# Rate-optimal Design for Anytime Best Arm Identification

**Junpei Komiyama**
MBZUAI / NYU / RIKEN AIP

**Kyoungseok Jang**
Chung-Ang University

**Junya Honda**
Kyoto University / RIKEN AIP

## Abstract

We consider the best arm identification problem, where the goal is to identify the arm with the highest mean reward from a set of $K$ arms under a limited sampling budget. This problem models many practical scenarios such as A/B testing. We consider a class of algorithms for this problem, which is provably minimax optimal up to a constant factor. This idea is a generalization of existing works in fixed-budget best arm identification, which are limited to a particular choice of risk measures. Based on the framework, we propose Almost Tracking, a closed-form algorithm that has a provable guarantee on the popular risk measure. Unlike existing algorithms, Almost Tracking does not require the total budget in advance nor does it need to discard a significant part of samples, which gives a practical advantage. Through experiments on synthetic and real-world datasets, we show that our algorithm outperforms existing anytime algorithms as well as fixed-budget algorithms. Our recommended algorithm for practitioners is found in the final section.

## 1 INTRODUCTION

In the Best Arm Identification (BAI) problem, a learner sequentially samples from $K$ arms with unknown means $P_1, \ldots, P_K$ and wishes to output the arm with the largest mean. This classical pure exploration task underpins applications in A/B testing (Li et al., 2010; Johari et al., 2017; Xiong et al., 2024), clinical trials (Villar et al., 2015; Aziz et al., 2021; Wang and Tiwari, 2023), and adaptive control (Koval et al., 2015; Paudel and Stein, 2023). The fixed-budget setting, which aims to maximize identification accuracy after exactly $T$ samples, captures scenarios where evaluation may be stopped externally (traffic shifts, budget exhaustion). Formally, an anytime algorithm consists of

- Sampling strategy, which selects the *sampling arm* $I(t)$ given the history up to round $t-1$.
- Recommendation strategy, which selects the *recommendation arm* $J(t)$, which is the estimate of the best arm given the history up to round $t$.

Popular fixed-budget algorithms (SR (Audibert et al., 2010), SH (Karnin et al., 2013)) rely on elimination schedules that presuppose $T$; anytime adaptations via doubling (Zhao et al., 2023) sacrifice statistical efficiency by discarding earlier samples. We propose an alternative, Almost Tracking, that (i) never requires the horizon $T$, (ii) avoids elimination, and (iii) retains all observations while (iv) has provable minimax rates as described below.

A natural performance measure of a BAI algorithm is the probability of error $\mathrm{PoE}(\boldsymbol{P}) := \mathbb{P}(J(T) \neq i^*(\boldsymbol{P}))$, which is the probability that the recommendation arm $J(T)$ does not match the true best arm[1] $i^*(\boldsymbol{P}) = \arg\max_i P_i$ when the (unknown) true model is $\boldsymbol{P}$. The smaller $\mathrm{PoE}(\boldsymbol{P})$ is, the better the algorithm is. On this performance measure, it is generally important to consider the minimax optimality. This is because the performance of an algorithm typically involves a trade-off across different problem instances.[2] Specifically, an algorithm that performs exceptionally well for one problem instance $\boldsymbol{P}$ may perform poorly for another problem instance $\boldsymbol{P'}$ (Wang et al., 2024). Consequently, in fixed-budget BAI, there is no universal notion of optimality. In view of this, minimax optimality is the following objective: Given a risk measure, minimize the risk of the algorithm in the worst-case instance.

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300. Copyright 2026 by the author(s).

[1] In this paper, we identify the mean of the distribution $P_i$ with the distribution itself, and assume the uniqueness of $i^*(\boldsymbol{P})$.

[2] This contrasts with uniform optimality in the fixed-confidence setting; see Section 1.3 for a comparison.

In fact, existing line of works (Audibert et al., 2010; Karnin et al., 2013; Carpentier and Locatelli, 2016; Zhao et al., 2023) can be viewed as special classes of minimax optimal algorithms with a particular choice of risk measure (Section 1.3).

In this context, Komiyama et al. (2022) derived a fundamental lower bound for any given risk measure, which implies that the optimal exploration allocation–i.e., the sampling proportions across arms–should be a function of the empirical means. However, establishing the corresponding theoretical rates for such an adaptive allocation is generally challenging due to the trackability issues discussed below, and hence most existing algorithms resort to less flexible exploration schedules than the theoretical optimum. A notable exception is the minimax-optimal algorithm called Delayed Optimal Tracking (DOT, Komiyama et al. 2022). However, DOT is computationally intractable. It needs to solve a dynamic programming, and each step of the dynamic programming involves a non-convex optimization over $K$ functions.[3] We desire a more practical algorithm that is easy to compute and implement.

**Summary of Our Results:**

- We establish a constant-factor approximation to a one-shot game optimum that links adaptive allocations to the minimax rate (Section 2). Based on the one-shot game, we introduce Simple Tracking.
- However, analysis of Simple Tracking is very challenging. We further propose Almost Tracking, a closed-form, anytime sampling rule that provably attains minimax rate optimality up to a constant-factor without knowing $T$, without discarding samples, and without any horizon-calibrated hyperparameter (Section 3).
- Almost Tracking requires a constant-factor approximation of the optimal allocation function. For the canonical $H_1$ risk measure (Audibert et al., 2010), we derived a closed-form allocation formula (Section 4).
- In our synthetic benchmarks and real datasets (OBD, MovieLens), Almost Tracking outperforms anytime (DSH/DSR) and fixed-budget baselines (SR/SH/CR), despite using neither the budget $T$ nor $T$-dependent tuning (Section 5).
- As a byproduct, we derive a tight characterization of Successive Rejects by establishing its exact error exponent (Section 6).

[3]See Section J for the non-convexity of the objective.

After these sections, we conclude the paper (Section 7). Our recommended algorithm for practitioners is found in the following section (Section 8).

### 1.1 Minimax Optimality

The raw worst-case PoE is not very meaningful since the identification of the best arm can be arbitrarily hard when the gap between the best arm and the other arms is small. A reasonable target to optimize is the minimax rate (Komiyama et al., 2022; Degenne, 2023) of the risk. Let $H(\boldsymbol{P}) : \mathcal{P}^K \to \mathbb{R}^+ \cup \{\infty\}$ be a risk measure, where $\mathcal{P}$ is the set of distributions. We then discuss the best possible *rate* $R > 0$ such that

$$\mathrm{PoE}(\boldsymbol{P}) \le \exp\left(-\frac{RT}{H(\boldsymbol{P})} + o(T)\right) \text{ for all } \boldsymbol{P}.$$

**Definition 1** (Rate)**.** We use $\mathrm{PoE}_{\mathcal{A}}(\boldsymbol{P})$ to represent the probability of error when we use algorithm $\mathcal{A}$. The (normalized) rate of algorithm $\mathcal{A}$ is defined as

$$R_{\mathcal{A}} := \inf_{\boldsymbol{P}\in\mathbb{R}^K : |i^*(\boldsymbol{P})|=1} H(\boldsymbol{P}) \liminf_{T\to\infty} \frac{1}{T} \log(1/\mathrm{PoE}_{\mathcal{A}}(\boldsymbol{P})).$$

Thus $R_{\mathcal{A}}$ is the largest uniform constant in the exponent after normalizing by instance complexity $H(P)$. Since it takes the minimum over all instances of $K$ arms, the rate $R_{\mathcal{A}}$ depends on $K$. For the most widely-studied risk measure $H_1(\boldsymbol{P})$ (Section 4), the best algorithm has $R_{\mathcal{A}} = \Theta(1/(\log K))$.

**Definition 2.** (Rate-optimality) An algorithm is *rate-optimal* (or minimax optimal up to a constant factor) if there exists a universal constant[4] $C > 0$ such that, for any $K$ and for any other algorithm $\mathcal{A}^*$, $R_{\mathcal{A}} \ge C R_{\mathcal{A}^*}$ holds.

### 1.2 Large Deviation

Let $n_i$ be the number of samples on arm $i$. It is known from large deviation theory (Dembo and Zeitouni, 2010) that the empirical mean $Q_i$ is approximately equal to $Q$ with probability roughly given by

$$\mathbb{P}[Q_i \approx Q] = \mathrm{poly}(n_i) \exp\left(-n_i D(Q\|P_i)\right),$$

where $\mathrm{poly}(n_i)$ is a polynomial factor and $D(\cdot\|\cdot)$ is the Kullback-Leibler (KL) divergence. It applies to a broad class of distributions, including members of the exponential family such as Bernoulli, Multinomial, and Gaussian distributions. Accordingly, our results in Section 2 extend to any distribution class that admits such inequalities. From Section 3, we focus on Gaussian rewards[5], where the KL divergence simplifies to

[4]A constant is universal if it is independent of model parameters, such as $K$ and $\mathcal{P}$.

[5]Throughout our analysis, we assume unit variance ($\sigma = 1$). The algorithm itself is variance-agnostic; only the error rate scales with $\sigma^2$.

$D(Q_i \| P_i) = (Q_i - P_i)^2/2$. Since the Gaussian large deviation bound can also be used as an upper bound for any sub-Gaussian distribution[6], our algorithms apply to the class of sub-Gaussian rewards. Appendix K describes limitations of each theorem.

### 1.3 Related Work

The study of identifying the best arm began with seminal work (Bechhofer, 1954) on ranking-and-selection in the 1950s. This line of research evolved into what became known as ordinal optimization (Chen et al., 2000; Glynn and Juneja, 2004) and later influenced the works called the best arm identification (BAI, Audibert et al. (2010)).

**Fixed-confidence BAI** In the fixed-confidence setting (Jamieson and Nowak, 2014), the learning agent decides whether to stop additional sampling or continue sampling for more information. The objective is to minimize the expected stopping time with a constraint that PoE at the end of the experiment should be smaller than a given confidence level $\delta \in (0,1)$. Two classes of algorithms, called Track and Stop algorithms (Kaufmann et al., 2016; Degenne and Koolen, 2019) and Top-Two algorithms (Russo, 2016; Jourdan et al., 2022; Lee et al., 2023; Mukherjee and Tajer, 2023; You et al., 2023; Bandyopadhyay et al., 2024), are well-known. Both algorithms are uniformly optimal across all instances. However, any such uniformly optimal algorithm in FC-BAI has arbitrarily worse performance in the FB-BAI setting (Komiyama et al., 2022, Appendix C).

**Fixed-budget BAI** The first paper on the modern formulation of BAI (Audibert et al., 2010) considers the fixed-budget setting where one is interested in the performance of an algorithm with a fixed number of samples. They also propose the successive reject (SR) algorithm. From the perspective of minimax optimality, their analysis is based on two particular complexity measures $H_1$ and $H_2$. Karnin et al. (2013) introduced an algorithm called successive halving (SH) that has the same rate as SR up to a constant factor. Carpentier and Locatelli (2016) show that SR and SH are minimax optimal up to a constant factor for $H_1$. Shahrampour et al. (2017) show extensive empirical comparison of these algorithms. Recently, Wang et al. (2023) proposed a new algorithm called Continuous Rejection (CR) and reported that it outperforms SR and SH. Note that, all of these algorithms are elimination-based and requires knowing the budget $T$ in advance. Zhao et al. (2023) introduce a doubling trick to SH to make it anytime. SR and SH are popular, and the idea of gradually eliminating the candidates arms is extended into many pure exploration problems. To name a few, linear bandits (Soare et al., 2014; Tao et al., 2018; Yang and Tan, 2022), bilinear models (Jang et al., 2021), combinatorial bandits (Chen et al., 2014; Du et al., 2021), unimodal bandits (Yu and Mannor, 2011; Ghosh et al., 2024), nonparametric models (Barrier et al., 2023), contextual problems (Li et al., 2022), and robust and nonstationary problems (Yu et al., 2018; Gupta et al., 2021; Takemori et al., 2024). In this paper, we propose an algorithm that fundamentally diverges from existing elimination-based approaches and has the potential to influence the design of best arm identification algorithms across various extended settings, with the value of anytime and provable minimax optimality.

## 2 CHARACTERIZING OPTIMIZATION

We can bound the minimax optimal rate from above (that is, bound the PoE from below) by considering the following game between an agent and an adversary.

**Definition 3.** (One-shot game) Information structure and move order:

1. The agent *commits ex ante* to a measurable mapping $\boldsymbol{w} : \mathcal{Q}^K \to \Delta^{(K)}$ (no knowledge of the particular $(\boldsymbol{P}, \boldsymbol{Q})$ realization).
2. Nature selects a pair $(\boldsymbol{P}, \boldsymbol{Q})$ with $\boldsymbol{P} \in \mathcal{P}^K$ : $|i^*(\boldsymbol{P})| = 1$, $\boldsymbol{Q} \in \mathcal{Q}^K$.
3. The mapping is evaluated at $\boldsymbol{Q}$ producing allocation $\boldsymbol{w}(\boldsymbol{Q})$; the agent suffers loss $L(\boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{w}) = H(\boldsymbol{P}) \sum_i w_i(\boldsymbol{Q}) D(Q_i \| P_i)$ when $i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})$.

Here, $\Delta^{(K)} \subset [0,1]^K$ is the $(K-1)$-dimensional probability simplex, and $\mathcal{Q}, \mathcal{P}$ are the space of empirical means and true means[7]. The value of the game is $\inf_{\boldsymbol{P},\boldsymbol{Q}} L$ with the supremum understood as the agent choosing the mapping before seeing $(\boldsymbol{P}, \boldsymbol{Q})$. Here, $\boldsymbol{P}$ represents the true means, $\boldsymbol{Q}$ a hypothesized empirical means that would mislead identification (misidentification event $i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})$), and $D(Q_i \| P_i)$ encodes the large deviation cost of that deception. The mapping $\boldsymbol{w}(\boldsymbol{Q})$ corresponds to the allocation of samples to each arm. An agent models a BAI algorithm: For example, $w_i(\boldsymbol{Q})$ corresponds to the algorithm that selects arm $i$ for $T w_i(\boldsymbol{Q}(T))$ times when the empirical mean at the end of the round $T$ is $\boldsymbol{Q}(T)$.

[6] A random variable $X$ is sub-Gaussian if $\mathbb{E}\left[e^{(X-\mathbb{E}[X])t}\right] \le \exp\left(\frac{t^2}{2}\right)$ for all $t > 0$.

[7] Though $\mathcal{Q}$ and $\mathcal{P}$ are identical to $\mathbb{R}$ in this paper we use these notations to clarify that they are the sets of empirical means and true means.

**Algorithm 1** Simple Tracking

Input: Target weight function $\boldsymbol{w}(\boldsymbol{Q})$.
Draw each arm once.
**for** $t = K+1, 2, \ldots, T$ **do**
  Draw arm $I(t) = \arg\max_{i\in[K]} \{w_i(\boldsymbol{Q}(t-1)) - N_i(t-1)/(t-1)\}$, where $N_i(t-1)$ be the number of samples for arm $i$ up to round $t-1$.
**end for**
**return** $J(T)$ = empirical best arm at the end.

The rate of any algorithm can be bounded by the worst-case loss of this game:

**Theorem 1** (Theorem 1 in Komiyama et al. (2022))**.** Under any algorithm $\mathcal{A}$ it holds that

$$R_{\mathcal{A}} \leq \sup_{\boldsymbol{w}(\cdot):\mathcal{Q}^K \to \Delta^{(K)}} \inf_{\boldsymbol{Q},\boldsymbol{P}:\mathcal{I}^*(\boldsymbol{P})\notin\mathcal{I}^*(\boldsymbol{Q})} H(\boldsymbol{P}) \sum_{i\in[K]} w_i(\boldsymbol{Q}) D(Q_i \| P_i) =: R^{\mathrm{go}}. \quad (1)$$

While the one-shot "game-optimal" value $R^{\mathrm{go}}$ upper bounds any attainable rate, it is a priori unclear whether it is itself (even approximately) achievable by a practical algorithm. This is because the above game considers a setting which is favorable for the agent where the agent can determine the allocation $w(\boldsymbol{Q})$ of plays after observing the entire empirical distributions $\boldsymbol{Q}$, which is not possible in the actual BAI task.

One naive idea to achieve $R^{\mathrm{go}}$ is to greedily track the optimal allocation, that is, to track the solution of the characterizing optimization of Eq. (1), which we call simple tracking in Algorithm 1. However, deriving a performance guarantee for Algorithm 1 with its target function set to the solution $\boldsymbol{w}^*(\boldsymbol{Q})$ of the optimization problem in Theorem 1 is highly nontrivial. The major challenges are as follows.

**Trackability issue:** It is possible that the empirical distribution of the current round $\boldsymbol{Q}(t)$ drastically changes from that of the last round $\boldsymbol{Q}(t-1)$, and in this case, the allocation can be far from the optimal allocation $\boldsymbol{w}^*(\boldsymbol{Q}(t))$. Such an event occurs with exponentially small probability and becomes negligible in the fixed-confidence scenario where we just need to evaluate the expected number of samples. However, it is not the case in the fixed-budget setting because the error probability is also exponentially small and it is possible that the error due to tracking is not negligible.

**Optimization issue:** At each round, Algorithm 1 computes $\boldsymbol{w}^*(\boldsymbol{Q}(t-1))$. This optimization is non-convex in general and makes this algorithm computationally intractable. While one can precompute and store the mapping $\boldsymbol{w}^*(\cdot)$ before the game, doing so compromises the practical utility and the interpretability of the algorithm.

From the theoretical perspective, due to the trackability issue it is unclear how close $R^{\mathrm{go}}$ is to the optimal rate. Our first contribution is that $R^{\mathrm{go}}$ is close to the optimal one up to a factor of two.

**Theorem 2.** There exists an algorithm $\mathcal{A}$ such that

$$\frac{R^{\mathrm{go}} - \varepsilon}{2} \leq R_{\mathcal{A}} \quad (2)$$

for any $\varepsilon > 0$. Consequently, we have $R^{\mathrm{go}}/2 \leq R_{\mathcal{A}} \leq R^{\mathrm{go}}$.

From this result, we can see that algorithms achieving rate close to $R^{\mathrm{go}}$ are also close to the minimax optimal.

In the proof of Theorem 2 in Appendix B, we also give an explicit algorithm to achieve this bound. However, from the practical viewpoint, the algorithm used in the proof discards a significant amount of samples and its performance is not particularly promising unlike the Simple Tracking.

From these observations, we address the above two difficulties (tracking and optimization issues) of simple tracking to provably realize a practical and (nearly) minimax-optimal algorithm.

## 3 ADDRESSING TRACKABILITY ISSUE

In this section we propose a generic framework of algorithms to resolve the trackability issue. Henceforth, we focus on sub-Gaussian reward distributions, and we use squared distance instead of the KL divergence. Theorem 1 discusses the best possible error rate (in the sense of minimax optimality) for a stronger model of the agent that can determine the allocation $\boldsymbol{w} = \boldsymbol{w}(\boldsymbol{Q})$ depending on the empirical distribution $\boldsymbol{Q}$ as if he/she knows $\boldsymbol{Q}$ before allocating the samples. The actual agent does not have such knowledge and needs to track the ideal allocation $\boldsymbol{w}(\boldsymbol{Q})$ based on the current empirical distribution, whose tracking error can essentially affect the error rate.

To alleviate this difficulty, we introduce the following algorithmic and theoretical tricks.

- A batched algorithm assigning samples to each arm without being affected by tracking errors of other arms.
- Error probability analysis incorporating the variance of empirical distributions.

**Algorithm 2** Almost Tracking

Input: Target allocation function $\boldsymbol{w}(\boldsymbol{Q})$, Parameter $C_{\text{suf}} \in (0,1)$. Draw per batch $N$.
Draw each arm for $\lceil N/K \rceil^{\text{int}}$ times. Here, $\lceil \cdot \rceil^{\text{int}}$ is a rounding operator defined in Appendix C.
**for** $b = 2, 3, \dots$ **do**
  Calculate

$$\mathcal{K}_{\text{insuf}} = \left\{ i \colon \frac{1}{b-1} \sum_{b'=1}^{b-1} w_{b',i} \leq \frac{1}{C_{\text{suf}}} w_i(\bar{\boldsymbol{Q}}_{b-1}) \right\} \tag{3}$$

$$s_{\text{insuf}} = \sum_{i \in \mathcal{K}_{\text{insuf}}} w_i(\bar{\boldsymbol{Q}}_{b-1}). \tag{4}$$

  Draw each arm for $\lceil w_{b,i} N \rceil^{\text{int}}$ times, where $w_{b,i} = 0$ for $i \notin \mathcal{K}_{\text{insuf}}$, and $w_{b,i} = w_i^*(\bar{\boldsymbol{Q}}_{b-1})/s_{\text{insuf}}$ for $i \in \mathcal{K}_{\text{insuf}}$.
**end for**
**return** $J^*(T)$ = empirical best arm at the end.

One issue of the simple tracking is that if there is an arm such that the current allocation is significantly smaller than the target allocation $w_i(\boldsymbol{Q})$ then all the allocation is assigned to this arm until the gap is mostly filled, which prohibits stably exploring other arms. To avoid this issue, we propose an algorithm Almost Tracking, which is formalized as Algorithm 2.

While Simple Tracking (Algorithm 1) myopically aims to follow the optimal allocation $w^*(\boldsymbol{Q}(t-1))$ by drawing the most insufficient arm, Almost Tracking (Algorithm 2) aims to follow the optimal allocation $w^*(\boldsymbol{Q}(t-1))$ but splits efforts into all insufficiently drawn arms in a batched manner. To be more specific, we compute the list $\mathcal{K}_{\text{insuf}}$ of insufficiently sampled arms and allocate fractions of

$$w_{b,i} = \begin{cases} w_i(\bar{\boldsymbol{Q}}_{b-1})/Z_b & \text{if } i \in \mathcal{K}_{\text{insuf}}, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where $\bar{\boldsymbol{Q}}_{b-1}$ is the empirical mean up to batch $b-1$, $Z_b = \sum_{i \in \mathcal{K}_{\text{insuf}}} w_i(\bar{\boldsymbol{Q}}_{b-1})$ is the normalization factor. The model parameter $C_{\text{suf}} \in (0,1)$ is supposed to be close to 1, in which case $\mathcal{K}_{\text{insuf}}$ just lists up all arms with its number of samples below the target allocation $w_i^*(\bar{Q}_{b-1})$.

Despite the structure of Almost Tracking, we cannot completely ensure the closeness between the final allocation and the ideal allocation $\boldsymbol{w}(\boldsymbol{Q})$ when $\bar{\boldsymbol{Q}}_b$ has drastically changed with batches. To evaluate the error due to this gap of the allocation, we incorporate the variance of the empirical means in the analysis. According to the large deviation principle, roughly speaking, if we allocate $N_{1,i}, N_{2,i}, \dots, N_{B,i}$ samples to arm $i$ in each batch, then the sequence of empirical means $(Q_{1,i}, Q_{2,i}, \dots, Q_{B,i})$ appears with probability

$$\begin{aligned} &\Pr[(\text{empirical distributions are } (Q_{1,i}, Q_{2,i}, \dots, Q_{B,i}))] \\ &\approx \exp\left( -\frac{1}{2} \sum_{b \in [B]} N_{b,i} (Q_{b,i} - P_i)^2 \right) \quad \text{(large deviation)} \end{aligned} \tag{6}$$

$$\begin{aligned} &\approx \underbrace{\exp\left( -\frac{1}{2} \left( \sum_{b \in [B]} N_{b,i} \right) (\bar{Q}_{B,i} - P_i)^2 \right)}_{\text{likelihood for overall empirical mean}} \\ &\quad \times \underbrace{\exp\left( -\frac{1}{2} \left( \sum_{b \in [B]} N_{b,i} (Q_{b,i} - \bar{Q}_{B,i})^2 \right) \right)}_{\text{likelihood for variance of empirical means}} \end{aligned} \tag{7}$$

by using the overall empirical distribution $\bar{Q}_{B,i}$ over all batches. This decomposition means that, empirical means $Q_{b,i}$ drastically changing with $b$ is less likely to occur compared with the empirical means $\bar{Q}_{b,i}$ constant over batches. Our theoretical contribution is that error due to the failure of tracking for drastically changing $Q_{b,i}$ can be compensated by this variance term.

