Delta'_a := \mu^* - \mu_a$, and $\Delta_a := \frac{1}{2} \max\{\epsilon, \Delta'_a\}$. Now, we are going to bound X_a.

Let a be an arbitrary arm in A. Assume that at some time $t \in T$,

$$N^t(a) \geq \frac{1}{\Delta_a^2} \max\left\{\log \frac{k_1 n}{\delta}, (\gamma + \frac{\gamma}{e}) \log \frac{(\gamma + \frac{\gamma}{e})}{\Delta_a^2}\right\}, \tag{32}$$

and we will show that either b^t does not equal to a or the algorithm returns before the next sample.

Let $x = \frac{\gamma}{\Delta_a^2}$ ($x > 4$ as $\Delta_a \leq \frac{1}{2}$ and $\gamma > 1$). Since $N^t(a) \geq (1 + e^{-1}) x \log((1 + e^{-1})x) > 4$, we have that

$$\begin{aligned} \frac{N^t(a)}{\log N^t(a)} &\overset{(i)}{>} \frac{(1 + e^{-1}) x \log((1 + e^{-1})x)}{\log((1 + e^{-1})x) + \log\log((1 + e^{-1})x)} \\ &= \frac{(1 + e^{-1}) x}{1 + \frac{\log\log((1+e^{-1})x)}{\log((1+e^{-1})x)}} \overset{(ii)}{\geq} x, \end{aligned} \tag{33}$$

where (i) is because $\frac{y}{\log y}$ is increasing for $y \geq e$, and (ii) is because $\frac{\log y}{y} \leq \frac{1}{e}$. It implies that

$$\frac{1}{2} N^t(a) \geq \frac{\gamma}{2\Delta_a^2} \log N^t(a) \tag{34}$$

Also, by (32) we have that

$$\frac{1}{2} N^t(a) \geq \frac{1}{2\Delta_a^2} \log \frac{k_1 n}{\delta}. \tag{35}$$

Thus, adding (34) and (35), we have that

$$N^t(a) > \frac{1}{2\Delta_a^2} \log \frac{k_1 n (N^t(a))^\gamma}{\delta}. \tag{36}$$

It follows that

$$\sqrt{\frac{1}{2N^t(a)} \log \frac{k_1 n (N^t(a))^\gamma}{\delta}} < \Delta_a \tag{37}$$

Recall that in the algorithm, for arm a, we define $U^t(a) := u(\hat{\mu}^t(a), N^t(a), \delta^{N^t(a)})$ and $L^t(a) := l(\hat{\mu}^t(a), N^t(a), \delta^{N^t(a)})$ as ((6) and (7)). By the choice of $\delta^{N^t(a)} = \frac{\delta}{k_1 n (N^t(a))^\gamma}$ in PACMaxing, and the choice of confidence bounds, we have that

$$U^t(a) - \hat{\mu}^t(a) = \hat{\mu}^t(a) - L^t(a) < \Delta_a, \tag{38}$$

$$U^t(a) - L^t(a) \leq 2\Delta_a. \tag{39}$$

Now, for this a, we will show that either the algorithm returns before next sample or $b^t \neq a$.

First we consider the case where $\Delta_a = \frac{\epsilon}{2}$. Here we assume that $\mathcal{E}_{out}$ does not happen. This means for any t and arm $b \in A$, μ_b is in $[L^t(b), U^t(b)]$. Since $b^t = a$ and $b^t := \arg\max_{b \in A} U^t(b)$, for all arms $b \neq a$, $U^t(a) \geq U^t(b)$. By (39), $L^t(a) \geq U^t(a) - \epsilon \geq U^t(b) - \epsilon$. This means that the algorithm returns arm a before the next sample as we have $B(t) \leq \epsilon$.