The following describes the performance guarantee for the proposed algorithm.

**Theorem 3.** (Trackability) Let $w_i(\boldsymbol{Q}) \geq w_{\min}$ holds for any $\boldsymbol{Q}, i$ for some value $w_{\min} = w_{\min}(K) > 0$ that only depends on $K$. Let $B := \lfloor T/N \rfloor \geq 2/w_{\min}$. Consider Algorithm 2 with $C_{\text{suf}} \in (0,1)$. There exists a universal constant $C_{\text{track}} > 0$ such that the following holds for any sequence of $\boldsymbol{Q}_1, \boldsymbol{Q}_2, \dots, \boldsymbol{Q}_B$:

$$\begin{aligned} &\frac{1}{B} \sum_{b \in [B]} \sum_{i \in [K]} w_{b,i} \frac{(Q_{b,i} - P_i)^2}{2} \\ &\geq C_{\text{track}} \inf_{\boldsymbol{Q} : i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})} \left( \sum_{i \in [K]} w_i(\boldsymbol{Q}) (Q_i - P_i)^2 / 2 \right). \end{aligned} \tag{8}$$

By using $N_{b,i} = (T/B) w_{b,i}$ and decomposition of Eq. (7), Theorem 3 lower-bounds Eq. (6) by a worst-case single mean $\boldsymbol{Q}$. This bridges the gap between the one-shot game (Definition 3) and the inherently sequential nature of the best arm identification problem.

Theorem 3 implies that, if we have a constant-factor approximation allocation, we can achieve rate-optimality.

**Definition 4.** (constant-factor approximation) An allocation $\boldsymbol{w}(\cdot)$ is a constant-factor approximation if

there exists a universal constant $C_{\mathrm{stbl}} \in (0,1)$ independent of $K$ such that

$$\inf_{\boldsymbol{Q},\boldsymbol{P}:i^*(\boldsymbol{P})\notin i^*(\boldsymbol{Q})} H_1(\boldsymbol{P}) \sum_{i\in[K]} w_i(\boldsymbol{Q}) \frac{(Q_i - P_i)^2}{2} \geq C_{\mathrm{stbl}} R^{\mathrm{go}}. \quad (9)$$

**Theorem 4.** (Rate-optimality of Algorithm 2) Let $w_i(\boldsymbol{Q}) \geq w_{\min}$ holds for any $\boldsymbol{Q}, i$ for some $w_{\min} > 0$. Let $B := \lfloor T/N \rfloor \geq 2/w_{\min}$ and $N = \Omega(K(\log K)^3)$. Then, for a sufficiently large $K$, Algorithm 2 with $C_{\mathrm{suf}} \in (0,1)$ with a constant-factor approximation allocation $\boldsymbol{w}$ is rate-optimal.

*Proof of Theorem 4.* Due to page limitation, the lemmas are deferred to the appendix. For any $\boldsymbol{P}$ with $|i^*(\boldsymbol{P})| = 1$ and any $\boldsymbol{Q} : i^*(\boldsymbol{Q}) \neq i^*(\boldsymbol{P})$, it always holds that

$$\sum_{b\in[B]} \sum_{i\in[K]} N_{b,i} \frac{(Q_{b,i} - P_i)^2}{2} \quad (10)$$

$$\geq C_{\mathrm{track}} \inf_{\boldsymbol{Q}} \sum_{i\in[K]} w_i(\boldsymbol{Q}) \frac{(Q_i - P_i)^2}{2} T \quad \text{(by Theorem 3)}$$

$$\geq \frac{C R^{\mathrm{go}} T}{H_1(\boldsymbol{P})}. \quad \text{(by definition of constant-factor approximation)} \quad (11)$$

for $C = C_{\mathrm{track}} C_{\mathrm{stbl}} > 0$. Therefore, we have

$$\mathbb{P}\left[i^*(\boldsymbol{Q}) \neq i^*(\boldsymbol{P})\right] \quad (12)$$

$$\leq \mathbb{P}\left[\sum_{b\in[B]} \sum_{i\in[K]} N_{b,i} \frac{(Q_{b,i} - P_i)^2}{2} \geq \frac{C R^{\mathrm{go}} T}{H_1(\boldsymbol{P})}\right] \quad \text{(by Eq. (11))}$$

$$\leq \exp\left(-C\left(1 - O\left(\frac{1}{\log K}\right)\right) \frac{R^{\mathrm{go}} T}{H_1(\boldsymbol{P})} + o(T)\right). \quad \text{(by Lemma 16, which uses large deviation)} \quad (13)$$

□

**Remark 1.** (Finite-time property) While our rate is asymptotic, the proof itself does not fundamentally rely on asymptotics. By carefully tracing the argument, one could in fact derive a finite-time bound. Namely, it is possible to express the $o(T)$ term in Eq. (13) explicitly as a function of the model parameters. We do not pursue this direction here, however, since the resulting expression would be unnecessarily cumbersome.

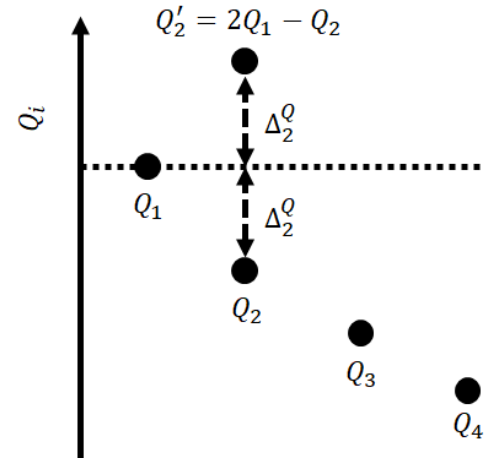



Figure 1: Hypothetical $\boldsymbol{Q}'$ such that $H_1'(i, \boldsymbol{Q}) := H_1(\boldsymbol{Q}')$ with $i = 2$. $Q_i' = Q_i$ for $i \neq 2$.

## 4 ADDRESSING OPTIMIZATION ISSUE

Theorem 4 ensures the rate-optimality of Algorithm 2 with a constant-factor approximation. However there remains a challenge to find a good allocation function $\boldsymbol{w}(\boldsymbol{Q})$, since its optimization (Eq. (1)) is non-convex and thus computationally costly.

As discussed in the introduction, the minimax optimality and corresponding allocation depend on the complexity measure. In this section, we focus on one of the most standard complexity measure $H_1(\boldsymbol{P}) = \sum_{i\neq i^*(\boldsymbol{P})} (P^* - P_i)^{-2}$. In particular, let

$$w_i(\boldsymbol{Q}) := \frac{1}{D_i(\boldsymbol{Q}) Z(\boldsymbol{Q})}, \quad (14)$$

where, for $\Delta_i^Q = \max_j Q_j - Q_i = Q^* - Q_i$,

$$H_1'(i, \boldsymbol{Q}) = \sum_{j\neq i} \frac{1}{(\Delta_j^Q + \Delta_i^Q)^2}, \quad (15)$$

$$D_i(\boldsymbol{Q}) = \begin{cases} (\Delta_i^Q)^2 H_1'(i, \boldsymbol{Q}) & (i \notin i^*(\boldsymbol{Q})) \\ \min_{i\notin i^*(\boldsymbol{Q})} D_i(\boldsymbol{Q}) & (i \in i^*(\boldsymbol{Q})) \end{cases}, \quad (16)$$

and $Z(\boldsymbol{Q}) = \sum_{i\in[K]} (1/D_i(\boldsymbol{Q}))$ is the normalization factor. $H_1'(i, \boldsymbol{Q})$ corresponds to $H_1(\boldsymbol{Q}')$ for hypothetical distribution $\boldsymbol{Q}'$ such that $Q_i$ is replaced with $2Q_1 - Q_i$ as illustrated in Figure 1.

This choice of $\boldsymbol{w}$ is inspired by Carpentier and Locatelli (2016), where $\boldsymbol{Q}'$ in Figure 1 is used to construct a lower bound for PoE with respect to $H_1$. We designed $\boldsymbol{w}$ to be inverse proportional to the cost of raising arm $i$ to be optimal arm ($= (\Delta_i^Q)^2$), normalized by the sample complexity when arm $i$ becomes optimal ($= H_1(\boldsymbol{Q}')$).

**Theorem 5.** Allocation of Eq. (14) is a constant-factor approximation (Definition 4).

This theorem implies the rate-optimality with respect to the risk measure $H_1$.

**Theorem 6.** (Rate-optimal computationally efficient algorithm) Assume that $N = \omega(K), T = \omega(NK \log K)$. Then, Algorithm 2 with $\boldsymbol{w}$ defined in Eq. (14) is rate-optimal for $H_1(\boldsymbol{P})$.

Theorem 6 follows from Theorem 5 combined with Theorem 4 with $w_{\min} = \Theta((K \log K)^{-1})$.

# 5 EXPERIMENTS

This section reports the results of our simulations. Our implementation is available at `https://github.com/jkomiyama/fb_bai_publish`. We compared the following algorithms:

- Uniform algorithm that draws arms in a round robin.
- Fixed-budget algorithms (require $T$): SR (Audibert et al., 2010), SH (Karnin et al., 2013), and two versions of CR (CR-A and CR-C) (Wang et al., 2023).
- Anytime algorithms (do not require $T$): DSH (Zhao et al., 2023) and Double Successive Rejects (DSR) that adopt the doubling trick. Our Simple Tracking (Algorithm 1), and Almost Tracking (Algorithm 2).
- Fixed-confidence algorithms: EB-TC and TS-TC (Shang et al., 2020; Jourdan et al., 2022), two empirically good versions of top-two Thompson sampling (Russo, 2020), and EB-TC$_{\varepsilon_0}$ (Jourdan et al., 2023). These algorithms are suboptimal in the fixed-budget setting.

Details of the algorithms are in Appendix L.1.

## 5.1 Simulation Results: Minimax Rates

We consider ten different instances of $\boldsymbol{P}$ with $K = 40$ arms with Gaussian rewards with a unit variance. Our goal is to maximize the performance of algorithms in the most challenging environment. The instances are derived from Wang et al. (2023) and we added theoretically challenging instances for SH and Tracking. Details of the instances are in Appendix L.2. Table 1 shows the estimated minimax rates by algorithm, which we compute the worst-case rate over ten synthetic instances, where the rate for instance $\boldsymbol{P}$ is defined as $T^{-1}H_1(\boldsymbol{P}) \log(1/\mathrm{PoE}(T))$, which corresponds to the exponent of the probability of error normalized by the complexity $H_1(\boldsymbol{P})$. As we can see, our algorithms outperform all existing algorithms, including anytime algorithms (DSH and DSR) as well as CR, SR, and SH, with respect to the risk measure $H_1(\boldsymbol{P})$. Notably, our algorithms outperform those that require $T$, despite not having access to it. SH does not perform as well as SR because it discards the samples at the end of each batch. DSH and DSR further discard even more samples, which leads to worse performance. Table 1 also presents the rates under $H_2(\boldsymbol{P}) = \max_i i\Delta_{(i)}^{-2}$, the complexity measure that SR is specifically optimized for (see Section 6). On this measure, the gap between SR, CR and Tracking is very narrow. This is reasonable because SR is tailored to $H_2(\boldsymbol{P})$, and CR has some similarities to SR. We further elaborate on these measures in Section 6.

TS-TC, a fixed-confidence algorithm, performs well overall even though it is not optimized for the fixed-budget setting. However, we note that the rate of EB-TC and TS-TC with a large $T$ can be arbitrarily bad (see Section L.5 for such an instance).

## 5.2 Simulation Results: Real-world Datasets

We further create instance $\boldsymbol{P}$ based on two real-world datasets. The Open Bandit Dataset (Saito et al., 2021) is a real-world logged bandit dataset collected from a fashion e-commerce platform. The Movielens 1M dataset (Harper and Konstan, 2015) is a dataset that includes anonymous ratings of popular movies. Details of the datasets are in Appendix L.4.

Figure 2 shows the results of the algorithms on real-world datasets. Tracking and AlmostTracking outperform anytime algorithms (DSH and DSR) as well as non-anytime algorithms (SR, and SH). The only exception is CR, which performs comparably to our algorithms in the Movielens 1M dataset. CR does not perform well in Open Bandit Dataset, which we elaborate in the section L.6.

# 6 RATE OF SUCCESSIVE REJECTS AND COMPARISON WITH OUR ALGORITHMS

To compare our algorithm with SR, we show the following characterization of SR's performance.[8]

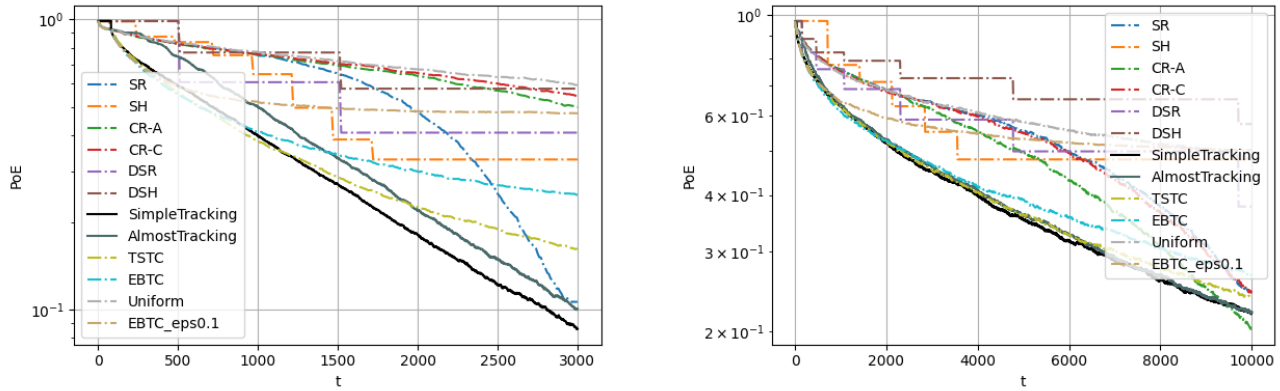
**Theorem 7.** For complexity measure $H_3(\boldsymbol{P})$ defined in Appendix G, PoE of SR satisfies

$$\lim_{T\to\infty} \frac{1}{T} H_3(\boldsymbol{P})\overline{\log}(K) \log(1/\mathrm{PoE}(T)) = 1.$$

[8]While Theorem 2 in Wang et al. (2023) derives the same upper bound, we complement this result by showing that it also serves as a lower bound.

Table 1: Estimated minimax rates of algorithms (larger better). Our algorithms, Simple tracking and Almost tracking are abbreviated as S.Track and A.Track, respectively. Left (resp. right) four algorithms are non-anytime (resp. anytime). Results with error bars (confidence region) are reported in Appendix L.3. Bold font highlights possible best with two-sigma confidence level. Results of Uniform and EB-TC$_{\varepsilon_0}$ are also in L.3.

| Metric | SR | SH | CR-A | CR-C | DSR | DSH | S.Track | A.Track | TS-TC | EB-TC |
|---|---|---|---|---|---|---|---|---|---|---|
| $H_1$ | 0.456 | 0.222 | 0.435 | 0.312 | 0.234 | 0.114 | **0.537** | **0.509** | 0.396 | 0.181 |
| $H_2$ | **0.256** | 0.106 | **0.271** | 0.194 | 0.131 | 0.056 | **0.260** | 0.255 | 0.227 | 0.104 |




(a) Open Bandit Dataset $K = 80$

(b) Movielens 1M Dataset $K = 31$

Figure 2: Comparison of PoE across algorithms on real-world datasets (smaller better). For algorithms that do not discard samples, $J(t)$ is the empirical best arm at time $t$. For discarding algorithms (SH, DSH, DSR), $J(t)$ is the empirical best arm at the end of the most recent batch.

Moreover, $(1/2)H_2(\boldsymbol{P}) \leq H_3(\boldsymbol{P}) \leq H_2(\boldsymbol{P})$ holds, where $H_2(\boldsymbol{P}) = \max_i i\Delta_{(i)}^{-2}$ is the complexity measure proposed in Audibert et al. (2010).

Theorem 7 tightly characterizes SR by providing matching upper and lower bounds, whereas Audibert et al. (2010) offered only an upper bound. This analysis enables us to fairly compare our algorithm with SR, eliminating the possibility of SR's analysis being loose.

**Performance Comparison Between SR and Almost Tracking** Theorem 1 in Carpentier and Locatelli (2016) implies that $R^{\mathrm{go}} = \Theta(1/(\log K))$. This fact, combined with our Theorem 6 implies our algorithm has PoE at most

$$\exp\left(-\frac{CT + o(T)}{(\log K)H_1(\boldsymbol{P})}\right) \tag{17}$$

for some universal constant $C > 0$, whereas Theorem 7 states that SR has PoE of

$$\exp\left(-\frac{T + o(T)}{(\overline{\log} K)H_3(\boldsymbol{P})}\right). \tag{18}$$

It holds that $H_3(\boldsymbol{P}) \leq H_1(\boldsymbol{P}) \leq H_3(\boldsymbol{P})(\log K)$, and thus Eq. (17) and Eq. (18) are identical up to a $\log K$ factor. Moreover, in the appendix, we show some examples where the error rate of Almost Tracking is strictly better than that of SR:

**Lemma 8.** (Instance where Almost Tracking outperforms SR) There exists an instance such that the rate of our algorithm is $O((\log K)/(\log\log K))$ times larger than that of SR.

**Lemma 9.** (Instance where SR may outperform Almost Tracking) There exists an instance such that the rate of SR is $O(\log K)$ times larger than that of Eq. (17).

Since Theorem 7 includes the performance lower bound of SR, Lemma 8 is tight—meaning that SR's performance is actually worse than Almost Tracking. On the other hand, Lemma 9 states that SR outperforms the rate of Eq. (17). Since Eq. (17) may not be tight, this result does not exclude the possibility that our algorithm performs as well as SR for all instances.

## 7 CONCLUSION

We have considered the fixed budget best arm identification problem. We have proposed algorithmic framework that achieves the rate-optimality in view of any given risk measure $H(\boldsymbol{P})$. Based on the framework,

we have proposed a closed-form algorithm that has rate-optimality for the standard risk measure $H_1(\boldsymbol{P})$. An interesting future work is to provide an efficient optimization algorithm to achieve the rate-optimality for a larger class of complexity measures. Unlike elimination-based methods, our algorithm does not require prior knowledge of $T$ and still outperforms both anytime and fixed-budget baselines, which motivates a redesign of algorithms for many structured pure exploration problems.

## 8 FOR PRACTITIONERS

Empirically, both Almost Tracking and Simple Tracking have strong performance across synthetic and real-world instances. When one desires the theoretically justified constant-factor guarantees with improved stability in adversarial instances, Almost Tracking is preferable. However, for most applied scenarios we recommend Simple Tracking because it is easy to implement and requires no horizon tuning. Practitioners can therefore start with Simple Tracking for $H_1$. For the sake of exposition, we provide an explicit expression of Simple Tracking for $H_1$ in Algorithm 3.

**Algorithm 3** Practical Recommendation: Simple Tracking for $H_1$

1: Draw each arm once.
2: **for** $t = K+1, K+2, \ldots, T$ **do**
3: Calculate number of draws $N_i(t-1)$ and empirical mean $Q_i(t-1)$ for each arm $i$.
4: Calculate weights

$$w_i(\boldsymbol{Q}(t-1)) := \frac{1}{D_i(\boldsymbol{Q})Z(\boldsymbol{Q})},$$

where

$$\Delta_i^Q = \max_j Q_j - Q_i \tag{19}$$

$$H_1'(i, \boldsymbol{Q}) = \sum_{j \neq i} \frac{1}{(\Delta_j^Q + \Delta_i^Q)^2}, \tag{20}$$

$$D_i(\boldsymbol{Q}) = \begin{cases} (\Delta_i^Q)^2 H_1'(i, \boldsymbol{Q}) & (i \notin i^*(\boldsymbol{Q})) \\ \min_{i \notin i^*(\boldsymbol{Q})} D_i(\boldsymbol{Q}) & (i \in i^*(\boldsymbol{Q})) \end{cases} \tag{21}$$

$$Z(\boldsymbol{Q}) = \sum_{i \in [K]} (1/D_i(\boldsymbol{Q})). \tag{22}$$

5: Draw arm $I(t) = \arg\max_{i \in [K]} \{w_i(\boldsymbol{Q}(t-1)) - N_i(t-1)/(t-1)\}$, where $N_i(t-1)$ be the number of samples for arm $i$ up to round $t-1$.
6: **end for**
7: **return** $J(T) =$ empirical best arm at the end.

## Acknowledgement

K. Jang was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [RS-2021-II211341, Artificial Intelligence Graduate School Program (Chung-Ang University)] and by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) [RS-2026-25499850]. J. Komiyama was supported by the MBZUAI Start-up Fund [BF0121, Machine Learning Department]. J. Honda was supported by JSPS KAKENHI (Grant Number JP25K03184).

### References

Audibert, J., Bubeck, S., and Munos, R. (2010). Best arm identification in multi-armed bandits. In *Conference on Learning Theory*, pages 41–53.

Aziz, M., Kaufmann, E., and Riviere, M.-K. (2021). On multi-armed bandit designs for dose-finding trials. *Journal of Machine Learning Research*, 22(14):1–38.

Bandyopadhyay, A., Juneja, S., and Agrawal, S. (2024). Optimal top-two method for best arm identification and fluid analysis. In Globersons, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J. M., and Zhang, C., editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Barrier, A., Garivier, A., and Stoltz, G. (2023). On best-arm identification with a fixed budget in non-parametric multi-armed bandits. In Agrawal, S. and Orabona, F., editors, *International Conference on Algorithmic Learning Theory, February 20-23, 2023, Singapore*, volume 201 of *Proceedings of Machine Learning Research*, pages 136–181. PMLR.

Bechhofer, R. E. (1954). A single-sample multiple decision procedure for ranking means of normal populations with known variances. *The Annals of Mathematical Statistics*, pages 16–39.

Carpentier, A. and Locatelli, A. (2016). Tight (lower) bounds for the fixed budget best arm identification bandit problem. In *Conference on Learning Theory*, pages 590–604.

Chen, C.-H., Lin, J., Yücesan, E., and Chick, S. E. (2000). Simulation budget allocation for further enhancing theefficiency of ordinal optimization. *Discrete Event Dynamic Systems*, 10(3):251–270.

Chen, S., Lin, T., King, I., Lyu, M. R., and Chen, W. (2014). Combinatorial pure exploration of multi-armed bandits. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Degenne, R. (2023). On the existence of a complexity in fixed budget bandit identification. In Neu, G. and Rosasco, L., editors, *The Thirty Sixth Annual Conference on Learning Theory, 12-15 July 2023, Bangalore, India*, volume 195 of *Proceedings of Machine Learning Research*, pages 1131–1154. PMLR.

Degenne, R. and Koolen, W. M. (2019). Pure exploration with multiple correct answers. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 14564–14573.

Dembo, A. and Zeitouni, O. (2010). *Large Deviations Techniques and Applications*. Springer, second edition.

Du, Y., Kuroki, Y., and Chen, W. (2021). Combinatorial pure exploration with full-bandit or partial linear feedback. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 7262–7270. AAAI Press.

Ghosh, D., Hanawal, M. K., and Zlatanov, N. (2024). Fixed budget best arm identification in unimodal bandits. *Transactions on Machine Learning Research*.

Glynn, P. W. and Juneja, S. (2004). A large deviations perspective on ordinal optimization. In *Conference on Winter simulation*, pages 577–585. IEEE Computer Society.

Gupta, S., Joshi, G., and Yagan, O. (2021). Best-arm identification in correlated multi-armed bandits. *CoRR*, abs/2109.04941.

Harper, F. M. and Konstan, J. A. (2015). The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4).

Jamieson, K. G. and Nowak, R. D. (2014). Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting. In *48th Annual Conference on Information Sciences and Systems, CISS 2014, Princeton, NJ, USA, March 19-21, 2014*, pages 1–6. IEEE.

Jang, K., Jun, K., Yun, S., and Kang, W. (2021). Improved regret bounds of bilinear bandits using action space analysis. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021*, volume 139 of *Proceedings of Machine Learning Research*, pages 4744–4754. PMLR.

Johari, R., Koomen, P., Pekelis, L., and Walsh, D. (2017). Peeking at a/b tests: Why it matters, and what to do about it. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 1517–1525, New York, NY, USA. Association for Computing Machinery.

Jourdan, M., Degenne, R., Baudry, D., de Heide, R., and Kaufmann, E. (2022). Top two algorithms revisited. *Advances in Neural Information Processing Systems*, 35:26791–26803.

Jourdan, M., Degenne, R., and Kaufmann, E. (2023). An $\varepsilon$-best-arm identification algorithm for fixed-confidence and beyond. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Karnin, Z., Koren, T., and Somekh, O. (2013). Almost optimal exploration in multi-armed bandits. In *International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 1238–1246. PMLR.

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

Komiyama, J., Tsuchiya, T., and Honda, J. (2022). Minimax optimal algorithms for fixed-budget best arm identification. In *Advances in Neural Information Processing Systems*.

Koval, M. C., King, J. E., Pollard, N. S., and Srinivasa, S. S. (2015). Robust trajectory selection for rearrangement planning as a multi-armed bandit problem. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2015, Hamburg, Germany, September 28 - October 2, 2015*, pages 2678–2685. IEEE.

Kučera, L. and Hladík, M. (2017). Linear algebra (not only) for computer scientists. Lecture notes, Charles University, Faculty of Mathematics and Physics.

Lee, J., Honda, J., and Sugiyama, M. (2023). Thompson exploration with best challenger rule in best arm identification. In Yanikoglu, B. and Buntine, W. L., editors, *Asian Conference on Machine Learning, ACML 2023, 11-14 November 2023, Istanbul, Turkey*, volume 222 of *Proceedings of Machine Learning Research*, pages 646–661. PMLR.

Li, L., Chu, W., Langford, J., and Schapire, R. E. (2010). A contextual-bandit approach to personal-

ized news article recommendation. In Rappa, M., Jones, P., Freire, J., and Chakrabarti, S., editors, *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*, pages 661–670. ACM.

Li, Z., Ratliff, L. J., Nassif, H., Jamieson, K. G., and Jain, L. (2022). Instance-optimal PAC algorithms for contextual bandits. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Mukherjee, A. and Tajer, A. (2023). Sprt-based efficient best arm identification in stochastic bandits. *IEEE J. Sel. Areas Inf. Theory*, 4:128–143.

Paudel, A. and Stein, G. J. (2023). Data-efficient policy selection for navigation in partial maps via subgoal-based abstraction. In *IROS*, pages 11281–11288.

Russo, D. (2016). Simple bayesian algorithms for best arm identification. In *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 1417–1418. PMLR.

Russo, D. (2020). Simple bayesian algorithms for best-arm identification. *Operations Research*, 68(6):1625–1647.

Saito, Y., Aihara, S., Matsutani, M., and Narita, Y. (2021). Open bandit dataset and pipeline: Towards realistic and reproducible off-policy evaluation. In Vanschoren, J. and Yeung, S., editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Shahrampour, S., Noshad, M., and Tarokh, V. (2017). On sequential elimination algorithms for best-arm identification in multi-armed bandits. *IEEE Transactions on Signal Processing*, 65(16):4281–4292.

Shang, X., Heide, R., Menard, P., Kaufmann, E., and Valko, M. (2020). Fixed-confidence guarantees for bayesian best-arm identification. In *International Conference on Artificial Intelligence and Statistics*, pages 1823–1832. PMLR.

Soare, M., Lazaric, A., and Munos, R. (2014). Best-arm identification in linear bandits. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Takemori, S., Umeda, Y., and Gopalan, A. (2024). Model-based best arm identification for decreasing bandits. In Dasgupta, S., Mandt, S., and Li, Y., editors, *International Conference on Artificial Intelligence and Statistics, 2-4 May 2024, Palau de Congressos, Valencia, Spain*, volume 238 of *Proceedings of Machine Learning Research*, pages 1567–1575. PMLR.

Tao, C., Blanco, S., and Zhou, Y. (2018). Best arm identification in linear bandits with linear dimension dependency. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4877–4886. PMLR.

Villar, S., Bowden, J., and Wason, J. (2015). Multi-armed bandit models for the optimal design of clinical trials: Benefits and challenges. *Statistical Science*, 30:199–215.

Wang, J. and Tiwari, R. (2023). Adaptive designs for best treatment identification with top-two thompson sampling and acceleration. *Pharmaceutical Statistics*, 22(6):1089–1103. Epub 2023 Aug 12.

Wang, P., Ariu, K., and Proutière, A. (2024). On universally optimal algorithms for A/B testing. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Wang, P.-A., Tzeng, R.-C., and Proutiere, A. (2023). Best arm identification with fixed budget: A large deviation perspective. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Xiong, Z., Camilleri, R., Fazel, M., Jain, L., and Jamieson, K. G. (2024). A/B testing and best-arm identification for linear bandits with robustness to non-stationarity. In Dasgupta, S., Mandt, S., and Li, Y., editors, *International Conference on Artificial Intelligence and Statistics, 2-4 May 2024, Palau de Congressos, Valencia, Spain*, volume 238 of *Proceedings of Machine Learning Research*, pages 1585–1593. PMLR.

Yang, J. and Tan, V. Y. F. (2022). Minimax optimal fixed-budget best arm identification in linear bandits. In *NeurIPS*.

You, W., Qin, C., Wang, Z., and Yang, S. (2023). Information-directed selection for top-two algorithms. In Neu, G. and Rosasco, L., editors, *The Thirty Sixth Annual Conference on Learning Theory, COLT 2023, 12-15 July 2023, Bangalore, India*, volume 195 of *Proceedings of Machine Learning Research*, pages 2850–2851. PMLR.

Yu, J. Y. and Mannor, S. (2011). Unimodal bandits. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 41–48, Madison, WI, USA. Omnipress.

Yu, X., Shao, H., Lyu, M. R., and King, I. (2018). Pure exploration of multi-armed bandits with heavy-

tailed payoffs. In Globerson, A. and Silva, R., editors, *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018*, pages 937–946. AUAI Press.

Zhao, Y., Stephens, C., Szepesvári, C., and Jun, K. (2023). Revisiting simple regret: Fast rates for returning a good arm. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J., editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 42110–42158. PMLR.

## Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. Yes: We have clearly described the mathematical setting, assumptions, algorithm, and model in Section 2–4.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. Yes: We have provided a detailed analysis of the properties and complexity of our proposed algorithm in Section 3–4.
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. Yes. We have provided anonymized source code in the supplementary material.

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. Yes. We have clearly stated all the assumptions of our theoretical results.
   (b) Complete proofs of all theoretical results. Yes. Most of the proofs are provided in the supplementary material.
   (c) Clear explanations of any assumptions. Yes. We have provided clear explanations of all assumptions. Most assumptions are standard in the literature.

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results. Yes. We have provided the code and instructions in the supplementary material.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). Not Applicable because we consider online learning setting.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). Yes. We have provided clear definitions of all measures and statistics. In particular, we measure the performance of our algorithm by the probability of error, normalized by sample complexity $H_1$ or $H_2$.
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). Yes. We conducted experiment with a PC with $\sim 10^2$ CPU cores and it finishes in less than a half day. Technically, one can reproduce it with a laptop, but it may take a long time.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. Yes, we used several real-world datasets, and we have cited accordingly.
   (b) The license information of the assets, if applicable. Yes. We plan to release our code under MIT license.
   (c) New assets either in the supplemental material or as a URL, if applicable. Not applicable. We have not generated any new dataset.
   (d) Information about consent from data providers/curators. Not applicable.
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. Not applicable. We do not have any human subjects.

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. Not Applicable. We do not have any human subjects.
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. Not Applicable. We do not have any human subjects.
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. Not Applicable. We do not have any human subjects.

# Supplementary Materials of "Rate-optimal Design for Anytime Best Arm Identification"

## A NOTATION

Table 2 summarizes major notation used in this paper.

We use $D\left(Q_i \| P_i\right)$ to denote the KL divergence between two distributions $Q_i$ and $P_i$. Since we assume the reward distributions are Gaussian, we have

$$D\left(Q_i \| P_i\right) = \frac{\left(P_i - Q_i\right)^2}{2}. \tag{23}$$

During the lemmas, we use $D\left(Q_i \| P_i\right)$ for the results that does not specifically require the Gaussian assumption. For lemmas where the Gaussianity is required, we use $\frac{(P_i - Q_i)^2}{2}$ instead of $D\left(Q_i \| P_i\right)$. Appendix K describes the reward assumptions required for each lemma.

Table 2: Notation table

| Symbol | Meaning |
|---|---|
| $[K] = \{1, 2, \ldots, K\}$ | Set of arms |
| $T$ | Total number of samples |
| $\mathcal{P}$ | Set of true distributions ($= \mathbb{R}$ for Gaussian) |
| $\mathcal{Q}$ | Set of empirical distributions ($= \mathbb{R}$ for Gaussian) |
| $\boldsymbol{P} = (P_1, P_2, \ldots, P_K) \in \mathcal{P}^K$ | True means (unknown). |
| $\boldsymbol{Q}(t) = (Q_1(t), Q_2(t), \ldots, Q_K(t)) \in \mathcal{Q}^K$ | Empirical means at time $t$ |
| $N_i(t)$ | # of pulls of arm $i$ at time $t$ |
| $B$ | # of batches |
| $N$ | # of samples per batch (Algorithm 2). $B = \lfloor T/N \rfloor$. |
| $\sigma^2 = 1$ | Variance of the reward (known, unit) |
| $H(\boldsymbol{P})$ | General complexity measure (risk measure) |
| $H_1(\boldsymbol{P})$ | $\sum_{i \neq i^*} (\max_j P_j - P_i)^{-2}$ |
| $H_2(\boldsymbol{P})$ | $\max_i i \Delta_{(i)}^{-2}$ |
| $H_3(\boldsymbol{P})$ | Characterizing complexity of SR. Defined in Eq. (104) |
| $D(Q_i \| P_i) = (P_i - Q_i)^2/2$ | Gaussian KL divergence |
| $i^*(\boldsymbol{P}) := \arg\max_{i \in [K]} P_i$ | We assume uniqueness for true $\boldsymbol{P}$. Namely, $\lvert i^*(\boldsymbol{P}) \rvert = 1$ |
| $i^*(\boldsymbol{Q}) := \arg\max_{i \in [K]} Q_i$ | Possibly non-unique for discrete cases (e.g., Bernoulli). |
| $P^*, Q^*$ | $\max_i P_i$, $\max_i Q_i$ |
| $D_i(\boldsymbol{Q})$ | Defined in Eq. (16) |
| $\boldsymbol{Q}_b = (Q_{b,1}, Q_{b,2}, \ldots, Q_{b,K})$ | Empirical mean vector at $b$-th batch |
| $Q_{b,i}$ | Empirical mean of the i-th arm at $b$-th batch |
| $N_{b,i}$ | # of i-th arm pulls in $b$-th batch |
| $\Delta^{(K)}$ | $K$-dimensional simplex |
| $\boldsymbol{w}(\boldsymbol{Q}) = (w_1(\boldsymbol{Q}), w_2(\boldsymbol{Q}), \ldots, w_K(\boldsymbol{Q}))$ | Weight function (allocation) |
| $\boldsymbol{w}_b = (w_{b,1}, w_{b,2}, \ldots, w_{b,K})$ | Realized batch weights at batch $b$ on arm $i$ |
| $\Delta_i$ | $P^* - P_i$ |
| $\Delta_{(i)}$ | $\Delta_j$ of the $i$-th best arm $j$ |
| $\Delta_i^Q$ | $(\max_j Q_j) - Q_i$ |
| $R_{\mathcal{A}}$ | Rate of Algorithm $\mathcal{A}$ (Definition 1) |
| $\lceil \boldsymbol{x} \rceil^{\text{int}} = (\lceil x_1 \rceil^{\text{int}}, \lceil x_2 \rceil^{\text{int}}, \ldots, \lceil x_K \rceil^{\text{int}})$ | Rounding fractional allocation to integer (Appendix C). |
| $\bar{\boldsymbol{Q}}_{b-1}$ | $(\bar{Q}_{b,1}, \bar{Q}_{b,2}, \ldots, \bar{Q}_{b,K})$ |
| $\bar{Q}_{b,i}$ | $\frac{\sum_{b'=1}^{b} w_{b',i} Q_{b',i}}{\sum_{b'=1}^{b} w_{b',i}}$ (= empirical mean up to batch $b$) |
| $\overline{\log}(K)$ | $\frac{1}{2} + \sum_{i=2}^{K} \frac{1}{i}$ |

**Algorithm 4** Pooled Allocation

**Require:** Number of samples $T$, Number of batches $B$, optimal weight function $\boldsymbol{w}^*(\boldsymbol{Q})$, optimal recommendation $i^*(\boldsymbol{Q})$ (any fixed tie-break).

Draw each arm for $T/B$ times (equivalently, $w_{b,i} = \mathbf{1}[i = b]$ for $b \in [K]$. Let $Q_{1,i}$ denote the initial empirical mean of arm $i$ (first initialization batch)).
Build empirical means $\boldsymbol{Q}_1^{\text{pool}}$.
Initialize the number of victories for each arm: $\boldsymbol{v} = (v_1, v_2, \ldots, v_K) = (0, 0, \ldots, 0)$.
**for** $b = K+1, K+2, \ldots, B$ **do**
  Increase $v_{i^*(\boldsymbol{Q}_{b-K}^{\text{pool}})}$ by one.
  Calculate optimal allocation based on the pooled means:

$$\boldsymbol{w}_b = \boldsymbol{w}^*(\boldsymbol{Q}_{b-K}^{\text{pool}}) \tag{24}$$

  Draw each arm for $w_{b,i} \times (T/B)$ times and obtain empirical mean vector $\boldsymbol{Q}_b$.
  Update the delayed means:

$$\boldsymbol{Q}_{b-K+1}^{\text{pool}} = (\mathbf{1} - \boldsymbol{w}_b) \odot \boldsymbol{Q}_{b-K}^{\text{pool}} + \boldsymbol{w}_b \odot \boldsymbol{Q}_b \tag{25}$$

  where $\odot$ denotes element-wise multiplication.
**end for**
**return** $J(T) = \arg\max_i v_i$

# B TWO-APPROXIMATION ALGORITHM

This section introduces Pooled Allocation (Algorithm 4), which achieves the two-approximation rate of Theorem 2. Since we consider $K, B$ to be sufficiently large and we do not use this algorithm in practice, we assume as if $T/B$ is an integer for simplicity.

**Theorem 10.** Assume that $K^2 = o(T)$. Algorithm 4 achieves the following rate:

$$R_{\text{PooledAllocation}} \geq \frac{R^{\text{go}}}{2} - \varepsilon \tag{26}$$

for any $\varepsilon > 0$.

Theorem 10 does not use the property of Gaussian KL divergence (i.e., squared form) and holds for a large class of distributions such as an exponential family of distributions. Therefore, we use KL divergence formula $D(Q_i \| P_i)$. For the case of Gaussian, $D(Q_i \| P_i) = (Q_i - P_i)^2/2$.

*Proof of Theorem 10.* By assumption $K^2 = o(T)$, we can take $B$ such that $K = o(B), KB = o(T)$. For $\boldsymbol{P}$ such that $i^*(\boldsymbol{P}) \neq J(T)$, we have

$$\frac{T}{B} \sum_{b \in [B]} \sum_i w_{b,i} D(Q_{b,i} \| P_i) \tag{27}$$

$$\geq \frac{T}{B} \sum_{b \in [B-K]} \sum_i w_i^*(\boldsymbol{Q}_b^{\text{pool}}) D\left(Q_{b,i}^{\text{pool}} \| P_i\right) \qquad \text{(Lemma 11 with } B' = B \text{ and } \boldsymbol{P}' = \boldsymbol{P})$$

$$\geq \frac{T(B-K)}{2B} \frac{R^{\text{go}}}{H(\boldsymbol{P})}, \tag{28}$$

where the last inequality follows from the facts that $w^*(\boldsymbol{Q}), i^*(\boldsymbol{Q})$ are optimal weight and recommendation, and $i^*(\boldsymbol{P}) \neq J(T)$ implies that at least half among $B - K$ batches have $i^*(\boldsymbol{P}) \neq i^*(\boldsymbol{Q}_b^{\text{pool}})$. By using this, we have

$$\mathbb{P}[i^*(\boldsymbol{P}) \neq J(T)] \leq \mathbb{P}\left[\frac{T}{B} \sum_{b \in [B]} \sum_i w_{b,i} D(Q_{b,i} \| P_i) \geq \frac{T(B-K)}{2B} \frac{R^{\text{go}}}{H(\boldsymbol{P})}\right] \tag{29}$$

$$\leq \exp\left(-\frac{T(B-K)}{2B}\frac{R^{\mathrm{go}}}{H(\boldsymbol{P})} + o(T)\right) \qquad \text{(by } KB = o(T) \text{ and Theorem 15)}$$

and the proof is complete by using $K = o(B)$. □

**Lemma 11.** For all $B' \in \{K, K+1, \ldots, B\}$ and any $\boldsymbol{P}'$, we have the following:

$$\sum_{b\in[B']}\sum_{i} w_{b,i} D\left(Q_{b,i} \| P'_i\right) \geq \left(\sum_{b\in[B'-K]}\sum_{i} w_{b+K,i} D\left(Q^{\mathrm{pool}}_{b,i} \| P'_i\right) + \sum_{i\in[K]} D\left(Q^{\mathrm{pool}}_{B'-K+1,i} \| P'_i\right)\right). \tag{30}$$

*Proof of Lemma 11.* We show Eq.(30) by induction for $B'$. It trivially holds for $B = K$. Assume that it holds for $B'$. Then

$$\begin{aligned}
&\sum_{b\in[B'+1]}\sum_{i\in[K]} w_{b,i} D\left(Q_{b,i} \| P'_i\right) \\
&\quad\geq \sum_{b\in[B'-K]}\sum_{i\in[K]} w_{b+K,i} D\left(Q^{\mathrm{pool}}_{b,i} \| P'_i\right) + \sum_{i\in[K]} D\left(Q^{\mathrm{pool}}_{B'-K+1,i} \| P'_i\right) \\
&\qquad + \underbrace{\sum_{i\in[K]} w_{B'+1,i} D\left(Q_{B'+1,i} \| P'_i\right)}_{B'+1\text{-th batch}} \qquad \text{(By the induction hypothesis)} \\
&\quad= \sum_{b\in[B'-K]}\sum_{i\in[K]} w_{b+K,i} D\left(Q^{\mathrm{pool}}_{b,i} \| P'_i\right) + \sum_{i\in[K]} w_{B'+1,i} D\left(Q^{\mathrm{pool}}_{B'-K+1,i} \| P'_i\right) \\
&\qquad + \sum_{i\in[K]} (1 - w_{B'+1,i}) D\left(Q^{\mathrm{pool}}_{B'-K+1,i} \| P'_i\right) + \sum_{i\in[K]} w_{B'+1,i} D\left(Q_{B'+1,i} \| P'_i\right) \\
&\quad= \sum_{b\in[B'-K+1]}\sum_{i\in[K]} w_{b+K,i} D\left(Q^{\mathrm{pool}}_{b,i} \| P'_i\right) \\
&\qquad + \sum_{i\in[K]} (1 - w_{B'+1,i}) D\left(Q^{\mathrm{pool}}_{B'-K+1,i} \| P'_i\right) + \sum_{i\in[K]} w_{B'+1,i} D\left(Q_{B'+1,i} \| P'_i\right) \\
&\quad\geq \sum_{b\in[B'-K+1]}\sum_{i\in[K]} w_{b+K,i} D\left(Q^{\mathrm{pool}}_{b,i} \| P'_i\right) + \sum_{i\in[K]} D\left(Q^{\mathrm{pool}}_{B'-K+2,i} \| P'_i\right),
\end{aligned}$$

where the last inequality follows from Jensen's inequality and $Q^{\mathrm{pool}}_{B'-K+2,i} = w_{b+K,i} Q_{B'+1,i} + (1 - w_{b+K,i}) Q^{\mathrm{pool}}_{B'-K+1,i}$. Therefore, the inequality holds for $B' + 1$. □

## C A CONSTANT-RATIO CEILING

For a practical implementation, the size of $B$ is limited, and thus considering the fractional part on the number of pulls is important. This section introduces a constant-ratio ceiling function $\lceil \boldsymbol{x} \rceil^{\mathrm{int}} = (\lceil x_1 \rceil^{\mathrm{int}}, \lceil x_2 \rceil^{\mathrm{int}}, \ldots, \lceil x_K \rceil^{\mathrm{int}})$ that converts a fractional number of pulls to an integer number of pulls. This ceiling is used in Algorithm 2.

**Lemma 12.** Assume that $N_B \geq 2K$. Then, the number of pulls $\lceil w_i N_B \rceil^{\mathrm{int}}$ computed by Algorithm 5 satisfies the followings:

- For each $i \in [K]$, $\lceil w_i N_B \rceil^{\mathrm{int}} \in \mathbb{N}$.
- $\sum_{i\in[K]} \lceil w_i N_B \rceil^{\mathrm{int}} = N_B$.
- For each $i \in [K]$, $\lceil w_i N_B \rceil^{\mathrm{int}} \geq w_i N_B / 4$.

In other words, $\lceil w_i N_B \rceil^{\mathrm{int}}$ an integer allocation that is at least the constant $(1/4)$ times of the optimal allocation for each arm.

**Algorithm 5** From fractional number of pulls to integer number of pulls

Input: Weight function $\boldsymbol{w} = (w_1, w_2, \ldots, w_K) \in \Delta^{(K)}$, $N_B \in \mathbb{N}$.
$N'_B = N_B - \sum_{i\in[K]} \mathbf{1}[w_i > 0]$
**for** $i \in [K]$ **do**
  **if** $w_i > 0$ **then**
    $N_i = 1 + \lfloor w_i N'_B \rfloor$
  **else**
    $N_i = 0$
  **end if**
**end for**
**while** $\sum_{i\in[K]} N_i < N_B$ **do**
  Choose $i$ with probability $w_i$ and set $N_i = N_i + 1$.
**end while**
**return** $\lceil w_i N_B \rceil^{\text{int}} = N_i$.

*Proof of Lemma 12.* The first two properties are trivial. For the third property, For $i : w_i = 0$, we have $\lceil w_i N_B \rceil^{\text{int}} = 0 = w_i N_B$. For $i : w_i N_B \in (0, 4]$, we have $\lceil w_i N_B \rceil^{\text{int}} \geq 1 \geq \frac{1}{4} w_i N_B$. For $i : w_i N_B > 4$, we have

$$\begin{aligned}
\lceil w_i N_B \rceil^{\text{int}} &\geq \lfloor w_i N'_B \rfloor && (31)\\
&\geq w_i N'_B - 1 && (32)\\
&\geq w_i N_B/2 - 1 && (\text{by } N_B \geq 2K)\\
&\geq w_i N_B/4 && (33)
\end{aligned}$$

and the proof is complete. □

# D PROOF OF THEOREM 3

For convenience, we restate the theorem here.

**Theorem 3.** (Trackability) Let $w_i(\boldsymbol{Q}) \geq w_{\min}$ holds for any $\boldsymbol{Q}, i$ for some value $w_{\min} = w_{\min}(K) > 0$ that only depends on $K$. Let $B := \lfloor T/N \rfloor \geq 2/w_{\min}$. Consider Algorithm 2 with $C_{\text{suf}} \in (0, 1)$. There exists a universal constant $C_{\text{track}} > 0$ such that the following holds for any sequence of $\boldsymbol{Q}_1, \boldsymbol{Q}_2, \ldots, \boldsymbol{Q}_B$:

$$\frac{1}{B} \sum_{b\in[B]} \sum_{i\in[K]} w_{b,i} \frac{(Q_{b,i} - P_i)^2}{2} \geq C_{\text{track}} \inf_{\boldsymbol{Q}: i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})} \left( \sum_{i\in[K]} w_i(\boldsymbol{Q})(Q_i - P_i)^2/2 \right). \tag{8}$$

*Proof of Theorem 3.* We denote $\boldsymbol{w}(\boldsymbol{Q}) = (w_1, w_2, \ldots, w_K) \in \Delta^{(K)}$ to represent the ideal allocation of one-shot game.