Next, we consider the case where $\Delta_a = \frac{\Delta'_a}{2}$. Let a^* be the most rewarding arm in A. Since $\mathcal{E}_{out}$ does not happen, by the definition of $\mathcal{E}_{out}$ and (39), we have $U^t(a) < L^t(a) + \Delta'_a \leq \mu_a + \Delta'_a \leq \mu^* \leq U^t(a)$, implying $b^t \neq a$. This leads to a contradiction.

Thus, we can conclude that when $\mathcal{E}_{out}$ does not happen,

$$X_a \leq 1 + \frac{1}{\Delta_a^2} \max\left\{\log \frac{k_1 n}{\delta}, (\gamma + \frac{\gamma}{e}) \log \frac{(\gamma + \frac{\gamma}{e})}{\Delta_a^2}\right\}. \tag{40}$$

Except the first n samples, there is one b^t sampled out of every two consecutive samples. Thus, with probability at least $1 - \delta$, the number of samples taken before termination is at most

$$\begin{aligned} & n + 2 \sum_{a \in A} X_a \\ \leq & 3n + \sum_{a \in A} \frac{2}{\Delta_a^2} \max\left\{\log \frac{k_1 n}{\delta}, (\gamma + \frac{\gamma}{e}) \log \frac{(\gamma + \frac{\gamma}{e})}{\Delta_a^2}\right\} \end{aligned} \tag{41}$$

The desired sample complexity follows.

Since $\Delta_a \leq \frac{\epsilon}{2}$, the *budget* value stated in this lemma is no less than that in (41). This completes the proof. □

5 PROOF OF LEMMA 9

The first step is to prove that with probability at least $1 - \frac{2\delta}{5}$, the m-th most rewarding arm of A_1 is in $M :=$

$\{a \in \mathcal{S} : \lambda_\rho \le \mu_a \le \lambda_{\rho/2}\}$. Here we note that $m := \lfloor \frac{3}{4}\rho n_3 \rfloor$ as defined in LambdaEstimation. To do it, we need to introduce an inequality directly derived from Chernoff Bound. Let $X^1, X^2, ..., X^t$ be t independent Bernoulli random variables, and for all i, $\mathbb{E}X^i \ge p$. Define $S := \sum_{i=1}^t X^i$. Let $B(t,p)$ denote a Binomial random variable with parameters t and p. For any $b \le tp$, we have $\mathbb{P}\{S \le b\} \le \mathbb{P}\{B(t,p) \le b\}$, and thus, by Chernoff Bound,

$$\mathbb{P}\{S \le b\} \le \exp\left\{-\frac{t}{2p}\left(p - \frac{b}{t}\right)^2\right\}. \tag{42}$$

Define $S_1 := \{a \in A_1 : \mu_a > \lambda_{\rho/2}\}$ and $S_2 := \{a \in A_1 : \mu_a \ge \lambda_\rho\}$. In this paper, we use $a \sim \mathcal{S}$ to denote that a is randomly drawn from $\mathcal{S}$. By (2) and (3), we have

$$\mathbb{P}_{a\sim\mathcal{S}}\{a \in S_1\} \le \frac{\rho}{2}, \tag{43}$$

$$\mathbb{P}_{a\sim\mathcal{S}}\{a \in S_2\} \ge \rho. \tag{44}$$

By the works of Arratia and Gordon [3], we have that for $x > tp$,

$$\mathbb{P}\left\{B(t,p) \ge x\right\} \le \exp\left\{-tD_{KL}\left(\frac{x}{t}||p\right)\right\}, \tag{45}$$

where $B(t,p)$ stands for a Binomial(t,p) random variable, and $D_{KL}(p||q) := p\log\frac{p}{q} + (1-p)\log\frac{1-p}{1-q}$. Thus, along with (43), we have that

$$\begin{aligned}
&\mathbb{P}\left\{|S_1| \ge \frac{3}{4}\rho n_3\right\} \le \exp\left\{-n_3 D_{KL}\left(\frac{3}{4}\rho||\frac{1}{2}\rho\right)\right\}\\
&= \exp\left\{-n_3\left[\frac{3}{4}\rho\log\frac{3}{2} - \left(-\frac{3}{4}\rho\right)\log\left(1 + \frac{\frac{\rho}{4}}{1-\frac{3\rho}{4}}\right)\right]\right\}\\
&\le \exp\left\{-n_3\rho\left[\frac{3}{4}\log\frac{3}{2} - \frac{1}{4}\right]\right\} \le \frac{\delta}{5},
\end{aligned} \tag{46}$$