We consider a batched algorithm and let $\boldsymbol{w}_{b,i}, Q_{b,i}$ be corresponding allocation and empirical means. We denote averaged quality over $B$ batches as:

$$\bar{w}_{B,i} = \sum_b \frac{w_{b,i}}{B} \tag{34}$$

$$\bar{Q}_{B,i} = \frac{\sum_b w_{b,i} Q_{b,i}}{\sum_b w_{b,i}}. \tag{35}$$

Let us use a widetilde operator to denote amortized[9] weights and amortized means as follows:

$$\widetilde{w}_{b,i} = w_{b,i} + \frac{C_{\text{suf}}}{(b-1)} \sum_{b'=1}^{b-1} w_{b',i} \tag{36}$$

[9]Intuitively speaking, we move part of weights during the first $b - 1$ rounds to the weight of round $b$. This is purely analytical technique and does not change the algorithmic design.

$$\widetilde{Q}_{b,i} = \frac{w_{b,i}Q_{b,i} + \frac{C_{\mathrm{suf}}}{(b-1)}\sum_{b'=1}^{b-1} w_{b',i}\bar{Q}_{b-1,i}}{w_{b,i} + \frac{C_{\mathrm{suf}}}{(b-1)}\sum_{b'=1}^{b-1} w_{b',i}}. \tag{37}$$

We consider a sampling rule such that

$$w_{b,i} \geq \frac{w_i(\bar{\boldsymbol{Q}}_{b-1})}{4} \tag{38}$$

holds for all $i$ such that

$$\frac{1}{b-1}\sum_{b'=1}^{b-1} w_{b',i} \leq \frac{1}{C_{\mathrm{suf}}} w_i(\bar{\boldsymbol{Q}}_{b-1}) \tag{39}$$

for some $C_{\mathrm{suf}} \in (0,1)$. Here, the factor 1/4 in Eq. (38) is derived from the fractional allocation ($\lceil\cdot\rceil^{\mathrm{int}}$) and is non-essential. Algorithm 2, combined with a fractional allocation (Lemma 12), belongs to this class of sampling rule.

(38),(39) imply

$$\widetilde{w}_{b,i} \geq \frac{w_i(\bar{\boldsymbol{Q}}_{b-1})}{4} \tag{40}$$

for any $i$.

Moreover, we have

$$\sum_{b=1}^{B}\sum_{i} w_{b,i}(Q_{b,i} - P_i)^2 \tag{41}$$

$$= \sum_{i} w_{B,i}(Q_{B,i} - P_i)^2 + \frac{C_{\mathrm{suf}}}{B-1}\sum_{b=1}^{B-1}\sum_{i} w_{b,i}(Q_{b,i} - P_i)^2 + \frac{B-1-C_{\mathrm{suf}}}{B-1}\sum_{b=1}^{B-1}\sum_{i} w_{b,i}(Q_{b,i} - P_i)^2 \tag{42}$$

$$\geq \sum_{i} \widetilde{w}_{B,i}(\widetilde{Q}_{B,i} - P_i)^2 + \frac{B-1-C_{\mathrm{suf}}}{B-1}\sum_{b=1}^{B-1}\sum_{i} w_{b,i}(Q_{b,i} - P_i)^2 \quad \text{(Jensen's inequality)} \tag{43}$$

Repeatedly applying (43) yields

$$\sum_{b=1}^{B} w_{b,i}(Q_{b,i} - P_i)^2 \tag{44}$$

$$\geq \widetilde{w}_{B,i}(\widetilde{Q}_{B,i} - P_i)^2 + \frac{B-1-C_{\mathrm{suf}}}{B-1}\widetilde{w}_{B-1,i}(\widetilde{Q}_{B-1,i} - P_i)^2 + \tag{45}$$

$$\frac{B-1-C_{\mathrm{suf}}}{B-1}\frac{B-2-C_{\mathrm{suf}}}{B-2}\widetilde{w}_{B-2,i}(\widetilde{Q}_{B-2,i} - P_i)^2 + \dots \tag{46}$$

$$= \sum_{b} C_b \widetilde{w}_{b,i}(\widetilde{Q}_{b,i} - P_i)^2, \tag{47}$$

where

$$C_b = \prod_{b'=b}^{B-1} \frac{b' - C_{\mathrm{suf}}}{b'} \tag{48}$$

is a increasing sequence in $b$.[10] Here, $C_b$ is the weight of the process such that, moving some weight from $\sum_{b'}^{b-1} w_{b'}$ to $w_b$ to build $\widetilde{w}_{b,i}$. This fact implies the weighted average of $(w_{b',i})_{b'\in[b]}$ and $(C'_b\widetilde{w}_{b',i})_{b'\in[b]}$ is identical, namely,

$$\sum_{b'\leq b} w_{b',i} = \sum_{b'\leq b} C_{b'}\widetilde{w}_{b',i} \tag{49}$$

[10]Note that $C_B = 1$.

$$\bar{Q}_{b,i} = \frac{\sum_{b' \le b} C_{b'} \widetilde{w}_{b',i} \widetilde{Q}_{b',i}}{\sum_{b' \le b} C_{b'} \widetilde{w}_{b',i}}. \tag{50}$$

Moreover, $\sum_b C_b \ge b(1 - C_{\mathrm{suf}}) = \Omega(B)$.[11]

For ease of discussion, we use $1/w_{\min}$ as if it were an integer.[12] The variance term $V(B)$ is

$$V(B) := \sum_b \sum_i C_b \widetilde{w}_{b,i} \left( (\widetilde{Q}_{b,i} - P_i)^2 - (\bar{Q}_{B,i} - P_i)^2 \right) \tag{51}$$

$$\ge \frac{1}{2} \sum_{b=1/w_{\min}} \sum_i C_b \widetilde{w}_{b,i} (\widetilde{Q}_{b,i} - \bar{Q}_{b-1,i})^2 \qquad \text{(by Lemma 13)}$$

$$\ge \frac{3}{40} \sum_{b=1/w_{\min}} \sum_i C_b \widetilde{w}_{b,i} (\bar{Q}_{B,i} - \bar{Q}_{b-1,i})^2 \qquad \text{(by Lemma 14)}$$

$$\tag{52}$$

and finally, we have

$$\text{Eq. (47)} \ge \sum_{b=1/w_{\min}} \sum_i C_b \widetilde{w}_{b,i} (\bar{Q}_{B,i} - P_i)^2 + V(B) \tag{53}$$

$$\ge \frac{3}{40} \sum_{b=1/w_{\min}} C_b \sum_i \widetilde{w}_{b,i} \left( (\bar{Q}_{B,i} - P_i)^2 + (\bar{Q}_{B,i} - \bar{Q}_{b-1,i})^2 \right) \qquad \text{(by (52))}$$

$$\ge \frac{3}{80} \sum_{b=1/w_{\min}} C_b \sum_i \widetilde{w}_{b,i} (\bar{Q}_{b-1,i} - P_i)^2 \qquad (X^2 + Y^2 \ge \tfrac{1}{2}(X-Y)^2)$$

$$\ge \frac{3}{320} \sum_{b=1/w_{\min}} C_b \sum_i w_i(\bar{\boldsymbol{Q}}_{b-1}) (\bar{Q}_{b-1,i} - P_i)^2 \qquad \text{(by (40))}$$

$$\ge \frac{3}{320} \left( \sum_{b=1/w_{\min}} C_b \right) \inf_{\boldsymbol{Q}} \sum_i w_i(\boldsymbol{Q}) (Q_i - P_i)^2 \tag{54}$$

and the fact $\sum_b C_b = \Omega(B)$ completes the proof of Theorem 3. □

### D.1 Transformation 1

**Lemma 13.** (Transformation 1) For any $i \in [K]$, it holds that

$$\sum_{b=1}^{B} C_b \widetilde{w}_{b,i} \left( (\widetilde{Q}_{b,i} - P_i)^2 - (\bar{Q}_{B,i} - P_i)^2 \right) \ge \frac{1}{2} \sum_{b=1/w_{\min}}^{B} C_b \widetilde{w}_{b,i} (\widetilde{Q}_{b,i} - \bar{Q}_{b-1,i})^2 \tag{55}$$

*Proof of Lemma 13.* During the proof, we omit index $i$ for ease of notation. For the ease of notation we use $v_b = C_b \widetilde{w}_b$. Let us represent the LHS minus the RHS of (55) as

$$X_{b'} := \sum_{b=1}^{b'} v_b \left( (\widetilde{Q}_b - P)^2 - (\bar{Q}_{b'} - P)^2 \right) - \frac{1}{2} \sum_{b=1}^{b'} v_b (\widetilde{Q}_b - \bar{Q}_{b-1})^2 \tag{56}$$

We first show

$$X_{b'} - X_{b'-1} \ge 0 \tag{57}$$

for $b' \ge 1/w_{\min} + 1$.

Letting

$$A := v_{b'} \tag{58}$$

[11] Sequence $A_b$ : s.t. $A_1 = 1$, $A_b = ((b - 1 - C_{\mathrm{suf}})/(b-1))A_{b-1} + 1$ satisfies $A_b \ge b(1 - C_{\mathrm{suf}})$.

[12] We can easily formalize this by introducing a floor operator.

$$B := \sum_{b=1}^{b'-1} v_b \tag{59}$$

$$X := \widetilde{Q}_{b'} - P \tag{60}$$

$$Y := \bar{Q}_{b'-1} - P, \tag{61}$$

we have

$$\begin{aligned}
&X_{b'} - X_{b'-1} && (62)\\
&= AX^2 - \left( \sum_{b=1}^{b'} v_b (\bar{Q}_{b'} - P)^2 - \sum_{b=1}^{b'-1} v_b (\bar{Q}_{b'-1} - P)^2 \right) - \frac{1}{2} A (X-Y)^2 && (63)\\
&= AX^2 - \left( (A+B) \left( \frac{AX+BY}{A+B} \right)^2 - BY^2 \right) - \frac{1}{2} A (X-Y)^2 && \text{(by (50))}\\
&= AX^2 + BY^2 - (A+B) \left( \frac{AX+BY}{A+B} \right)^2 - \frac{1}{2} A (X-Y)^2 && (64)\\
&= \frac{1}{A+B} AB (X-Y)^2 - \frac{1}{2} A (X-Y)^2 && (65)
\end{aligned}$$

Eq.(65) is $\geq 0$ if $A \leq B$ holds. In the following, we show $A \leq B$ for $b'$ is sufficiently large.

First, it is easy to see that

$$\frac{\sum_{b=1}^{b'} C_b}{C_{b'+1}} \geq b' - 1. \tag{66}$$

By (40), we have $w_{\min} \leq \widetilde{w}_b \leq 1$ uniformly. Therefore,

$$\begin{aligned}
B &= \sum_{b=1}^{b'-1} C_b \widetilde{w}_b && (67)\\
&\geq w_{\min} \sum_{b=1}^{b'-1} C_b && (68)\\
&\geq w_{\min} (b'-1) && \text{(by (66))}\\
&\geq 1 \geq A. && \text{(if } b' = 1/w_{\min} + 1\text{)}
\end{aligned}$$

In summary,

$$\begin{aligned}
X_B &= \sum_{b=1}^{B} (X_b - X_{b-1}) && (69)\\
&= \sum_{b=1/w_{\min}+1}^{B} (X_b - X_{b-1}) + X_{1/w_{\min}} && (70)\\
&\geq X_{1/w_{\min}} && (71)\\
&= \underbrace{\sum_{b=1}^{1/w_{\min}} v_b \left( (\widetilde{Q}_b - P)^2 - (\bar{Q}_{1/w_{\min}} - P)^2 \right)}_{\geq 0} - \frac{1}{2} \sum_{b=1}^{1/w_{\min}} v_b (\widetilde{Q}_b - \bar{Q}_{b-1})^2 && \text{(by (57)} \geq 0 \text{ for } b \geq 1/w_{\min} + 1\text{)}\\
&\geq -\frac{1}{2} \sum_{b=1}^{1/w_{\min}} v_b (\widetilde{Q}_b - \bar{Q}_{b-1})^2, && (72)
\end{aligned}$$

which is Eq. (55).

□

### D.2 Transformation 2

We omit index $i$ for ease of notation. For the ease of notation we use $v_b = C_b \widetilde{w}_b$.

**Lemma 14.** (Transformation 2) Let $C_{\text{trans2}} = \frac{20}{3}$. For any $c > 1$, it holds that

$$C_{\text{trans2}} \sum_{b=c}^{B} v_b (\widetilde{Q}_b - \bar{Q}_{b-1})^2 - \sum_{b=c}^{B} v_b (\bar{Q}_B - \bar{Q}_{b-1})^2 \geq 0 \tag{73}$$

*Proof of Lemma 14.* Let

$$v_{:b} = \sum_{b'=c}^{b-1} v_{b'} \tag{74}$$

$$v_{b:} = \sum_{b'=b}^{B} v_{b'} \tag{75}$$

$$x_b = \bar{Q}_b - \bar{Q}_{b-1} \tag{76}$$

$$x_{b:} = \sum_{b'=b}^{B} x_{b'} \tag{77}$$

By using

$$\frac{\sum_b v_b \widetilde{Q}_b}{\sum_b v_b} = \bar{Q}_b,$$

Eq. (73) is equivalent to

$$C_{\text{trans2}} \sum_{b=c}^{B} v_b \left( \frac{v_b + v_{:b}}{v_b} \right)^2 x_b^2 - \sum_{b=c}^{B} v_b x_{b:}^2 \geq 0. \tag{78}$$

We will show that this holds for any $(v_b)_{b\in[B]} > 0$, $(x_b)_{b\in[B]} \in \mathbb{R}$. This equation is a quadratic formula for $(x_b)_{b\in[B]}$.

Namely, letting $\boldsymbol{x} = (x_c, x_{c+1}, \ldots, c_B)^\top$ be a size $B - c + 1$ vector, Eq. (78) $= \boldsymbol{x}^T M \boldsymbol{x}$, where

$$M = \begin{bmatrix} D_c & -O_c & -O_c & \cdots & -O_c \\ -O_c & D_{c+1} & -O_{c+1} & \cdots & -O_{c+1} \\ -O_c & -O_{c+1} & D_{c+2} & \cdots & -O_{c+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -O_c & -O_{c+1} & -O_{c+2} & \cdots & D_B \end{bmatrix} \in \mathbb{R}^{(B-c+1)\times(B-c+1)}$$

where $D_i = C_{\text{trans2}} \frac{v_{:i+1}^2}{v_i} - v_{:i+1}$ and $O_i = v_{:i+1}$.

The problem of showing the positivity of Eq. (78) for all $(v_b)_{b\in[B]} > 0$, $(x_b)_{b\in[B]} \in \mathbb{R}$ is equivalent to showing the positive-definiteness of $M$ for any $(v_b)_{b\in[B]}$. Sylvester's criterion states that, a symmetric matrix $M$ is positive definite if and only if the Gaussian elimination transforms it to a triangular matrix with positive diagonal using only the elementary operation of adding a multiple of the column with another column (Kučera and Hladík, 2017). Namely, we will derive the diagonals of:

$$M' = \begin{bmatrix} D'_c & 0 & 0 & \cdots & 0 \\ -O'_c & D'_{c+1} & 0 & \cdots & 0 \\ -O'_c & -O'_{c+1} & D'_{c+2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -O'_c & -O'_{c+1} & -O'_{c+2} & \cdots & D'_B \end{bmatrix}$$

The $O'_i$ and $D'_i$ above follows the following recurrence relation:

$$D'_b = D_b - Z_{b-1}$$
$$O'_b = O_b + Z_{b-1}$$

where $Z_b = \sum_{i=c+1}^{b} \frac{O'^2_i}{D'_i}$.

**Claim 1.** Let $b \geq c$ be arbitrary. There exists $\xi_1, \cdots, \xi_b \in (3, \infty]$ such that $Z_b = \sum_{i=1}^{b} \frac{v_i}{\xi_i}$, where $x/\infty = 0$.

*Proof.* We will prove it by induction.

**Base Case $b = c$:** $Z_c = 0$ and thus the claim holds.

**Induction Step:** Suppose that the induction hypothesis holds for all $b < b_0$. For $b = b_0$,

$$Z_{b_0} = Z_{b_0-1} + \frac{O'^2_b}{D'_b} = \sum_{i=1}^{b-1} \frac{v_i}{\xi_i} + \frac{O'^2_b}{D'_b}$$

Which means, if we prove that $\frac{O'^2_{b_0}}{D'_{b_0}} = \frac{v_{b_0}}{\xi_{b_0}}$ for some $\xi_{b_0} > 3$, then it verifies the induction step.

$$\begin{aligned}
\frac{O'^2_{b_0}}{D'_{b_0}} &= \frac{(O_{b_0} + Z_{b_0-1})^2}{(D_{b_0} - Z_{b_0-1})} \\
&= \frac{\left(O_{b_0} + \sum_{i=1}^{b_0-1} \frac{v_i}{\xi_i}\right)^2}{(D_{b_0} - \sum_{i=1}^{b_0-1} \frac{v_i}{\xi_i})} && \text{(Induction hypothesis)} \\
&= \frac{\left(v_{:b_0+1} + \sum_{i=1}^{b_0-1} \frac{v_i}{\xi_i}\right)^2}{(C_{\text{trans2}} \frac{v^2_{:b_0+1}}{v_{b_0}} - v_{:b_0+1} - \sum_{i=1}^{b_0-1} \frac{v_i}{\xi_i})} \\
&\leq \frac{\left(1 + \frac{1}{3}\right)^2 v^2_{:b_0+1}}{(C_{\text{trans2}} v^2_{:b_0+1} - v_{b_0} \cdot v_{:b_0+1} - \frac{1}{3} v_{b_0} \cdot v_{:b_0+1})} \cdot v_{b_0} && (\xi_i > 3 \text{ for all } i \leq b_0 - 1,\ v_{b_0} \geq 0) \\
&\leq \frac{\left(1 + \frac{1}{3}\right)^2 v^2_{:b_0+1}}{C_{\text{trans2}} v^2_{:b_0+1} - (1 + \frac{1}{3}) v^2_{:b_0+1}} \cdot v_{b_0} && (v_{b_0} < v_{:b_0+1}) \\
&= \frac{\frac{16}{9}}{C_{\text{trans2}} - \frac{4}{3}} v_{b_0} = \frac{v_{b_0}}{3} && (C_{\text{trans2}} = \tfrac{20}{3})
\end{aligned}$$

Therefore, since $0 \leq \frac{O'^2_{b_0}}{D'_{b_0}} \leq \frac{v_{b_0}}{3}$, there exists a constant $\xi_{b_0} > 3$ such that $\frac{O'^2_{b_0}}{D'_{b_0}} = \frac{v_{b_0}}{\xi_{b_0}}$ and therefore one can write $Z_{b_0} = \sum_{i=1}^{b_0} \frac{v_i}{\xi_i}$, which proves the induction step, which derives the claim. □

Now, based on Claim 1, one can see $D'_b > 0$ for all $b \leq B$ since

$$\begin{aligned}
D'_b &= D_b - Z_{b-1} \\
&= \frac{C_{\text{trans2}} v^2_{:b+1}}{v_b} - v_{:b+1} - \sum_{i=1}^{b-1} \frac{v_i}{\xi_i} \\
&\geq (C_{\text{trans2}} - 1) v_{:b+1} - \frac{1}{3} v_{:b} && (\xi_i \geq 3 \text{ for all } i \in [B]) \\
&\geq \left(C_{\text{trans2}} - \frac{4}{3}\right) v_{:b+1} \geq 0
\end{aligned}$$

Therefore, $M'$ has only positive diagonal entries. By Sylvester's criterion $M$ is a positive definite matrix, which completes the proof. □

# E LARGE DEVIATION LEMMA

This section introduces the following Lemma, which uses the large deviation bound. We apply this lemma to analyze Almost Tracking (Algorithm 2) and Pooled Allocation (Algorithm 4).

**Lemma 15.** (PoE of the batched algorithm, large deviation) Let $x > 0$ be arbitrary. A BAI algorithm is batch-based if it splits $T$ rounds into $B$ predefined batches of consecutive subset of rounds. Let $N_{b,i}$ and $Q_{b,i}$ be corresponding statistics on the $b$-th batch. For any batch-based adaptive algorithm, it always holds that

$$\mathbb{P}\left[\sum_{b\in[B]}\sum_{i\in[K]} N_{b,i}\frac{(Q_{b,i}-P_i)^2}{2} \geq x\right] \leq \exp\left(-x + \frac{TK}{N}\max\left\{\log\left(\frac{xN}{TK},0\right)\right\} + \frac{TK}{N}(\log N + C_{\mathrm{cn}})\right), \tag{79}$$

where $C_{\mathrm{cn}} > 0$ is an absolute constant.

Lemma 15 does not use the property of Gaussian KL divergence (i.e., squared form) and holds in many cases with large deviation such as exponential family of distributions. Therefore, we use KL divergene formula $D\left(Q_i \| P_i\right)$.

Since the algorithm is adaptive, $N_{b,i}$ is a random variable depending on the history. To use the concentration inequality, we consider the following equivalent formulation of a batched algorithm.

- First, the environment (nature) draws $T$ arms for each batch $b \in [B]$ and arm $i \in [K]$, which are independent samples from $P_i$.
- Second, the batched algorithm runs. Based on the history up to batch $b-1$, it selects the number of samples to draw and receives the first $N_{b,i}$ observations.

The main advantage of this procedure is we can deal the samples independently from the algorithm. For example, we can apply Hoeffding inequality to the first $n_{b,i}$ samples of arm $i$ at batch $b$, even though in some cases algorithm does not draw $n_{b,i}$ samples.