Also, by (42), it holds that

$$\mathbb{P}\left\{|S_2| \le \frac{3}{4}\rho n_3\right\} \le \exp\left\{-\frac{n_3}{2\rho}\left(\frac{1}{4}\rho\right)^2\right\} \le \frac{\delta}{5}. \tag{47}$$

The above two statement (46) and (47) implies that with probability at least $1-\frac{2\delta}{5}$, $|S_1| < \frac{3}{4}\rho n_3$ and $|S_2| > \frac{3}{4}\rho n_3$. Recalling that $m = \lfloor \frac{3}{4}\rho n_3 + 1\rfloor$, the m-th most rewarding arm of A_1 is in M with probability at least $1 - \frac{2\delta}{5}$.

The second step is to prove that $\mu_{\hat{a}}$ is in $[\lambda_\rho - \epsilon_1, \lambda_{\rho/2} + \epsilon_2]$ with probability at least $1 - \frac{4\delta}{5}$. The call of Halving$(A_1, m, \epsilon_1, \frac{\delta}{5})$ returns an m-sized set of arms A_2, and with probability at least $1 - \frac{\delta}{5}$, every arm a in it has $\mu_a \ge \lambda'_{[m]} - \epsilon_1$, where $\lambda'_{[m]}$ is the m-th most rewarding arm in A_1 [20]. We note that with probability at least $1 - \frac{2\delta}{5}$, the m-th most rewarding arm of A_1 is in M, implying $\lambda'_{[m]} \ge \lambda_\rho$. Thus, we have that

$$\mathbb{P}\left\{A_2 \subset E_{\epsilon_1}\middle| |S_1| < \frac{3}{4}\rho n_3 < |S_2|\right\} \ge 1 - \frac{\delta}{5} \tag{48}$$

Besides, by (46) and $|A_2| = m \ge \frac{3}{4}\rho n_3$, at least one arm a^w of A_2 is in M (i.e., $\mu_{a^w} \le \lambda_{\rho/2}$) if $|S_1| < \frac{3}{4}\rho n_3$. The call of Halving$_2(A_3, 1, \epsilon_2, \frac{\delta}{5})$ returns an arm $\hat{a}$ of A_2 having $\mu_{\hat{a}} \le \mu_{a^w} + \epsilon_2 \le \lambda_{\rho/2} + \epsilon_2$ with probability at least $1 - \frac{\delta}{5}$ [20] if $|S_1| < \frac{3}{4}\rho n_3$, i.e.,

$$\mathbb{P}\left\{\mu_{\hat{a}} \le \lambda_{\rho/2} + \epsilon_2\middle| |S_1| < \frac{3}{4}\rho n_3\right\} \ge 1 - \frac{\delta}{5}. \tag{49}$$

It follows from $\hat{a} \in A_2$, the definition of E_{ϵ_1}, (46), (47), (48), and (49) that

$$\mathbb{P}\left\{\mu_{\hat{a}} \in \left[\lambda_\rho - \epsilon_1, \lambda_{\rho/2} + \epsilon_2\right]\right\} \ge 1 - \frac{4\delta}{5}. \tag{50}$$