*Proof of Lemma 15.* For $s \in \mathbb{N}$, let

$$\mathcal{V}_s := \left\{(v_{b,i})_{b\in[B],i\in[K]} \in \mathbb{N}^{B\times K} : \sum_{b,i} v_{b,i} \in [x+s, x+s+1)\right\},$$

and $\mathcal{V} := \bigcup_{s=\{0,1,2,\dots\}} \mathcal{V}_s$. Note that $\mathcal{V}_s$ is a finite set of combination and we can derive the following:

$$\begin{aligned}
|\mathcal{V}_s| &= \binom{x+s+BK-1}{BK-1} && (80)\\
&\leq \binom{x+s+BK}{BK} && (81)\\
&\leq \left(\frac{e(x+s+BK)}{BK}\right)^{BK} \quad \left(\text{By } \binom{n}{k} \leq (en/k)^k\right) && (82)
\end{aligned}$$

Let $\mathcal{N}(T) = (\{0,1,2,\dots,N\})^{BK}$. Let $\boldsymbol{n} = \{n_{b,i}\}_{b\in[B],i\in[K]} \in \mathcal{N}(T)$. Let $Q'_{b,i}$ be the empirical mean of the first $n_{b,i}$ samples drawn for arm $i$ in batch $b$. Then,

$$\begin{aligned}
&\mathbb{P}\left[\sum_{b\in[B]}\sum_{i\in[K]} N_{b,i}\frac{(Q_{b,i}-P_i)^2}{2} \geq x\right]\\
&\leq \sum_{\boldsymbol{n}\in\mathcal{N}(T)} \mathbb{P}\left[\bigcap_{b,i}\{N_{b,i}=n_{b,i}\}, \sum_b\sum_i N_{b,i}\frac{(Q_{b,i}-P_i)^2}{2} \geq x\right]\\
&\leq \sum_{\boldsymbol{v}\in\mathcal{V}}\sum_{\boldsymbol{n}\in\mathcal{N}(T)} \mathbb{P}\left[\bigcap_{b,i}\left\{N_{b,i}=n_{b,i}, n_{b,i}D\left(Q_{b,i}\|P_i\right) \in [v_{b,i}, v_{b,i}+1)\right\}\right]
\end{aligned}$$

$$
\begin{aligned}
&\leq \sum_{\boldsymbol{v}\in\mathcal{V}} \sum_{\boldsymbol{n}\in\mathcal{N}(T)} \mathbb{P}\left[\bigcap_{b,i} \left\{n_{b,i} D\left(Q'_{b,i}\|P_i\right) \in [v_{b,i}, v_{b,i}+1)\right\}\right] \\
&= \sum_{\boldsymbol{v}\in\mathcal{V}} \sum_{\boldsymbol{n}\in\mathcal{N}(T)} \prod_{b,i} \mathbb{P}\left[n_{b,i} D\left(Q'_{b,i}\|P_i\right) \in [v_{b,i}, v_{b,i}+1)\right] && \text{(independence)} \\
&= \sum_{s\geq 0} \sum_{\boldsymbol{v}\in\mathcal{V}_s} \sum_{\boldsymbol{n}\in\mathcal{N}(T)} \prod_{b,i} \mathbb{P}\left[n_{b,i} D\left(Q'_{b,i}\|P_i\right) \in [v_{b,i}, v_{b,i}+1)\right] \\
&\leq \sum_{s\geq 0} \sum_{\boldsymbol{v}\in\mathcal{V}_s} \sum_{\boldsymbol{n}\in\mathcal{N}(T)} \exp\left(-\sum_{b,i} v_{b,i}\right) && \text{(Hoeffding's inequality)} \\
&\leq \sum_{s\geq 0} \sum_{\boldsymbol{v}\in\mathcal{V}_s} \sum_{\boldsymbol{n}\in\mathcal{N}(T)} \exp\left(-x-s\right) && \text{(by } \boldsymbol{v}\in\mathcal{V}_s\text{)} \\
&\leq \sum_{s\geq 0} \sum_{\boldsymbol{v}\in\mathcal{V}_s} (N+1)^{KB} \exp\left(-x-s\right) \\
&\leq \sum_{s\geq 0} e^{BK} \left(\frac{x+s+BK}{BK}\right)^{BK} (N+1)^{KB} \exp\left(-x-s\right) && \text{(Eq. (82))} \\
&\leq \left(\frac{e}{BK}\right)^{BK} (N+1)^{KB} \sum_{s\geq 0} (x+s+BK)^{BK} \exp(-x-s) \\
&\leq \left(\frac{e}{BK}\right)^{BK} (N+1)^{KB} \sum_{s\geq 0} 3^{BK} (x^{BK}+s^{BK}+(BK)^{BK}) \exp(-x-s) \\
&\leq \left(\frac{e}{BK}\right)^{BK} (3(N+1))^{KB} \exp(-x) \sum_{s\geq 0} (x^{BK}+s^{BK}+(BK)^{BK}) \exp(-s) \\
&\leq \left(\frac{e}{BK}\right)^{BK} (3(N+1))^{KB} \exp(-x) O\left(x^{BK}+(BK)^{BK}\right) \\
&\leq e^{BK} (3(N+1))^{KB} \exp(-x) O(\exp(BK \max\{\log(x/BK), 0\})) \\
&\leq \exp\left(-x + BK \max\{\log(x/(BK)), 0\} + BK((\log N) + C_{\mathrm{cn}})\right)
\end{aligned}
$$

where $C_{\mathrm{cn}} > 0$ is a constant that does not depend on $T, N, K$. In the final transformation, we have used $(N+1)^{BK} = \exp(BK\log(N+1)), {C_{\mathrm{cn}}}^{BK} = \exp(O(BK))$ for any $C_{\mathrm{cn}} > 0$, $\sum_s s^{BK}\exp(-s) = O((BK)^{BK})$. By using $N = T/B$, we obtain Eq. (79).

By using Lemma 15 and simple calculation, we obtain the following lemma:

**Lemma 16.** Let $x = \frac{CR^{\mathrm{go}}T}{H_1(\boldsymbol{P})} = O(T/\log K)$. Assume that $N = \Omega(K(\log K)^3)$. Then,

$$
\mathbb{P}\left[\sum_{b\in[B]} \sum_{i\in[K]} N_{b,i} \frac{(Q_{b,i}-P_i)^2}{2} \geq x\right] \leq \exp\left(-C\left(1 - O\left(\frac{1}{\log K}\right)\right)\frac{R^{\mathrm{go}}T}{H_1(\boldsymbol{P})} + o(T)\right). \tag{83}
$$

*Proof of Lemma 16.* The exponent of Lemma 15 is

$$
-x + \frac{TK}{N}\max\left\{\log\left(\frac{xN}{TK}\right), 0\right\} + \frac{TK}{N}(\log N + C_{\mathrm{cn}}) = -\frac{CR^{\mathrm{go}}T}{H_1(\boldsymbol{P})} + o(T) + \frac{TK}{N}((\log N) + C_{\mathrm{cn}}) \tag{84}
$$

$$
= -\frac{CR^{\mathrm{go}}T}{H_1(\boldsymbol{P})} + o(T) + (1+C_{\mathrm{cn}})O\left(\frac{T}{(\log K)^2}\right) \tag{85}
$$

and by using $\frac{CR^{\mathrm{go}}T}{H_1(\boldsymbol{P})} = O(T/(\log K))$, we have Eq. (83). □

□

# F PROOF OF THEOREM 5

This section proves the approximate optimality of the allocation for $H_1$. We restate the theorem here for convenience.

**Theorem 5.** Allocation of Eq. (14) is a constant-factor approximation (Definition 4).

The core of Theorem 5 is the "stability lemma" that we introduce in Appendix F.1.

*Proof of Theorem 5.* Let

$$S(\boldsymbol{Q},\boldsymbol{P}) := \sum_i \frac{(Q_i-P_i)^2}{D_i(\boldsymbol{Q})} H_1(\boldsymbol{P}).$$

We have

$$\inf_{\boldsymbol{Q}\in\mathcal{Q}^K,\boldsymbol{P}:i^*(\boldsymbol{P})\notin i^*(\boldsymbol{Q})} H_1(\boldsymbol{P})\sum_{i\in[K]} w_i(\boldsymbol{Q})D(Q_i\|P_i) = \inf_{\boldsymbol{Q}\in\mathcal{Q}^K,\boldsymbol{P}:i^*(\boldsymbol{P})\notin i^*(\boldsymbol{Q})} \frac{S(\boldsymbol{Q},\boldsymbol{P})}{Z(\boldsymbol{Q})} \tag{86}$$

$$\geq \inf_{\boldsymbol{Q}\in\mathcal{Q}^K,\boldsymbol{P}:i^*(\boldsymbol{P})\notin i^*(\boldsymbol{Q})} \frac{C_{\mathrm{stbl}}}{\inf_{\boldsymbol{Q}} Z(\boldsymbol{Q})} \quad \text{(Lemma 17)}$$

$$\geq \inf_{\boldsymbol{Q}\in\mathcal{Q}^K,\boldsymbol{P}:i^*(\boldsymbol{P})\notin i^*(\boldsymbol{Q})} \frac{C_{\mathrm{stbl}}}{\log K+1} \quad \text{(by Lemma 22)} \tag{87}$$

Results in Carpentier and Locatelli (2016) implies the optimal rate is $R^{\mathrm{go}} = \Theta(\frac{1}{\log K})$ for $H_1(\boldsymbol{P})$, which matches the rate of Eq. (87). □

## F.1 Stability Lemma

**Lemma 17.** (Stability) Let $\boldsymbol{P},\boldsymbol{Q}\in\mathbb{R}^K$ be two arbitrary vectors such that $\boldsymbol{P}$ has unique best arm $i^*(\boldsymbol{P})$ and $i^*(\boldsymbol{P})\notin i^*(\boldsymbol{Q})$. Then, there exists a universal constant $C_{\mathrm{stbl}}>0$ such that $S(\boldsymbol{Q},\boldsymbol{P})\geq C_{\mathrm{stbl}}$.

*Proof of Lemma 17.* **Minimize $P$ given $Q$:** Let

$$S_{2,j}(\boldsymbol{Q}) := \min_{\boldsymbol{P}:i^*(\boldsymbol{P})=j} S(\boldsymbol{Q},\boldsymbol{P}). \tag{88}$$

We assume the uniqueness of the best arm in $\boldsymbol{Q}$ and $\boldsymbol{P}$. The case where the empirical best arm is not unique is discussed in Appendix F.4. Without loss of generality, assume that $Q_1>Q_2\geq\cdots\geq Q_K$. Let $\Delta_i^Q = Q_1-Q_i$. Let $j=\arg\max_i P_i$ and $\Delta_i = P_j-P_i$. By definition, $j\neq 1$.

We have

$$\begin{aligned}
&S_{2,j}(\boldsymbol{Q})\\
&= \min_{\boldsymbol{P}:i^*(\boldsymbol{P})=j}\left(\sum_i \frac{(Q_i-P_i)^2}{D_i(\boldsymbol{Q})}H_1(\boldsymbol{P})\right)\\
&= \min_{\boldsymbol{P}:i^*(\boldsymbol{P})=j}\left(\sum_i \frac{(Q_i-P_i)^2}{D_i(\boldsymbol{Q})}\sum_{i\neq j}\frac{1}{(P_j-P_i)^2}\right) && \text{(By definition)}\\
&= \min_{\boldsymbol{P}:i^*(\boldsymbol{P})=j,P_j\geq Q_j,\forall_{i\neq j}P_i\leq Q_i}\left(\sum_i \frac{(Q_i-P_i)^2}{D_i(\boldsymbol{Q})}\sum_{i\neq j}\frac{1}{(P_j-P_i)^2}\right) && \text{( Claim 2)}\\
&= \min_{\boldsymbol{P}:i^*(\boldsymbol{P})=j,P_j\geq Q_j,\forall_{i<j}\,Q_i-(P_j-Q_j)\leq P_i\leq Q_i,\forall_{i>j}\,P_i\leq Q_i}\left(\sum_i \frac{(Q_i-P_i)^2}{D_i(\boldsymbol{Q})}\sum_{i\neq j}\frac{1}{(P_j-P_i)^2}\right) && \text{(Claim 3)}
\end{aligned}$$

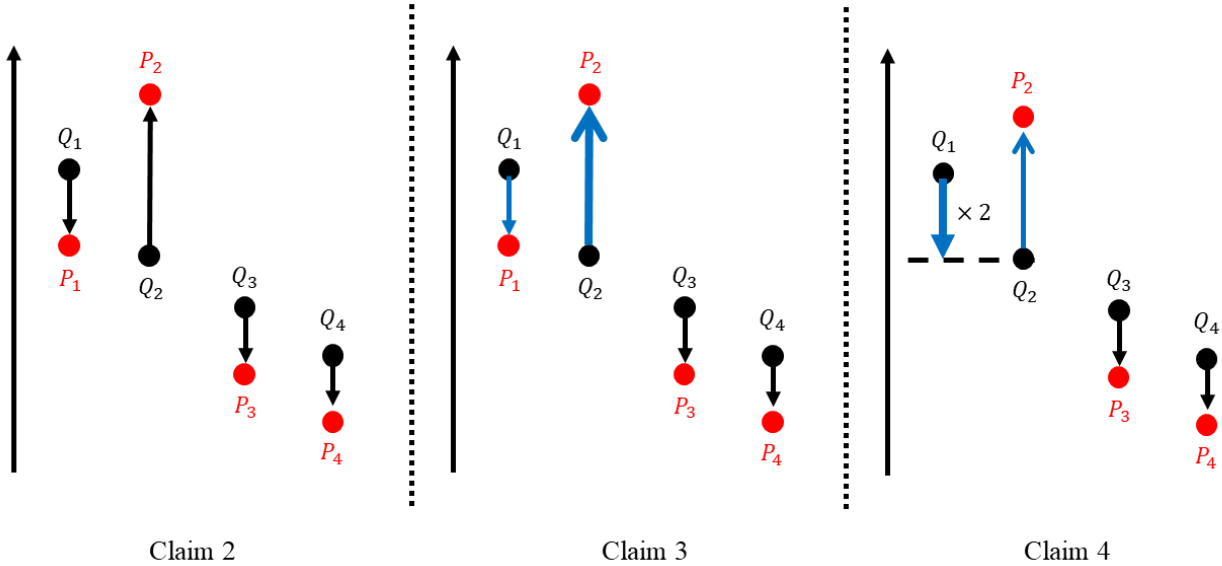



Figure 3: Illustration of the added constraints by Claims 2–4. We use a four-armed instance where the best arm in $\boldsymbol{P}$ is arm $j = 2$. Claim 2 (left) states that $P_2 > Q_2$ and $P_i \leq Q_i$ for $i \neq 2$. Claims 3 (middle) states that bold blue arrow of arm 2 is longer than the standard-size blue arrow of arm 1. Claim 4 (right) states that blue arrow of arm 2 is at most twice as the blue arrow of arm 1.

$$\geq \frac{1}{4} \min_{\boldsymbol{P}: i^*(\boldsymbol{P})=j, Q_j \leq P_j \leq 2Q_1 - Q_j, \forall_{i<j}\, Q_i - (P_j - Q_j) \leq P_i \leq Q_i, \forall_{i>j}\, P_i \leq Q_i} \left( \sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})} \sum_{i \neq j} \frac{1}{(P_j - P_i)^2} \right) \qquad \text{(Claim 4)}$$

$$= \Omega(1) \qquad \text{(Claim 5)}$$

and the proof of Lemma 17 is completed. □

### F.2 Proof of Claims

**Claim 2.** $\min_{\boldsymbol{P}: i^*(\boldsymbol{P})=j} S(\boldsymbol{Q}, \boldsymbol{P}) = \min_{\boldsymbol{P}: i^*(\boldsymbol{P})=j, P_j \geq Q_j, \forall_{i \neq j} P_i \leq Q_i} S(\boldsymbol{Q}, \boldsymbol{P})$

*Proof.* For a $\boldsymbol{P} \in \{\boldsymbol{P} : i^*(\boldsymbol{P}) = j, P_j < Q_j\}$, define $\boldsymbol{P}'$ as

$$P'_s = \begin{cases} Q_j & s = j \\ P_s & \text{Otherwise} \end{cases}.$$

Then, one can check $i^*(\boldsymbol{P}') = j$ and $S(\boldsymbol{Q}, \boldsymbol{P}) > S(\boldsymbol{Q}, \boldsymbol{P}')$. Therefore, we can assume that the minimum argument should satisfy $P_j \geq Q_j$. Similarly, we can prove that the optimal argument satisfies $P_i \leq Q_i$ for all $i \neq j$. □

**Claim 3.**

$$\min_{\boldsymbol{P}: i^*(\boldsymbol{P})=j, P_j \geq Q_j, \forall_{i \neq j} P_i \leq Q_i} S(\boldsymbol{Q}, \boldsymbol{P}) = \min_{\boldsymbol{P}: i^*(\boldsymbol{P})=j, P_j \geq Q_j, \forall_{i<j}\, Q_i - (P_j - Q_j) \leq P_i \leq Q_i, \forall_{i>j}\, P_i \leq Q_i} S(\boldsymbol{Q}, \boldsymbol{P})$$

*Proof.* For $\boldsymbol{P} \in \{\boldsymbol{P} : i^*(\boldsymbol{P}) = j, P_j \geq Q_j, \forall_{i \neq j} P_i \leq Q_i\}$ suppose that there exists a coordinate $t < j$ such that

$Q_t - P_t \geq P_j - Q_j$. Then, define $P'$ as

$$P'_s = \begin{cases} Q_j + \frac{Q_t - P_t + P_j - Q_j}{2} & s = j \\ Q_t - \frac{Q_t - P_t + P_j - Q_j}{2} & s = t \\ P_s & \text{Otherwise} \end{cases}$$

. Then, since $Q_t - P_t \geq P_j - Q_j$, $P'_j > Q_j + P_j - Q_j = P_j$. In addition, $P'_j - P'_t = P_j - P_t$. This implies $H_1(\boldsymbol{P}') < H_1(\boldsymbol{P})$. Now let's compare $\sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})}$ and $\sum_i \frac{(Q_i - P'_i)^2}{D_i(\boldsymbol{Q})}$.

$$\begin{aligned}
\sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})} - \sum_i \frac{(Q_i - P'_i)^2}{D_i(\boldsymbol{Q})} &= \frac{(Q_j - P_j)^2 - (Q_j - P'_j)^2}{D_j(\boldsymbol{Q})} + \frac{(Q_t - P_t)^2 - (Q_t - P'_t)^2}{D_t(\boldsymbol{Q})} \\
&= \frac{(P'_j - P_j)(2Q_j - P_j - P'_j)}{D_j(\boldsymbol{Q})} + \frac{(P'_t - P_t)(2Q_t - P_t - P'_t)}{D_t(\boldsymbol{Q})} \\
&= \frac{\frac{Q_t - P_t + Q_j - P_j}{2} \cdot \frac{3Q_j - 3P_j - Q_t + P_t}{2}}{D_j(\boldsymbol{Q})} + \frac{\frac{Q_t - P_t + Q_j - P_j}{2} \cdot \frac{3Q_t - 3P_t + P_j - Q_j}{2}}{D_t(\boldsymbol{Q})} \\
&\geq \frac{\frac{Q_t - P_t + Q_j - P_j}{2} \cdot \frac{3Q_j - 3P_j - Q_t + P_t}{2}}{D_j(\boldsymbol{Q})} + \frac{\frac{Q_t - P_t + Q_j - P_j}{2} \cdot \frac{3Q_t - 3P_t + P_j - Q_j}{2}}{D_j(\boldsymbol{Q})} \\
&\qquad (D_j(\boldsymbol{Q}) > D_t(\boldsymbol{Q}) \text{ and numerator is positive}) \\
&\geq \frac{\frac{Q_t - P_t + Q_j - P_j}{2} \cdot \frac{2(Q_j - P_j) - 2(Q_t - P_t)}{2}}{D_j(\boldsymbol{Q})} > 0, \\
&\qquad (\text{by assumption } Q_t - P_t \geq P_j - Q_j)
\end{aligned}$$

and thus $\sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})} > \sum_i \frac{(Q_i - P'_i)^2}{D_i(\boldsymbol{Q})}$. Combined with $H_1(\boldsymbol{P}') < H_1(\boldsymbol{P})$, we have $S(\boldsymbol{Q}, \boldsymbol{P}') < S(\boldsymbol{Q}, \boldsymbol{P})$. □

**Claim 4.**

$$\begin{aligned}
&\min_{\boldsymbol{P}: i^*(\boldsymbol{P}) = j, Q_j \leq P_j, \forall_{i<j}\, Q_i - (P_j - Q_j) \leq P_i \leq Q_i, \forall_{i>j}\, P_i \leq Q_i} S(\boldsymbol{Q}, \boldsymbol{P}) \\
\geq &\frac{1}{4} \min_{\boldsymbol{P}: i^*(\boldsymbol{P}) = j, Q_j \leq P_j \leq 2Q_1 - Q_j, \forall_{i<j}\, Q_i - (P_j - Q_j) \leq P_i \leq Q_i, \forall_{i>j}\, P_i \leq Q_i} S(\boldsymbol{Q}, \boldsymbol{P})
\end{aligned}$$

*Proof.* Let $S_{2,j}(\boldsymbol{P}; \boldsymbol{Q}) := \sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})} H_1(\boldsymbol{P})$, and let $\boldsymbol{P}^0 := \arg\min_{\boldsymbol{P}: i^*(\boldsymbol{P}) = j} S_{2,j}(\boldsymbol{P})$. If $\boldsymbol{P}^0_j \leq 2Q_1 - Q_j$, then the Claim holds trivially. Suppose that $\boldsymbol{P}^0_j > 2Q_1 - Q_j$, and let $E := \boldsymbol{P}^0_j - Q_1 > Q_1 - Q_j$. Now, let's define $\boldsymbol{P}'$ as

$$P'_s = \begin{cases} Q_1 + \Delta^Q_j & s = j \\ Q_s - (Q_s - P_s) \cdot \frac{\Delta^Q_j}{E} & \text{Otherwise} \end{cases}$$

Let's check how this operation changes the value $S_{2,j}(\boldsymbol{P}, \boldsymbol{Q})$, especially for each $W^1_{2,j}(\boldsymbol{P}; \boldsymbol{Q}) := \sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})}$ and $W^2_{2,j}(\boldsymbol{P}; \boldsymbol{Q}) := \sum_i \frac{1}{(P_j - P_i)^2}$ (Note that $S_{2,j}(\boldsymbol{P}; \boldsymbol{Q}) = W^1_{2,j}(\boldsymbol{P}; \boldsymbol{Q}) \cdot W^2_{2,j}(\boldsymbol{P}; \boldsymbol{Q})$).

**First Term** $W^1_{2,j}(\boldsymbol{P}; \boldsymbol{Q})$ Since $E > \Delta^Q_j$, $Q_s - P'_s < Q_s - P_s$ for all $s \neq j$, it holds that $W^1_{2,j}(\boldsymbol{P}'; \boldsymbol{Q}) = \sum_i \frac{(Q_i - P'_i)^2}{D_i(Q)} \leq \sum_i \frac{(Q_i - P^0_i)^2}{D_i(Q)} = W^1_{2,j}(\boldsymbol{P}^0; \boldsymbol{Q})$. Namely, this operation decreases $W^1_{2,j}$. In particular, for each $i \neq j$, $Q_s - P'_s = (Q_s - P_s) \cdot \frac{\Delta^Q_j}{E}$, and $P'_j - Q_j = 2\Delta^Q_j = \frac{2\Delta^Q_j}{E + \Delta^Q_j} \cdot (P^0_j - Q_j) \leq \frac{2\Delta^Q_j}{E} \cdot (P^0_j - Q_j)$. Therefore, $W^1_{2,j}(\boldsymbol{P}'; \boldsymbol{Q}) \leq \frac{(\Delta^Q_j)^2}{E^2} \left( \sum_{i \neq j} \frac{P^0_i - Q_i}{D_i(\boldsymbol{Q})} \right) + \frac{4(\Delta^Q_j)^2}{E^2} \frac{P^0_j - Q_j}{D_j(\boldsymbol{Q})} < \frac{4(\Delta^Q_j)^2}{E^2} W^1_{2,j}(\boldsymbol{P}^0; \boldsymbol{Q})$.

**Second Term** $W^2_{2,j}(\boldsymbol{P}; \boldsymbol{Q})$ By Claim 2, $P^0_j > P'_j > Q_1 > P'_s > P^0_s$ for all $s \neq j$ so $W^2_{2,j}(\boldsymbol{P}^0; \boldsymbol{Q}) = \sum_{i \neq j} \frac{1}{(P^0_j - P^0_i)^2} \leq \sum_{i \neq j} \frac{1}{(P'_j - P'_i)^2} = W^2_{2,j}(\boldsymbol{P}'; \boldsymbol{Q})$, which means this operation increase the function $W^2_{2,j}$. More specifically, for each $i \neq j$,

$$(P'_j - P'_i) = Q_1 + E \cdot \frac{\Delta^Q_j}{E} - Q_s + (Q_s - P^0_s) \cdot \frac{\Delta^Q_j}{E}$$

$$
\begin{aligned}
&= E \cdot \frac{\Delta_j^Q}{E} + \Delta_s^Q + (Q_s - P_s^0) \cdot \frac{\Delta_j^Q}{E} \\
&\geq E \cdot \frac{\Delta_j^Q}{E} + \Delta_s^Q \frac{\Delta_j^Q}{E} + (Q_s - P_s^0) \cdot \frac{\Delta_j^Q}{E} && (\tfrac{\Delta_j^Q}{E} \leq 1) \\
&= \frac{\Delta_j^Q}{E} \cdot \left(E + \Delta_s^Q + Q_s - P_s^0\right) = \frac{\Delta_j^Q}{E} \cdot \left(P_j^0 - P_s^0\right). && (E = P_j^0 - Q_1, \Delta_s^Q + Q_s = Q_1)
\end{aligned}
$$

Therefore, we can conclude $W_{2,j}^2(\boldsymbol{P}'; \boldsymbol{Q}) \leq W_{2,j}^2(\boldsymbol{P}^0; \boldsymbol{Q}) \cdot \frac{E^2}{(\Delta_j^Q)^2}$.