The third step is to prove that $\hat{\lambda}$ is in $[\lambda_\rho - \epsilon, \lambda_{\rho/2}]$ with probability at least $1 - \delta$. Since $\hat{a}$ is sampled for n_4 times, by (50) and Hoeffding's Inequality, we have

$$\begin{aligned}
&\mathbb{P}\left\{\hat{\lambda} \notin \left[\lambda_\rho - \epsilon, \lambda_{\frac{\rho}{2}}\right]\right\}\\
=&\mathbb{P}\left\{\hat{\mu} \notin \left[\lambda_\rho - \epsilon_1 - \epsilon_3, \lambda_{\frac{\rho}{2}} + \epsilon_2 + \epsilon_3\right]\right\}\\
\le&\mathbb{P}\left\{\mu_{\hat{a}} \notin \left[\lambda_\rho - \epsilon_1, \lambda_{\frac{\rho}{2}} + \epsilon_2\right]\right\} + \mathbb{P}\left\{|\hat{\mu} - \mu_{\hat{a}}| \ge \epsilon_3\right\}\\
\le&\frac{4\delta}{5} + 2\exp\left\{-2n_4\epsilon_3^2\right\} \le \frac{4\delta}{5} + \frac{\delta}{5} \le \delta.
\end{aligned} \tag{51}$$

This completes the proof of correctness.

It remains to prove the sample complexity. Line 4 uses $O(\frac{n_3}{\epsilon^2}\log\frac{m}{\delta}) = O(\frac{1}{\rho\epsilon^2}\log^2\frac{1}{\delta})$ samples [20], and Line 5 uses $O(\frac{m}{\epsilon^2}\log\frac{1}{\delta}) = O(\frac{1}{\epsilon^2}\log^2\frac{1}{\delta})$ samples. Line 6 takes $n_4 = O(\frac{1}{\epsilon^2}\log\frac{1}{\delta})$ samples. The desired results follows by summing these three upper bounds up. □

6 PROOF OF THEOREM 11

Proof. Each call of AL-Q-IK is wrong with probability at most $\frac{\delta}{k}$. The correctness follows.

By Theorem 5, the t-th repetition uses $O(\frac{1}{\epsilon^2}(\frac{n+1-t}{m+1-t} + \log\frac{k}{\delta}))$ samples in expectation. For all $x \in (0,1]$, we have $\frac{\log(1+x)}{x} \ge \log 2$. It implies

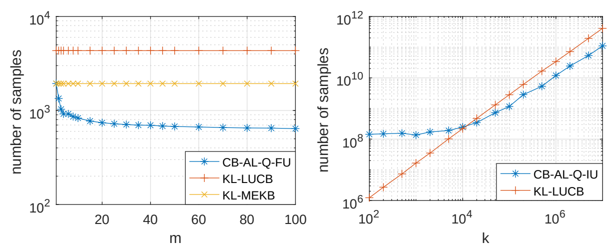
$$\log\frac{m+2-t}{m+1-t} \ge \frac{\log 2}{m+1-t}, \tag{52}$$

and thus,

$$
\begin{aligned}
&\sum_{t=1}^{k}\left\{\frac{1}{\epsilon^2}\left(\frac{n+1-t}{m+1-t}+\log\frac{k}{\delta}\right)\right\}\\
\leq&\frac{k}{\epsilon^2}\log\frac{k}{\delta}+\frac{n}{\epsilon^2\log 2}\sum_{t=1}^{k}\log\frac{m+2-t}{m+1-t}\\
\leq&\frac{k}{\epsilon^2}\log\frac{k}{\delta}+\frac{n}{\epsilon^2\log 2}\log\frac{m+1}{m+1-k}. \qquad (53)
\end{aligned}
$$

The sample complexity follows. □

7 ADDITIONAL NUMERICAL RESULTS



(a) Comparison of the finite-armed pure exploration algorithms. $n = 1000$, $k = 1$, $\rho = 0.001$,$\epsilon = 0.05$, and $\delta = 0.001$.

(b) Comparison of CB-AL-Q-IU and KL-LUCB under prior $\mathcal{F}_h$. $\rho = 0.05$,$\epsilon = 0.1$, and $\delta = 0.01$.