Overall, $S_{2,j}(\boldsymbol{Q}) = W_{2,j}^1(\boldsymbol{P}^0; \boldsymbol{Q}) \cdot W_{2,j}^2(\boldsymbol{P}^0; \boldsymbol{Q}) \geq \frac{E^2}{4(\Delta_j^Q)^2} W_{2,j}^1(\boldsymbol{P}'; \boldsymbol{Q}) \cdot \frac{(\Delta_j^Q)^2}{E^2} W_{2,j}^2(\boldsymbol{P}'; \boldsymbol{Q}) = \frac{1}{4} S_{2,j}(\boldsymbol{P}'; \boldsymbol{Q})$. Also, one can check that $P'$ satisfies the constraint of the second optimization. Therefore, in both cases of $\boldsymbol{P}^0$, the claim holds. □

**Claim 5.** $\min_{\boldsymbol{P}: i^*(\boldsymbol{P}) = j, Q_j \leq P_j \leq 2Q_1 - Q_j, \forall_{i<j}\, Q_i - (P_j - Q_j) \leq P_i \leq Q_i, \forall_{i>j}\, P_i \leq Q_i} S(\boldsymbol{Q}, \boldsymbol{P}) = \Omega(1)$

*Proof.* Let's focus on $W_{2,j}^1(\boldsymbol{P}; \boldsymbol{Q})$ first.

$$
\begin{aligned}
W_{2,j}^1(\boldsymbol{P}; \boldsymbol{Q}) &= \sum_l \frac{(Q_l - P_l)^2}{D_l(\boldsymbol{Q})} \\
&\geq \sum_{l \geq j} \frac{(Q_l - P_l)^2}{D_l(\boldsymbol{Q})} \\
&\geq \frac{(\Delta_j^Q)^2}{D_j(\boldsymbol{Q})} + \sum_{l > j} \frac{(P_l - Q_l)^2}{D_l(\boldsymbol{Q})} && (\text{for } l = j, \text{ we can use } P_j \geq Q_1)
\end{aligned}
$$

Next, we can lower bound $W_{2,j}^2(\boldsymbol{P}; \boldsymbol{Q})$ as follows:

$$
\begin{aligned}
W_{2,j}^2(\boldsymbol{P}; \boldsymbol{Q}) &= \sum_{i \neq j} \frac{1}{(P_i - P_j)^2} \\
&= \sum_{i<j} \frac{1}{(P_i - P_j)^2} + \sum_{i>j} \frac{1}{(P_i - P_j)^2} \\
&\geq \sum_{i<j} \frac{1}{9(\Delta_j^Q)^2} + \sum_{i>j} \frac{1}{(P_i - P_j)^2} && (\text{for } i < j,\ P_j - P_i = P_j - Q_1 + Q_1 - Q_i + Q_i - P_i \leq 3\Delta_j) \\
&\geq \frac{j}{18(\Delta_j^Q)^2} + \sum_{i>j} \frac{1}{(P_i - P_j)^2} && (j \geq 2 \to j - 1 \geq j/2)
\end{aligned}
$$

Now, let $M_i := |Q_i - P_i|$ for notational convenience.

$$
\begin{aligned}
&S_{2,j}(\boldsymbol{P}; \boldsymbol{Q}) \\
&= W_{2,j}^1(\boldsymbol{P}; \boldsymbol{Q}) \cdot W_{2,j}^2(\boldsymbol{P}; \boldsymbol{Q}) \\
&\geq \left( \frac{(\Delta_j^Q)^2}{D_j(\boldsymbol{Q})} + \sum_{l>j} \frac{(P_l - Q_l)^2}{D_l(\boldsymbol{Q})} \right) \cdot \left( \frac{j}{18(\Delta_j^Q)^2} + \sum_{l>j} \frac{1}{(P_l - P_j)^2} \right) \\
&\geq \frac{1}{D_j(\boldsymbol{Q})} \left( 1 + \sum_{l>j} \frac{M_l^2}{(\Delta_l^Q)^2} \right) \cdot \left( \frac{j}{18} + \sum_{l>j} \frac{(\Delta_j^Q)^2}{(M_j + Q_j - Q_l + M_l)^2} \right) && (\text{Lemma 19}) \\
&\geq \frac{1}{D_j(\boldsymbol{Q})} \left( 1 + \sum_{l>j} \frac{M_l^2}{(\Delta_l^Q)^2} \right) \cdot \left( \frac{j}{18} + \sum_{l>j} \frac{(\Delta_j^Q)^2}{(\Delta_j^Q + \Delta_l^Q + M_l)^2} \right) && (M_j \leq 2\Delta_j)
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{D_j(\boldsymbol{Q})}\left(1+\sum_{l>j}\frac{M_l^2}{(\Delta_l^Q)^2}\right)\cdot\left(\frac{j}{18}+\sum_{l>j,M_l\geq\Delta_l}\frac{(\Delta_j^Q)^2}{(\Delta_j^Q+\Delta_l^Q+M_l)^2}+\sum_{l>j,M_l<\Delta_l}\frac{(\Delta_j^Q)^2}{(\Delta_j^Q+\Delta_l^Q+M_l)^2}\right)\\
&\geq \frac{1}{D_j(\boldsymbol{Q})}\left(1+\sum_{l>j}\frac{M_l^2}{(\Delta_l^Q)^2}\right)\cdot\left(\frac{j}{18}+\sum_{l>j,M_l\geq\Delta_l}\frac{(\Delta_j^Q)^2}{9M_l^2}+\sum_{l>j,M_l<\Delta_l}\frac{(\Delta_j^Q)^2}{9(\Delta_l^Q)^2}\right) \quad \text{(by } j>l \text{ implies } \Delta_j^Q\leq\Delta_l^Q\text{)}\\
&\geq \frac{1}{D_j(\boldsymbol{Q})}\left(1+\sum_{l>j}\frac{M_l^2}{(\Delta_l^Q)^2}\right)\cdot\left(\frac{j}{18}+\sum_{l>j,M_l\geq\Delta_l}\frac{(\Delta_j^Q)^2}{9M_l^2}\right)+\frac{1}{D_j(\boldsymbol{Q})}\sum_{l>j,M_l<\Delta_l}\frac{(\Delta_j^Q)^2}{9(\Delta_l^Q)^2}\\
&\geq \frac{1}{D_j(\boldsymbol{Q})}\left(1+\sum_{l>j,M_l\geq\Delta_l}\frac{M_l^2}{(\Delta_l^Q)^2}\right)\cdot\left(\frac{j}{18}+\sum_{l>j,M_l\geq\Delta_l}\frac{(\Delta_j^Q)^2}{9M_l^2}\right)+\frac{1}{D_j(\boldsymbol{Q})}\sum_{l>j,M_l<\Delta_l}\frac{(\Delta_j^Q)^2}{9(\Delta_l^Q)^2}\\
&\geq \frac{1}{D_j(\boldsymbol{Q})}\left(\sqrt{\frac{j}{18}}+\sum_{l>j,M_l\geq\Delta_l}\frac{\Delta_j^Q}{3\Delta_l^Q}\right)^2+\frac{1}{D_j(\boldsymbol{Q})}\sum_{l>j,M_l<\Delta_l}\frac{(\Delta_j^Q)^2}{9(\Delta_l^Q)^2} \quad \text{(Cauchy)}\\
&\geq \frac{1}{D_j(\boldsymbol{Q})}\left(\frac{j}{18}+\sum_{l>j}\frac{(\Delta_j^Q)^2}{9(\Delta_l^Q)^2}\right) \quad ((a+b)^2>a^2+b^2 \text{ when } a,b>0)\\
&= \Omega(1) \quad \text{(From Lemma 20 and 21, } D_j(\boldsymbol{Q})=\Theta\left(j+\textstyle\sum_{l>j}\frac{(\Delta_j^Q)^2}{(\Delta_l^Q)^2}\right)\text{)}
\end{aligned}
$$

□

### F.3 Other Lemmas

**Lemma 18.** For $s<j$, $D_s(\boldsymbol{P})<D_j(\boldsymbol{P})$. For $u>j$, $D_u(\boldsymbol{P})<\frac{\Delta_u^2}{\Delta_j^2}D_j(\boldsymbol{P})$.

*Proof.* Now for $s<j$, $\Delta_s<\Delta_j$, and therefore

$$
\begin{aligned}
D_j(\boldsymbol{P})-D_s(\boldsymbol{P}) &= (\Delta_j+x)^2\sum_{z\neq j}\frac{1}{(\Delta_z+x)^2}-(\Delta_s+x)^2\sum_{z\neq s}\frac{1}{(\Delta_z+x)^2}\\
&\geq (\Delta_j+x)^2\sum_{z\neq j}\frac{1}{(\Delta_z+x)^2}-(\Delta_j+x)^2\sum_{z\neq s}\frac{1}{(\Delta_z+x)^2}\\
&= (\Delta_j+x)^2\left(\frac{1}{(\Delta_s+x)^2}-\frac{1}{(\Delta_j+x)^2}\right)>0
\end{aligned}
$$

Now for $u>j$, $\Delta_u>\Delta_j$, and therefore

$$
\begin{aligned}
\frac{2\Delta_u^2}{\Delta_j^2}D_j(\boldsymbol{P})-D_u(\boldsymbol{P}) &= \frac{2\Delta_u^2}{\Delta_j^2}(\Delta_j+x)^2\sum_{z\neq j}\frac{1}{(\Delta_z+x)^2}-(\Delta_u+x)^2\sum_{z\neq u}\frac{1}{(\Delta_z+x)^2}\\
&\geq 2(\Delta_u+x)^2\sum_{z\neq j}\frac{1}{(\Delta_z+x)^2}-(\Delta_u+x)^2\sum_{z\neq u}\frac{1}{(\Delta_z+x)^2}\\
&> (\Delta_u+x)^2\left(\frac{1}{(\Delta_j+x)^2}-\frac{2}{(\Delta_u+x)^2}+\frac{1}{x^2}\right)>0
\end{aligned}
$$

□

**Lemma 19.** For $l>j$, $\frac{D_j(\boldsymbol{Q})}{D_l(\boldsymbol{Q})}\geq\frac{1}{4}\frac{\Delta_j^2}{\Delta_l^2}$

*Proof.* First, note that for $a,b,c,d>0$, $\frac{a+c}{b+d}\geq\min\left(\frac{a}{b},\frac{c}{d}\right)$. To see this, WLOG $\frac{a}{b}>\frac{c}{d}$, which means $ad>bc$. Then

$$
\frac{(a+c)}{b+d}-\frac{c}{d}=\frac{ad+cd-bc-cd}{(b+d)d}>0 \tag{89}
$$

Now, from Lemma 20 and 21, $D_j(\boldsymbol{Q}) \geq j + \sum_{s>j} \frac{\Delta_j^2}{\Delta_s^2}$ and $D_l(\boldsymbol{Q}) \leq 4(l + \sum_{s>l} \frac{\Delta_l^2}{\Delta_s^2})$. Therefore,

$$\begin{aligned}
\frac{D_j(\boldsymbol{Q})}{D_l(\boldsymbol{Q})} &\geq \frac{1}{4} \cdot \frac{j + \sum_{s>j} \frac{\Delta_j^2}{\Delta_s^2}}{l + \sum_{s>l} \frac{\Delta_l^2}{\Delta_s^2}} \\
&= \frac{1}{4} \cdot \frac{\overbrace{\left(j + \sum_{s:j<s\leq l} \frac{\Delta_j^2}{\Delta_s^2}\right)}^{A} + \overbrace{\left(\sum_{s>l} \frac{\Delta_j^2}{\Delta_s^2}\right)}^{C}}{\underbrace{l}_{B} + \underbrace{\sum_{s>l} \frac{\Delta_l^2}{\Delta_s^2}}_{D}} \\
&\geq \min\left(\frac{\left(j + \sum_{s:j<s\leq l} \frac{\Delta_j^2}{\Delta_s^2}\right)}{l}, \frac{\sum_{s>l} \frac{\Delta_j^2}{\Delta_s^2}}{\sum_{s>l} \frac{\Delta_l^2}{\Delta_s^2}}\right) && \text{(Eq. (89))} \\
&\geq \min\left(\frac{\left(j + (l-j) \cdot \frac{\Delta_j^2}{\Delta_l^2}\right)}{l}, \frac{\Delta_j^2}{\Delta_l^2}\right) \geq \frac{\Delta_j^2}{\Delta_l^2} && (\Delta_s \leq \Delta_l \text{ for } s \leq l)
\end{aligned}$$

and the proof ends. □

**Lemma 20.** Suppose that $P_1 \geq P_2 \geq ... \geq P_K$. Then $D_i(\boldsymbol{P}) \geq i + \sum_{j>i} \frac{\Delta_i^2}{\Delta_j^2}$ .

*Proof.*

$$\begin{aligned}
D_i(\boldsymbol{P}) \geq \inf_{x>0} (\Delta_i + x)^2 \sum_{j\neq i} \frac{1}{(\Delta_j + x)^2} &\geq \sum_{j\neq i} \inf_{x>0} \frac{(\Delta_i + x)^2}{(\Delta_j + x)^2} \\
&\geq \sum_{j\neq i} \min\{1, \frac{\Delta_i^2}{\Delta_j^2}\} \\
&= i + \sum_{j>i} \frac{\Delta_i^2}{\Delta_j^2}.
\end{aligned}$$

□

**Lemma 21.** Suppose that $P_1 \geq P_2 \geq ... \geq P_K$. Then $D_i(\boldsymbol{P}) \leq 4\left(i + \sum_{j>i} \frac{\Delta_i^2}{\Delta_j^2}\right)$ .

*Proof.*

$$\begin{aligned}
D_i(\boldsymbol{P}) &= (\Delta_i + \Delta_i)^2 \sum_{j\neq i} \frac{1}{(\Delta_j + \Delta_i)^2} \\
&= 4 \cdot \left(\sum_{j>i} \frac{\Delta_i^2}{(\Delta_i + \Delta_j)^2} + \sum_{j<i} \frac{\Delta_i^2}{(\Delta_i + \Delta_j)^2}\right) \leq 4 \cdot \left(\sum_{j>i} \frac{\Delta_i^2}{\Delta_j^2} + i\right)
\end{aligned}$$

.

□

**Lemma 22.** For any $\boldsymbol{P}$, $Z(\boldsymbol{P}) = \Omega(1)$ and $Z(\boldsymbol{P}) = O(\log K)$.

*Proof.* Lemma 21 that implies $1/D_i \geq 1/(4K)$, from which $Z(\boldsymbol{P}) = \Omega(1)$ directly follows.

By Lemma 20, we have

$$Z(\boldsymbol{P}) := \sum_i \frac{1}{D_i} \leq \sum_i \frac{1}{i + \sum_{j>i} \frac{\Delta_i^2}{\Delta_j^2}} \leq \sum_i \frac{1}{i} \leq \log(K) + 1. \tag{90}$$

□

### F.4 Proof of Lemma 17 for multiple empirical best arms

Let $\boldsymbol{Q}$ be any empirical means with multiple best arm $|i^*(\boldsymbol{Q})| \geq 2$. Let $\boldsymbol{P} : i^*(\boldsymbol{P}) \in i^*(\boldsymbol{Q})$ be any true distribution with unique best arm (Note: if $i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})$, then we can just remove all best arm ties in $i^*(\boldsymbol{Q})$ and just keep one, which is trivial). Our goal here is to show

$$\sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})} H(\boldsymbol{P}) \geq C \sum_i \frac{(Q'_i - P_i)^2}{D_i(\boldsymbol{Q})} H(\boldsymbol{P}) \tag{91}$$

for some universal constant $C > 0$. Where $\boldsymbol{Q}', \boldsymbol{P}$ are distributions with unique best arm and $i^*(\boldsymbol{Q}') \neq i^*(\boldsymbol{P})$.

Without loss of generality, we assume $Q_1 = Q_2 \geq Q_3 \geq \cdots \geq Q_K$ and $P_1 > \{P_2, P_3, \ldots, P_K\}$. Let $j > 2$ be the first index such that $Q_j < Q_1$.

Moreover, let $\boldsymbol{Q}'$ be the distribution such that

$$Q'_i = \begin{cases} Q_i & i \neq 2 \\ Q_2 + \varepsilon & i = 2 \end{cases}, \tag{92}$$

for a small enough constant $\epsilon > 0$. Then $i^*(\boldsymbol{Q}') = 2$ is the unique best arm and for any $i$, we have

$$(Q_i - P_i)^2 - (Q'_i - P_i)^2 = O(\varepsilon)$$

Moreover, by using Lemma 20 and Lemma 21, we have the followings. Namely, for any $i \geq j$, we have

$$D_i(\boldsymbol{Q}) \leq 4 \left( j + \sum_{l>j} \frac{(\Delta_i^Q)^2}{(\Delta_l^Q)^2} \right) \tag{93}$$

and

$$D_i(\boldsymbol{Q}') \geq j + \sum_{l>j} \frac{(\Delta_i^Q)^2}{(\Delta_l^Q + \varepsilon)^2}. \tag{94}$$

For any $i < j$, we have

$$D_i(\boldsymbol{Q}) = D_j(\boldsymbol{Q}) \leq 4 \left( j - 1 + \sum_{l \geq j} \frac{(\Delta_j^Q)^2}{(\Delta_l^Q)^2} \right) \tag{95}$$

and

$$D_i(\boldsymbol{Q}') = D_{j-1}(\boldsymbol{Q}') \geq j - 1 + \sum_{l \geq j} \frac{(\Delta_j^Q)^2}{(\Delta_l^Q + \varepsilon)^2} \tag{96}$$

which together imply

$$\sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})} H(\boldsymbol{P}) \geq \frac{1}{4} \sum_i \frac{(Q'_i - P_i)^2}{D_i(\boldsymbol{Q}')} H(\boldsymbol{P}) - O(\varepsilon) \tag{97}$$

Since $1 = i^*(\boldsymbol{P}) \neq i^*(\boldsymbol{Q}') = 2$, we obtained the desired $\boldsymbol{Q}', \boldsymbol{P}$ such that Eq. (91) holds for $C > 1/4 - \varepsilon$ with any $\varepsilon > 0$.

# G TIGHTER ANALYSIS OF SR

## G.1 Upper Bound on the Probability of Error of SR

In this section, we show the rate of SR derived in Audibert et al. (2010) is tight up to a constant factor. The next section shows the exact constant factor.

We assume that $P_1 > P_2 \geq \cdots \geq P_K$ and consider Gaussian rewards with unit variance. The original SR paper (Audibert et al., 2010) considered rewards over $[0, 1]$. If we run the same argument as the SR paper for unit-variance Gaussian rewards instead of rewards over $[0, 1]$, we obtain

$$\limsup_{T\to\infty} \frac{1}{T} \log \mathrm{PoE}(\boldsymbol{P}) \leq -\frac{1}{H_2(\boldsymbol{P})\overline{\log} K} \tag{98}$$

for

$$\overline{\log}(K) = \frac{1}{2} + \sum_{i=2}^{K} \frac{1}{i} \tag{99}$$

$$H_2(\boldsymbol{P}) = 4 \max_{j\in[K]} j\Delta_j^{-2} \tag{100}$$

where there is an extra factor of 4. This is because the Hoeffding inequality $\mathbb{P}[Q_{i,n} \geq P_i + x] \leq \mathrm{e}^{-2nx^2}$ $(x \geq 0)$ needs to be replaced with the Chernoff bound $\mathbb{P}[Q_{i,n} \geq P_i + x] \leq \mathrm{e}^{-nx^2/2}$.

Note that this Eq. (98) is only an upper bound on the probability of error of SR, and there remains the possibility that the probability of error of SR is smaller. In the next section, we show that this rate SR is tight up to a factor of 2.

## G.2 Tight Bound on the Probability of Error of SR

We first introduce the modified mean of top-$j$ arms and a complexity $H_3(\boldsymbol{P})$ for the lower bound of the probability of error. We then show that $H_3$ is tight up to a factor of 2.

Define a subset of top-$j$ arms consisting of the best arm and other arms with means at most $P$ as

$$\mathcal{S}^{(j)}(P) = \{1\} \cup \{i \in \{2, 3, \ldots, j\} : P_i \leq P\}. \tag{101}$$

The *modified mean* of top-$j$ arms is defined by

$$\bar{P}^{\mathrm{mod}}(j) = \frac{1}{|\mathcal{S}^{(j)}(\bar{P}^{\mathrm{mod}}(j))|} \sum_{i\in\mathcal{S}^{(j)}(\bar{P}^{\mathrm{mod}}(j))} P_i. \tag{102}$$

Though this expression is written in an implicit form, we can easily see that $\bar{P}^{\mathrm{mod}}(k)$ satisfying this expression uniquely exists and it satisfies $\bar{P}^{\mathrm{mod}}(j) \in [P_j, P_1]$.

Define

$$\Delta_i^{(j)} = \begin{cases} P_1 - \bar{P}^{\mathrm{mod}}(j) & i = 1 \\ (\bar{P}^{\mathrm{mod}}(j) - P_i)_+ & i = 2, 3, \ldots, j \end{cases} \tag{103}$$

and

$$H_3(\boldsymbol{P})^{-1} = \min_{j\in\{2,3,\ldots,K\}} \frac{1}{2j} \sum_{i=1}^{j} (\Delta_i^{(j)})^2, \tag{104}$$

that is,

$$H_3(\boldsymbol{P}) = \max_{j\in\{2,3,\ldots,K\}} \frac{2j}{\sum_{i=1}^{j} (\Delta_i^{(j)})^2}. \tag{105}$$

We will show that $H_3$ is the tight exponent of the probability of error of SR.

**Theorem 23** (Restatement of Theorem 7)**.** The error probability of successive reject for Gaussian arms satisfies

$$\lim_{T\to\infty} \frac{1}{T} \log \mathrm{PoE} = -\frac{1}{H_3(\boldsymbol{P})\overline{\log} K}. \tag{106}$$

Before proving Theorem 23, we consider the relation between $H_2$ and $H_3$.

**Theorem 24.** It holds that

$$\frac{1}{2} < \frac{H_3(\boldsymbol{P})}{H_2(\boldsymbol{P})} \le 1. \tag{107}$$

In addition, there exists a sequence of instances that the left inequality becomes arbitrarily close to the equality, and there exists another instance such that the right inequality holds with equality.

From this result, we see that the existing exponent bound $H_2(\boldsymbol{P})\overline{\log} K$ (i.e., Eq. (98)) is tight up to a factor of 2.

Recall that there is an instance such that the exponent of our algorithm is better by a factor of $O(\log\log K)$ compared with $O(H_2 \log K)$. From this result, the better exponent for this instance does not come from the looseness of the existing error probability upper bound for SR, but rather comes from essentially better error probability for this instance. In other words, our algorithm is admissible with respect to SR.

*Proof of Theorem 24.* First we derive the upper bound of $H_3$.

$$\begin{aligned} H_3(\boldsymbol{P})^{-1} &= \min_k \frac{1}{2j} \sum_{i=1}^{j} (\Delta_i^{(j)})^2 \\ &\ge \min_k \frac{1}{2j} \left( (\Delta_1^{(j)})^2 + (\Delta_j^{(j)})^2 \right) \\ &\ge \min_k \frac{1}{2j} \left( 2 \cdot \left( \frac{\Delta_j}{2} \right)^2 \right) \\ &\ge \min_k \frac{1}{4j} (\Delta_j)^2 = H_2^{-1}. \end{aligned} \tag{108}$$

The inequalities become equalities when $K = 2$.

Next consider the lower bound. From the definition of $\bar{P}^{\mathrm{mod}}(j)$, we see that

$$\sum_{i=2}^{j} \Delta_i^{(j)} = \Delta_1^{(j)}. \tag{109}$$

In addition, $\Delta_i^{(j)}$ trivially satisfies $\Delta_i^{(j)} \ge 0$ and $\Delta_1^{(j)} + \Delta_i^{(j)} \le \Delta_j$. By Lagrange multiplier method, under these constraints

$$\sum_{i=1}^{j} (\Delta_i^{(j)})^2 \tag{110}$$

is maximized when $\Delta_2 = \Delta_3 = \cdots = \Delta_j$, where $\bar{P}^{\mathrm{mod}}(j) = P_1 - (1 - 1/j)\Delta_j$. Then

$$\begin{aligned} \sum_{i=1}^{j} (\Delta_i^{(j)})^2 &\le \left( \frac{(j-1)\Delta}{j} \right)^2 + (j-1) \left( \Delta - \frac{(j-1)\Delta}{j} \right)^2 \\ &= (1 - 1/j)\Delta_j^2. \\ &< \Delta_j^2. \end{aligned} \tag{111}$$

Then we have

$$H_3(\boldsymbol{P})^{-1} \le \min_j \frac{1}{2j} \sum_{i=1}^{j} (\Delta_i^{(j)})^2$$

$$
\begin{aligned}
&< \min_j \frac{\Delta_j^2}{2j} \\
&= 2H_2(\boldsymbol{P})^{-1},
\end{aligned} \tag{112}
$$

where inequalities become arbitrarily close to equalities when $\Delta_2 = \Delta_3 = \cdots = \Delta_K = \Delta$ with $\Delta > 0$ and $K \to \infty$. This completes the proof of Theorem 24. □

*Proof of Theorem 23.* First we prove the error probability upper bound.

The best arm is rejected at the $k$-th phase only when the means of at least $K-k$ suboptimal arms including the already rejected ones exceeded the mean of the best arm. Then

$$
\begin{aligned}
\mathrm{PoE} &\le \sum_{k=1}^{K-1} \sum_{S \subset [K] \setminus \{1\} : |S| = K-k} \mathbb{P}\left[\prod_{j \in S} \{\hat{X}_{j,k} \ge \hat{X}_{1,n_k}\}\right] \\
&\le \sum_{k=1}^{K-1} \sum_{S \subset [K] \setminus \{1\} : |S| = K-k} \mathbb{P}\left[\prod_{j=2}^{K-k+1} \{\hat{X}_{j,k} \ge \hat{X}_{1,n_k}\}\right] \\
&= \sum_{k=1}^{K-1} \binom{K-1}{K-k} \mathbb{P}\left[\prod_{j=2}^{K-k+1} \{\hat{X}_{j,k} \ge \hat{X}_{1,n_k}\}\right] \\
&= \sum_{k=1}^{K-1} \binom{K-1}{K-k} \mathbb{P}\left[(\hat{X}_{1,n_k}, \ldots, \hat{X}_{K-k+1,n_k}) \in S_k\right].
\end{aligned} \tag{113}
$$

where $S_k = \{x \in \mathbb{R}^{K-k+1} : x_j \ge x_1, \forall j \in \{2, 3, \ldots, K-k+1\}\}$. Since $S_k$ is a convex set, by Cramér theorem we have

$$
\begin{aligned}
&\Pr\left[(\hat{X}_{1,n_k}, \ldots, \hat{X}_{K-k+1}) \in S_k\right] \\
&\le \exp\left(-n_k \inf_{x \in S_k} \sup_{\lambda \in \mathbb{R}^{K-k+1}} \left\{\lambda^\top x - \log \mathbb{E}[\mathrm{e}^{\lambda^\top X}]\right\}\right) \\
&\le \exp\left(-n_k \inf_{x \in S_k} \sum_{j=1}^{K} \frac{(x - P_i)^2}{2}\right),
\end{aligned} \tag{114}
$$

for $X = (X_1, X_2, \ldots, X_{K-k+1})$ with $X_i$ independently following $N(P_i, 1)$. We can easily see that the infimum is attained at $x_1 = \bar{P}^{\mathrm{mod}}(K-k+1)$ and $x_j = \max\{\bar{P}^{\mathrm{mod}}(K-k+1), P_j\}$, which results in

$$
\Pr\left[(\hat{X}_{1,n_k}, \ldots, \hat{X}_{K-k+1}) \in S_k\right] \le \exp\left(-\frac{n_k}{2} \sum_{j=1}^{K-k+1} (\Delta_j^{(K-k+1)})^2\right). \tag{115}
$$

Finally we obtain

$$
\begin{aligned}
\mathrm{PoE} &\le \sum_{k=1}^{K-1} \binom{K-1}{K-k} \exp\left(-\frac{T-K}{2(K+1-k)\overline{\log} K} \sum_{j=1}^{K-k+1} (\Delta_j^{(K-k+1)})^2\right) \\
&\le \left(\sum_{k=1}^{K-1} \binom{K-1}{K-k}\right) \exp\left(-\frac{T-K}{2\overline{\log} K} \inf_{k \in [K-1]} \frac{1}{K+1-k} \sum_{j=1}^{K-k+1} (\Delta_j^{(K-k+1)})^2\right) \\
&\le \left(\sum_{k=1}^{K-1} \binom{K-1}{K-k}\right) \exp\left(-\frac{T-K}{2\overline{\log} K} \inf_{k \in \{2,3,\ldots,K\}} \frac{1}{k} \sum_{j=1}^{k} (\Delta_1^{(k)})^2\right) \\
&\le 2^K \exp\left(-\frac{T-K}{H_3 \overline{\log} K}\right).
\end{aligned} \tag{116}
$$

Then we immediately obtain the asymptotic upper bound

$$\lim_{T\to\infty}\frac{1}{T}\log \mathrm{PoE} \le -\frac{1}{H_3\overline{\log}K}. \tag{117}$$

Next we consider the lower bound. Though we can obtain an explicit error probability lower bound when we consider Gaussians, we only derive an asymptotic bound so that it can be easily extended to general distributions. Let $Y_{i,k}$ be the mean of the samples from arm $i$ at the $k$-th phase, and $m_k = n_k - n_{k-1}$ be the number of samples of each arm at the $k$-th phase. Then we have

$$\hat{X}_{i,n_k} = \frac{\sum_{j=1}^{k} m_j Y_{i,j}}{n_k}. \tag{118}$$

Take optimal $j^*$ achieving the maximum in (104) and define $k^* = K - j^* + 1$. Consider event $\mathcal{E}$ that for all $k = 1, 2, \ldots, k^*$,

$$\begin{aligned}
&Y_{1,k} \in (P_{j^*+1}, \bar{P}^{\mathrm{mod}}(j^*)],\\
&Y_{j,k} \in (\bar{P}^{\mathrm{mod}}(j^*), \infty),\ j \in \{2, 3, \ldots, j^*\},\\
&Y_{j,k} \in (-\infty, P_{j^*+1}],\ j \in \{j^*+1, \ldots, K\}.
\end{aligned} \tag{119}$$

Under this event, arm 1 is not rejected for the first $k^* - 1$ phases and is rejected at the $k^*$-th phase. Then we have $\mathrm{PoE} \ge \Pr[\mathcal{E}]$. Here, by Cramér's theorem we have

$$\begin{aligned}
\lim_{m_k\to\infty}\frac{1}{m_k}\log \Pr[Y_{1,k} \in (P_{j^*+1}, \bar{P}^{\mathrm{mod}}(j^*)]] &= -\frac{(\Delta_1^{(j^*)})^2}{2}\\
\lim_{m_k\to\infty}\frac{1}{m_k}\log \Pr[Y_{j,k} \in (\bar{P}^{\mathrm{mod}}(j^*), \infty)] &= -\frac{(\Delta_j^{(j^*)})^2}{2},\ j \in \{2, 3, \ldots, j^*\}\\
\lim_{m_k\to\infty}\frac{1}{m_k}\log \Pr[Y_{j,k} \in (-\infty, \bar{P}^{\mathrm{mod}}(j^*+1)]] &= 0,\ j \in \{j^*+1, \ldots, K\}.
\end{aligned} \tag{120}$$

Then we have

$$\begin{aligned}
\lim_{T\to\infty}\frac{1}{T}\log \mathrm{PoE} &\ge \lim_{T\to\infty}\frac{1}{T}\log \Pr[\mathcal{E}]\\
&= -\lim_{T\to\infty}\sum_{k=1}^{k^*}\frac{m_k}{T}\sum_{j=1}^{j^*}\frac{(\Delta_j^{(j^*)})^2}{2}\\
&= -\lim_{T\to\infty}\frac{n_{k^*}}{T}\sum_{j=1}^{j^*}\frac{(\Delta_j^{(j^*)})^2}{2}\\
&= -\frac{1}{(K+1-k^*)\overline{\log}K}\sum_{j=1}^{j^*}\frac{(\Delta_j^{(j^*)})^2}{2}\\
&= -\frac{1}{j^*\overline{\log}K}\sum_{j=1}^{j^*}\frac{(\Delta_j^{(j^*)})^2}{2}\\
&= -\frac{1}{H_3(\boldsymbol{P})\overline{\log}K}.
\end{aligned} \tag{121}$$

This completes the proof of Theorem 23. □

# H PROOF OF LEMMA 8

The goal of this section is to show an example where the proposed algorithm outperforms SR. The structure of this section is as follows: Appendix H.1 shows the instance where the proposed algorithm's rate is larger than SR's known rate by a factor of $\Omega(\log K/(\log\log K))$. However, this does not exclude the possibility that SR's known rate $H_2$ is loose (Appendix G.1). To see our algorithm provably outperforms SR, we show the SR's rate based on $H_2$ is tight up to a factor of 2 (Appendix G.2).

### H.1 Explicit Construction of the Instance

Let $\boldsymbol{P}$ be such that

$$\begin{aligned} P_1 &= 2, \\ P_2 = P_3 &= \cdots = P_{\log K} = 1, \\ P_{\log K+1} &= \cdots = P_K = 0. \end{aligned}$$

Then, on this instance, SR's rate[13] is proportional to

$$\frac{1}{(\log K) H_2(\boldsymbol{P})} \propto \frac{1}{(\log K) K}. \tag{122}$$

Meanwhile, Theorem 3 implies the rate of Almost Tracking is lower bounded by

$$\inf_{\boldsymbol{Q}: i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})} \sum_i w_i(\boldsymbol{Q}) \frac{(Q_i - P_i)^2}{2} \propto \frac{1}{H_1(\boldsymbol{P})} \inf_{\boldsymbol{Q}: i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})} \frac{S(\boldsymbol{Q}, \boldsymbol{P})}{Z(\boldsymbol{Q})} \tag{123}$$

$$\propto \frac{1}{K} \inf_{\boldsymbol{Q}: i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})} \frac{S(\boldsymbol{Q}, \boldsymbol{P})}{Z(\boldsymbol{Q})} \tag{124}$$

**Lemma 25.** For any $\boldsymbol{Q}$ such that $i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})$, it holds that

$$\frac{S(\boldsymbol{Q}, \boldsymbol{P})}{Z(\boldsymbol{Q})} = \Omega\left(\frac{1}{(\log \log K)}\right). \tag{125}$$

Lemma 25, combined with Eq. (122) and Eq. (124), implies that our algorithm outperforms SR by a factor of $\Omega(\log K / (\log \log K))$.

*Proof of Lemma 25.*

$$\begin{aligned} \frac{S(\boldsymbol{Q}, \boldsymbol{P})}{Z(\boldsymbol{Q})} &:= \frac{\sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})}}{\sum_i \frac{1}{D_i(\boldsymbol{Q})}} H_1(\boldsymbol{P}) && (126) \\ &\geq \frac{\sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})}}{\sum_i \frac{1}{D_i(\boldsymbol{Q})}} \times K && (127) \\ &\geq \frac{1}{4} \frac{\sum_i (Q_i - P_i)^2}{\sum_i \frac{1}{D_i(\boldsymbol{Q})}} && \text{(Lemma 21 implies } 1/D_i(\boldsymbol{Q}) \geq 1/(4K)) \\ &\geq \frac{1}{4} \frac{\sum_i (Q_i - P_i)^2}{\sum_i \frac{1}{i + \sum_{j>i} \frac{(\Delta_i^Q)^2}{(\Delta_j^Q)^2}}}, && \text{(by Lemma 20)} \\ & && (128) \end{aligned}$$

where $\Delta_i^Q := Q_1 - Q_i$. We say arm $i$ *moved significantly* if $|P_i - Q_i| \geq 1/3$. Eq.(128) implies that

$$\frac{S(\boldsymbol{Q}, \boldsymbol{P})}{Z(\boldsymbol{Q})} \geq \Omega\left(\frac{\#\text{ of significant moves}}{\log K}\right).$$

If number of significant moves are $\geq \log K$, then $\frac{S(\boldsymbol{Q}, \boldsymbol{P})}{Z(\boldsymbol{Q})} = \Omega(1)$. In the following, we assume there are at most $(\log K) - 1$ significant moves, which implies $\max_i Q_i \geq 2/3$.

Let $i, j > \log K$. If both $i, j$ did not move significantly, then $Q_i, Q_j \in [-1/3, 1/3]$ and thus

$$\frac{(\Delta_i^Q)^2}{(\Delta_j^Q)^2} \geq \left(\frac{2/3 - 1/3}{2/3 - (-1/3)}\right)^2 \geq 1/9 = \Omega(1). \tag{129}$$

[13] Here, an algorithm has rate $r$ if its probability of error is $\text{poly}(T) \exp(-rT)$ asymptotically.

By assumption, the number of significantly moved arms is at most $\log K$. Therefore,

$$\text{(Denominator of Eq. (128))} = \sum_{i\in[K]} \frac{1}{i + \sum_{j>i} \frac{(\Delta_i^Q)^2}{(\Delta_j^Q)^2}} \tag{130}$$

$$\leq \sum_{i\leq \log K} \frac{1}{i} + \sum_{i\in[\log K+1, 2\log K]} \frac{1}{i} + K \times \frac{1}{(K-\log K)/9} \tag{131}$$

$$\leq \sum_{i\leq 2\log K} \frac{1}{i} + O(1) \tag{132}$$

$$= O(\log(2\log K)+1) = O(\log\log K), \tag{133}$$

where, the second term is derived by the following discussion: At most $\log K$ arm among those in $\{\log K + 1, \ldots, K\}$ are significantly moved. If $i$ has not significantly moved, then at least $K - \log K$ of $\{j : (\Delta_i^Q)^2/(\Delta_j^Q)^2\}$ are $\geq 1/9$ due to Eq. (129) and the fact that at most $\log K$ of $j$ has significant move.

In summary,

$$(128) \geq \frac{1}{\sum_i \frac{1}{i+\sum_{j>i}\frac{(\Delta_i^Q)^2}{(\Delta_j^Q)^2}}}$$

$$\text{($\arg\max_i Q_i \neq 1$ requires at least one significant move, which implies $\textstyle\sum_i (Q_i - P_i)^2 = \Omega(1)$)}$$

$$\geq \Omega\left(\frac{1}{\log\log K}\right). \qquad \text{(by Eq. (133))}$$

□

# I PROOF OF LEMMA 9

This section shows an instance where SR outperforms our algorithm's bound. The particular instance is such that $P_i = -\sqrt{i}$ for $i = 1, 2, \ldots, K$ where $i^*(\boldsymbol{P}) = 1$. In this case, the rate of SR is

$$\frac{1}{(\log K) H_2(\boldsymbol{P})} \propto \frac{1}{(\log K)}. \tag{134}$$

For Almost Tracking, $H_1(\boldsymbol{P}) = \Theta(\log K)$ and the rate is

$$\frac{1}{H_1(\boldsymbol{P})} \inf_{\boldsymbol{Q}: i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})} \frac{S(\boldsymbol{Q}, \boldsymbol{P})}{Z(\boldsymbol{Q})} := \inf_{\boldsymbol{Q}: i^*(\boldsymbol{P}) \notin i^*(\boldsymbol{Q})} \frac{1}{Z(\boldsymbol{Q})} \sum_i \frac{(Q_i - P_i)^2}{D_i(\boldsymbol{Q})} \tag{135}$$

$$= O\left(\frac{1}{(\log K)^2}\right) \tag{136}$$

where the last step is derived from the instance $\boldsymbol{Q}$ such that

$$Q_i = \begin{cases} P_i & \text{if } i \neq 2 \\ 2P_1 - P_2 & \text{if } i = 2. \end{cases}$$

Here, $D_i = \Theta(i + \sum_{j>i} \frac{\Delta_i^2}{\Delta_j^2})$ (Eq. (20) and Eq. (21)), from which we have

$$D_1(\boldsymbol{Q}) = D_2(\boldsymbol{Q}) = \Theta(\log K) \tag{137}$$

$$Z(\boldsymbol{Q}) = \sum_i \frac{1}{D_i(\boldsymbol{Q})} = \Theta\left(\sum_j \frac{1}{\log K + j}\right) = \Theta(\log K). \tag{138}$$

In summary, comparing Eq. (134) and Eq. (135), SR outperforms the bound of our algorithm by the rate of $\Omega(\log K)$. Note that Eq. (135) only bounds our rate from below and the actual rate of our algorithm might be better than this.

# J NON-CONVEXITY OF THE OBJECTIVE

Eq. (1) is an optimization of $\boldsymbol{w}(\cdot) = \boldsymbol{w}(\boldsymbol{Q})$:

$$\sup_{\boldsymbol{w}(\cdot)\in\Delta^{(K)}} \inf_{\boldsymbol{Q}\in\mathcal{Q}^K} \inf_{\boldsymbol{P}:\mathcal{I}^*(\boldsymbol{P})\notin\mathcal{I}^*(\boldsymbol{Q})} H(\boldsymbol{P}) \sum_{i\in[K]} w_i(\boldsymbol{Q}) D(Q_i\|P_i).$$

The optimization can be solved by maximizing

$$V_{\boldsymbol{Q}}(\boldsymbol{w}) := \inf_{\boldsymbol{P}:\mathcal{I}^*(\boldsymbol{P})\notin\mathcal{I}^*(\boldsymbol{Q})} H(\boldsymbol{P}) \sum_{i\in[K]} w_i D(Q_i\|P_i) \tag{139}$$

for each $\boldsymbol{w} = \boldsymbol{w}(\boldsymbol{Q})$. Eq. (139), which is a infimum of a linear objective, is concave in $\boldsymbol{w}$. However, obtaining the value of $V_{\boldsymbol{Q}}(\boldsymbol{w})$ for a given vector $\boldsymbol{w}$ is a non-convex problem of finding the infimum of $\boldsymbol{P}$ for typical $H(\boldsymbol{P})$.

# K TECHNICAL LIMITATIONS

In this paper, we have considered the best arm identification problem with sub-Gaussian rewards. While sub-Gaussian rewards are common in practice and cover major distributions, including any bounded distributions as well as Gaussian distributions, it is still important to consider the case where the rewards are not sub-Gaussian. Moreover, while the results in Section 3 hold universally for any risk measure $H(\boldsymbol{P})$, the results in Section 4 build upon the most widely adopted risk measure $H_1(\boldsymbol{P})$.

- The two-approximation algorithm (Appendix B) is quite general. It should hold with a large class of one-parameter distributions (e.g., one-parameter exponential family) and any risk measure $H(\boldsymbol{P})$.
- The constant-ratio ceiling (Appendix C) should hold with any distribution and any risk measure.
- Appendix D: Theorem 3, which guarantees the trackability of Almost Tracking, depends on the sub-Gaussian assumption.
- Appendix E: Lemma 15, which is about large deviation, is easily extendable for a larger class of distributions, such as one-parameter exponential family of distributions.
- Appendix F: Theorem 5, or the stability Lemma 17, which is used by this theorem, depends on the sub-Gaussian assumption and property of the risk measure $H_1(\boldsymbol{P})$.
- Appendix G: Theoretical results for SR (Appendix G) depend on the sub-Gaussian assumption, just like the existing results of Audibert et al. (2010).
- Appendix H and Appendix I: The instances of Lemma 8 and Lemma 9 are for sub-Gaussian rewards since we compare SR and Almost Tracking.

# L DETAILS OF THE EXPERIMENTS

Our code is implemented in Python and runs on a standard Linux server. Our code does not require any GPUs nor does it require a large amount of memory.

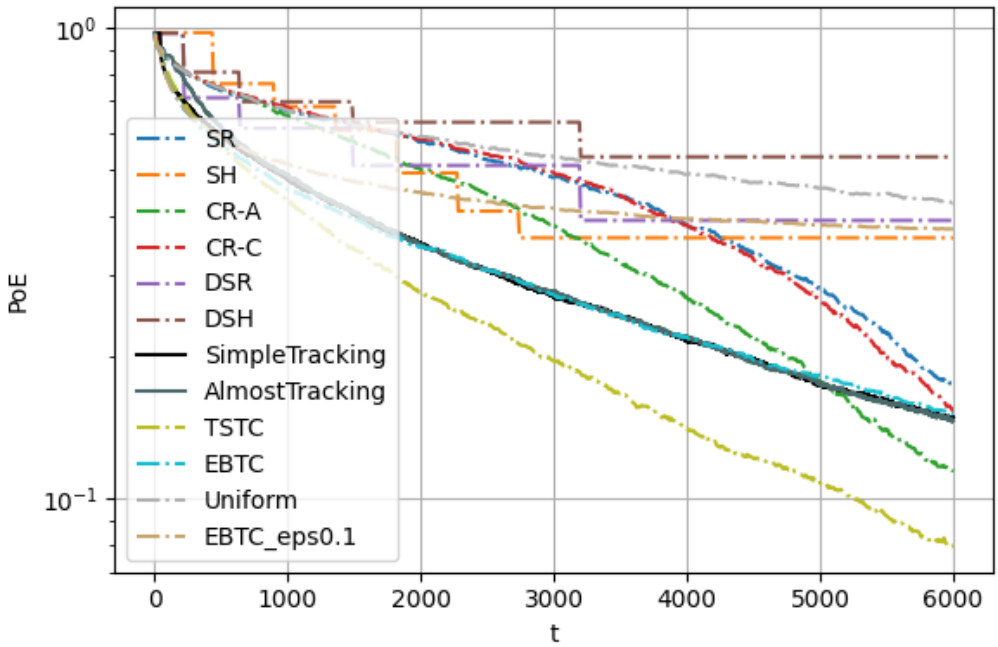
## L.1 Compared Algorithms

- Successive rejects (SR, Audibert et al. (2010)): This algorithm splits the rounds into $K-1$ batches of predefined sizes. At the end of each batch, it eliminates one arm. This algorithm requires $T$.
- Sequential halving (SH, Karnin et al. (2013)): This algorithm splits the rounds into predefined size of $\lceil \log_2 K \rceil$ batches. At the end of each batch, it eliminates half of the arms. This algorithm requires $T$.
- Continuous rejects (CR, Wang et al. (2023)): This algorithm is a dynamic version of SR where the size of each batch is adaptive. This algorithm requires $T$. There are two versions of CR. In our experiments, we show the results of CR-A, which consistently outperforms CR-C. We set $\theta_0 = 0.01$.

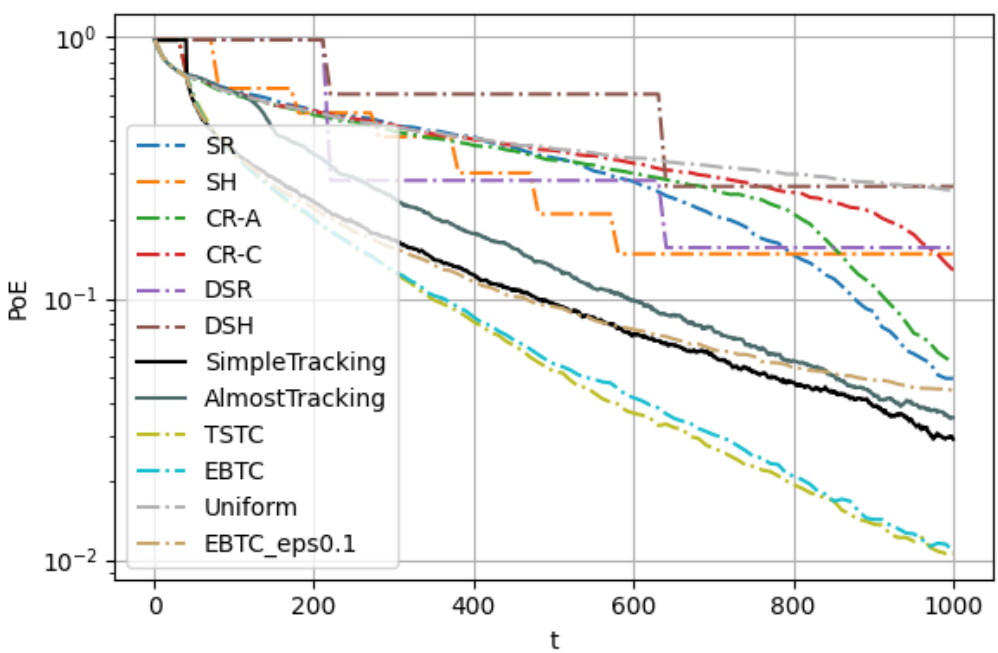
- Double sequential halving (DSH, Zhao et al. (2023)): This is a meta-algorithm that implements a doubling epoch strategy that combined with SH. It begins with $O(K \log_2 K)$ samples and doubles the sample budget in each subsequent epoch, running a complete SH procedure within each epoch. This algorithm does not require $T$ beforehand.
- Double successive rejects (DSR, a natural doubling-based anytime variant of SR): This algorithm is an version of DSH where the base SH algorithm is replaced by SR.
- EB-TC and TS-TC (Shang et al., 2020; Jourdan et al., 2022) are two empirically good versions of top-two Thompson sampling (Russo, 2020) that are designed for the fixed-confidence identification.
- EB-TC$_{\varepsilon_0}$ (Jourdan et al., 2023) is a version of EB-TC with $\varepsilon_0$-best identification. The value of $\varepsilon_0$ is set to 0.10. We chose the "fixed" version of it with $\beta = 0.5$. These choices are based on their report that setting $\varepsilon_0 = 0$ suffers from poor empirical performance for moderate value of $\delta$ as well as Figures 5, 6, and 7 therein.
- Uniform is a naive algorithm that draws arms in a round-robin fashion.
- Simple tracking: Algorithm 1 in this paper. This algorithm does not require $T$ beforehand.
- Almost tracking: Algorithm 2 in this paper. This algorithm does not require $T$ beforehand. We set the size of each batch $N$ to be $2K$ and $C_{\mathrm{suf}} = 0.999$ for all our experiments.