Figure 3: Additional Numerical Results.

First, we compare the pure exploration algorithms in the finite cases to demonstrate that by adopting the QE setting, the number of samples taken can be greatly reduced compared with the KE setting. Other comparisons on the finite-armed algorithms are omitted as their performance is similar to their infinite-armed versions, especially when n is large. Also, when $k = 1$, their performance are almost the same.

The algorithms compared include CB-AL-Q-FK (CBB version of AL-Q-FK by replacing the subroutines with CBB ones), KL-LUCB for the finite case [22], and MEKB [25]. Here we modify MEKB to the CBB version with the KL-Divergence confidence bounds given by Kaufmann and Kalyanakrishnan [22]. The results are summarized in Figure 3 (a). KL-LUCB and MEKB were designed to find one $(\epsilon, 1)$-optimal arm from a finite set. MEKB has the prior knowledge of $\lambda_{[1]}$, and can be regarded as the $m = 1$ version of AL-Q-FK. There are totally 1000 arms. For each arm, its rewards follow the Bernoulli distribution, and its expected reward is generated by taking an independent instance of the Uniform([0,1]) distribution. All algorithms are tested on the same dataset. Every point is averaged over 100 independent trials.

Here we note that the KE algorithms KL-LUCB and MEKB were designed to find an $(\epsilon, 1)$-optimal arm, so their performance are independent of m.

According to Figure 3 (a), the two algorithms CB-AL-Q-FK and KL-MEKB that have knowledge of $\lambda_{[m]}$ or $\lambda_{[1]}$ perform better than KL-LUCB, the one without the knowledge, consistent with the theory. When $m = 1$, the performance of CB-AL-Q-IK and KL-MEKB are close. However, when $m > 1$, CB-AL-Q-IK takes less samples, and the gaps increases as m. The reason lies in that (CB-)AL-Q-IK's sample complexity depends on $\frac{n}{m}$ while (KL-)MEKB's depends on n. Thus, the numerical results indicate that by adopting the QE setting, one can find "good" enough arms by much less samples.

Next, we compare CB-AL-Q-IU and (α, ϵ)-KL-LUCB. CB-AL-Q-IU is the CBB version of AL-Q-IU by replacing its subroutines by CBB ones. (CB-)AL-Q-IU is designed for large k values, and it does not perform well under small k values, even if it is always in order-sense better or equivalent compared to KL-LUCB. The reason is that its subroutine (CB-)LambdaEstimation has a large constant factor. However, since the sample complexities of these two algorithms both depend at least linearly on k while that of (CB-)LambdaEstimation is independent of k, when k is large, the influence of (CB-)LambdaEstimation vanishes, and the improvement of (CB-)AL-Q-IK emerges. The results are summarized in Figure 3 (b). In Figure 3 (b), the algorithms are tested under a "hard instance" $\mathcal{F}_h$, where ρ fraction of the arms has expected reward $\frac{1}{2} + 0.55\epsilon$ and the others have $\frac{1}{2} - 0.55\epsilon$. The results are consistent with the theory, and suggest that CB-AL-Q-IK can use much less samples than KL-LUCB when k is sufficiently large.

We admit that AL-Q-IU may not be practical as it takes 10^8 samples even for $k = 1$, but it also has several contributions. (I) It gives a hint for solving the Q-IU problem. If we can improve LambdaEstimation, we can get a practical algorithm for the Q-IU problem that works much better than the literature for large k values. (II) We can see from Figure 3 (b), KL-LUCB increases faster as k. It is consistent with the theory that KL-LUCB depends on $k\log^2 k$ while (CB-)AL-Q-IU depends on $k\log k$. When k is extremely large (though may not be practical), (CB)-AL-Q-IU can be much better. (III) In order sense, the performance of (CB-)AL-Q-IU is better than the literature. Thus, our work gives better theoretical insights about the Q-IU problem.