### L.2 Instances Used to Calculate Table 1

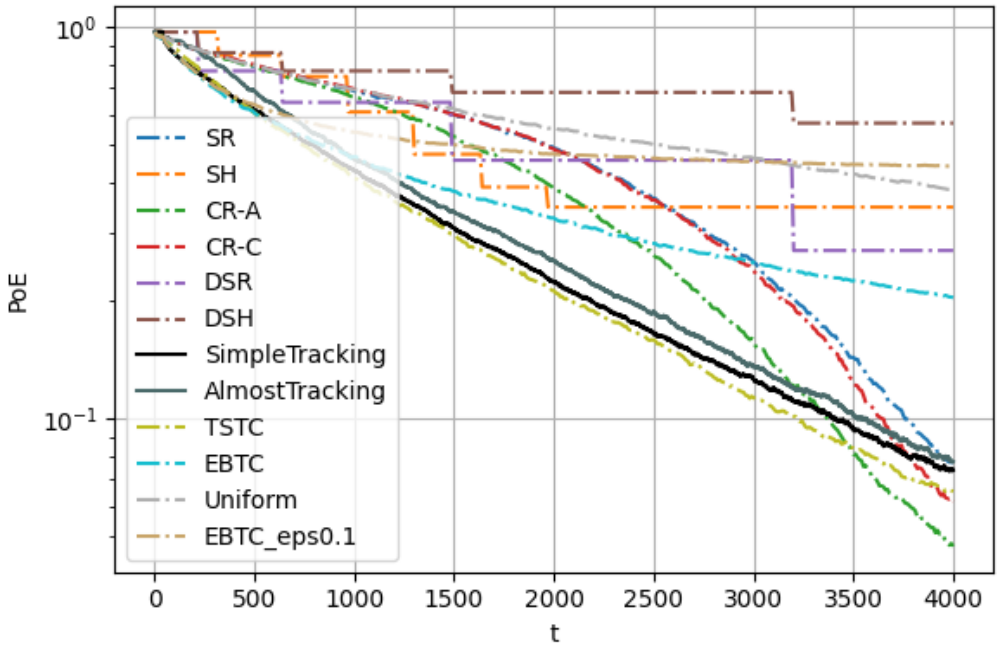
This section shows the instances that we used to calculate the minimax rates in Table 1. For all instances, we set $K = 40$. The budget $T$ is determined so that the PoE of the best algorithm is between 0.01 and 0.1. We run each simulation for 10,000 times and the figures are empirical PoE values.



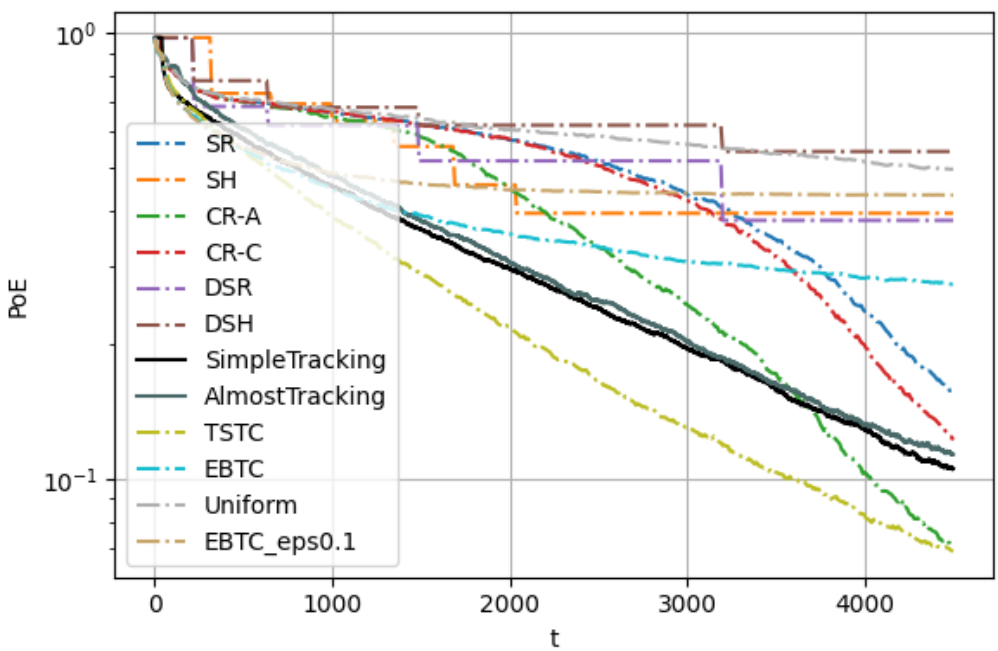

Instance 1: $\boldsymbol{P} = \{1 - (i-1) \times 0.05\}$



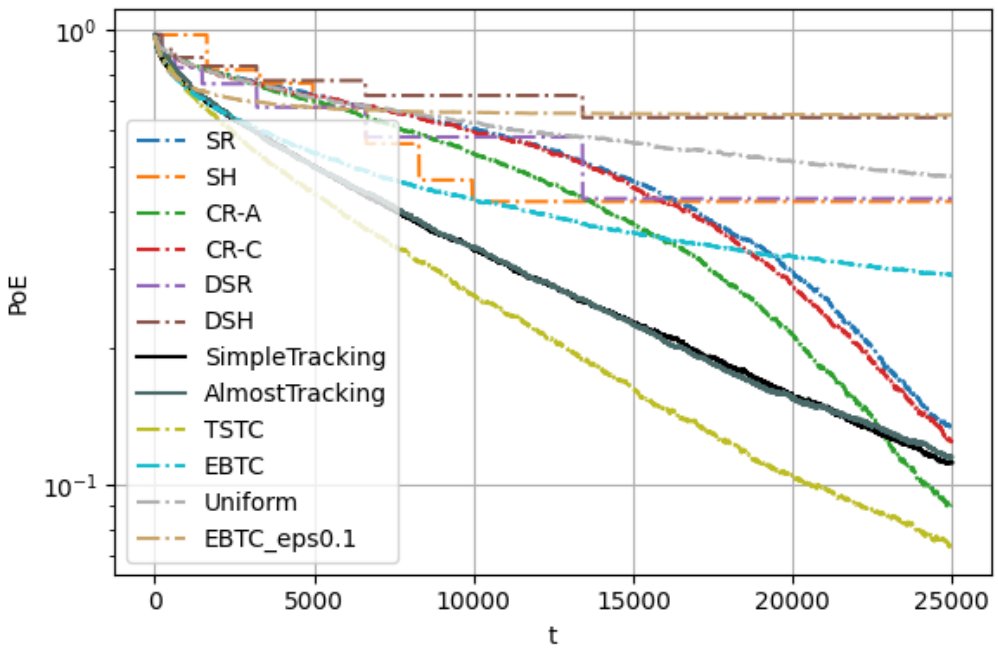

Instance 2: $\boldsymbol{P} = \{10\frac{(i-1)^{0.8}}{39^{0.8}}\}$



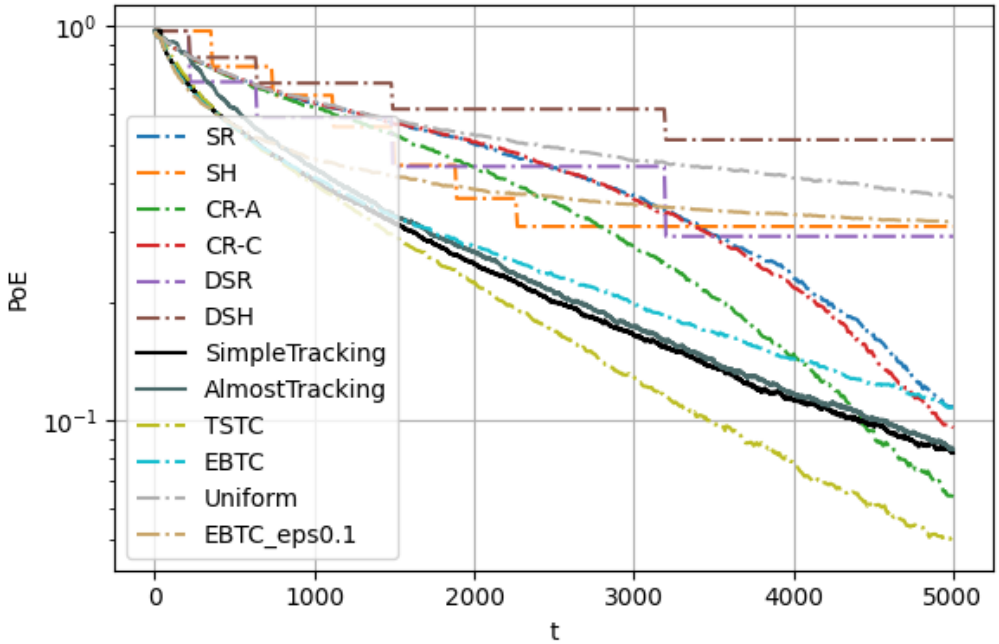

Instance 3: $\boldsymbol{P} = \{1 - \sqrt{i-1}/10\}$. This instance is in favor of SR that we discussed in Section I.



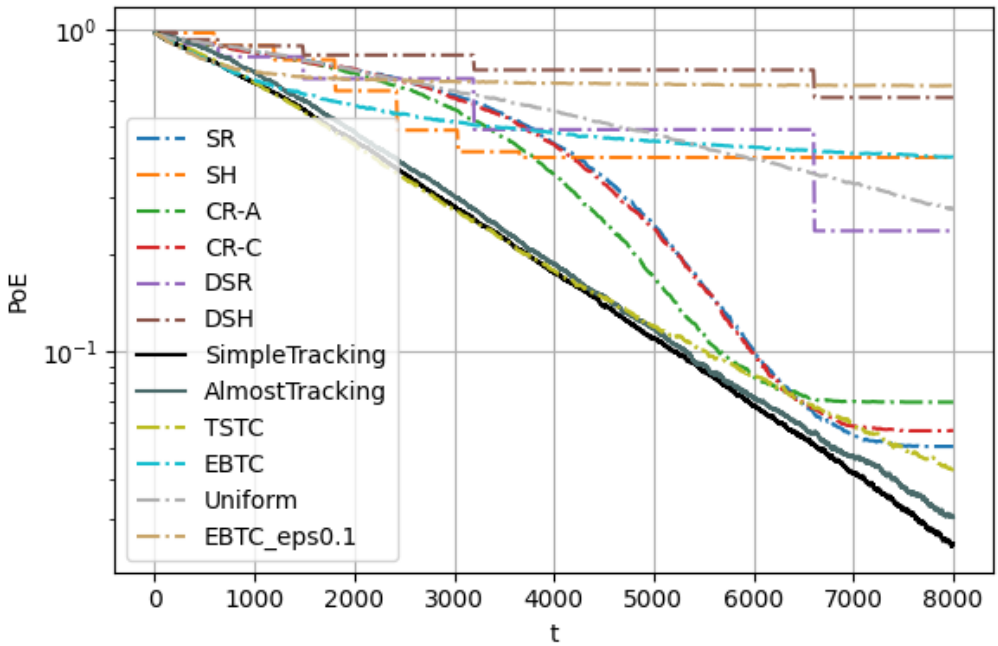

Instance 4: $\boldsymbol{P} = \{1, 0.9, 0.9, 0.9, 0.9, \underbrace{0, 0, \ldots, 0, 0}_{\text{35 arms with zero mean}}\}$. This instance is in favor of Almost Tracking that we discussed in Section H.



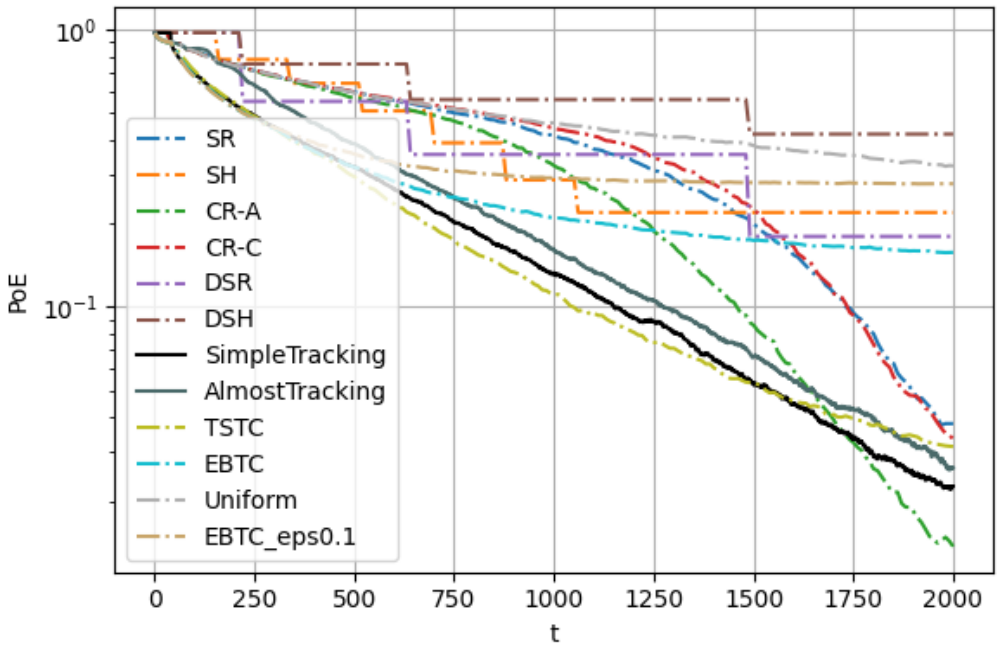

Instance 5: $\boldsymbol{P} = \{\sin((K-1)\pi/(2K))\} \cup \{\sin(9\pi(K-i)/(20K))\}_{i=2}^{40}$. This is Concave set of arms (Wang et al., 2023).



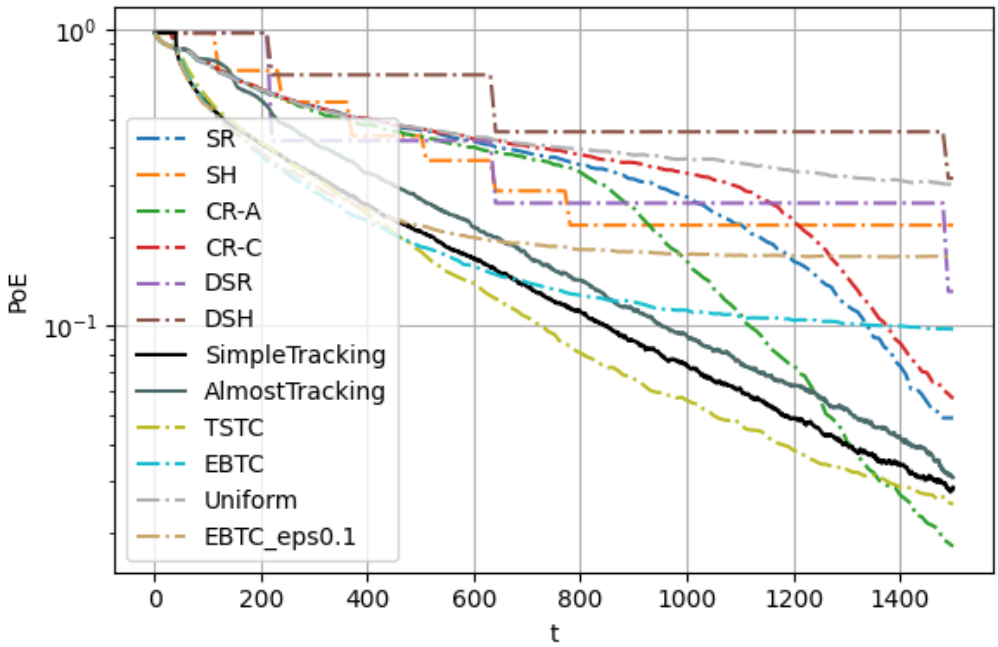

Instance 6: $\boldsymbol{P} = \{0.75 \times 3^{-i/10}\}$. This is Convex set of arms (Wang et al., 2023).



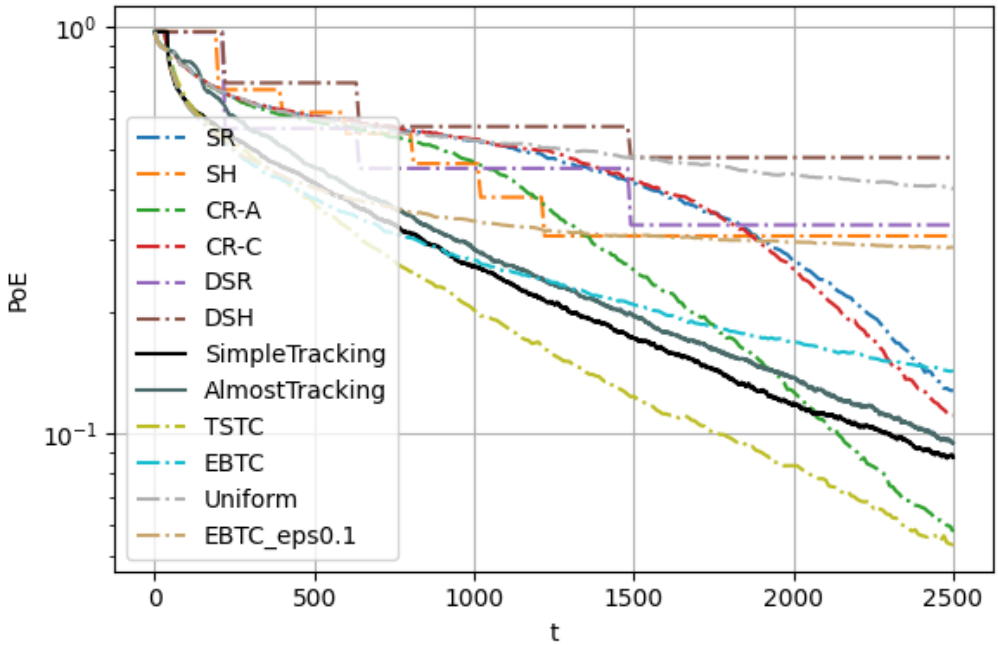

Instance 7: $\boldsymbol{P} = \{1, \underbrace{0.8, 0.8, \ldots, 0.8, 0.8}_{39 \text{ arms}}\}$.




Instance 8: $\boldsymbol{P} = \{1, \underbrace{0.8, 0.8, 0.8}_{3 \text{ arms}}, \underbrace{0.8, 0.8, \ldots, 0.8, 0.8}_{6 \text{ arms}}, \underbrace{0.2, 0.2, \ldots, 0.2, 0.2}_{10 \text{ arms}}, \underbrace{0, 0, \ldots, 0, 0}_{20 \text{ arms}}\}$.




Instance 9: $\boldsymbol{P} = \{1.0, 0.8, 0.8, \underbrace{0, 0, \ldots, 0, 0}_{37 \text{ arms}}\}$.




Instance 10: $\boldsymbol{P} = \{1.0, 0.9, 0.85, 0.8, \underbrace{0, 0, \ldots, 0, 0}_{36 \text{ arms}}\}$.

Figure 4: Probability of Error (PoE) as a function of rounds $t$.

### L.3 Restatement of Table 1 with Confidence Intervals

Performance of algorithms with 95% confidence intervals are shown in Table 3 and 4.

Table 3: Minimax rate for $H_1$ (lower, plugin, upper) for each algorithm. The column "Plugin" is based on the empirical average of PoE, which is the same as Table 1 in the main paper. "Lower Bound" and "Upper Bound" are based on the lower and upper bounds of PoE with 95% confidence intervals. One can see that Simple Tracking and Almost Tracking outperform the other algorithms with statistical significance.

| Algorithm | Lower Bound | Plugin | Upper Bound |
|---|---|---|---|
| SR | 0.443 | 0.456 | 0.470 |
| SH | 0.216 | 0.222 | 0.229 |
| CR-A | 0.423 | 0.435 | 0.447 |
| CR-C | 0.304 | 0.312 | 0.319 |
| DSR | 0.227 | 0.234 | 0.242 |
| DSH | 0.111 | 0.114 | 0.116 |
| Simple Tracking | 0.521 | 0.537 | 0.555 |
| Almost Tracking | 0.494 | 0.509 | 0.525 |
| TS-TC | 0.375 | 0.396 | 0.424 |
| EB-TC | 0.177 | 0.181 | 0.186 |
| Uniform | 0.154 | 0.158 | 0.163 |
| EB-TC$_{\varepsilon_0}$ | 0.122 | 0.125 | 0.128 |

Table 4: Minimax rate for $H_2$ (lower, plugin, upper) for each algorithm. The column "Plugin" is based on the empirical average of PoE, which is the same as Table 1 in the main paper. "Lower Bound" and "Upper Bound" are based on the lower and upper bounds of PoE with 95% confidence intervals.

| Algorithm | Lower Bound | Plugin | Upper Bound |
|---|---|---|---|
| SR | 0.249 | 0.256 | 0.263 |
| SH | 0.103 | 0.106 | 0.108 |
| CR-A | 0.264 | 0.271 | 0.279 |
| CR-C | 0.189 | 0.194 | 0.199 |
| DSR | 0.128 | 0.131 | 0.134 |
| DSH | 0.055 | 0.056 | 0.058 |
| Simple Tracking | 0.253 | 0.260 | 0.267 |
| Almost Tracking | 0.249 | 0.255 | 0.262 |
| TS-TC | 0.216 | 0.227 | 0.244 |
| EB-TC | 0.102 | 0.104 | 0.107 |
| Uniform | 0.089 | 0.091 | 0.093 |
| EB-TC$_{\varepsilon_0}$ | 0.070 | 0.072 | 0.074 |

### L.4 Instances Derived from Real-world Data

In addition to the synthetic datasets, we adopt two real-world datasets to evaluate the performance of our algorithms. The Open Bandit Dataset Saito et al. (2021) is a publicly available, real-world logged bandit dataset collected a prominent Japanese fashion e-commerce platform (ZOZO, Inc.). It consists of 12 million impressions consists of 80 advertisements. We view each advertisements as a Gaussian arm and normalized the reward with average standard deviation so that the variance of each arm is 1.

The Movielens 1M dataset Harper and Konstan (2015) comprises over 1,000,000 anonymous ratings of nearly 3,900 movies by 6,040 MovieLens users who joined in 2000. Each datapoint includes user IDs, movie IDs, 5-star ratings, timestamps, and many other features. In our experiments, we filter movies with 2,000 or less ratings to keep 31 popular movies. We use each movie as an arm and its average rating as the mean reward, which is normalized by the average standard deviation of all movies so that the variance of each arm is 1.

Table 5: Open Bandit: CTR values extracted from the Open Bandit dataset.

| | | | | |
|---|---|---|---|---|
| 0.0029265 | 0.0014464 | 0.0021134 | 0.0026464 | 0.0018947 |
| 0.0032350 | 0.0024874 | 0.0052780 | 0.0037272 | 0.0025919 |
| 0.0015018 | 0.0033327 | 0.0018368 | 0.0020283 | 0.0029336 |
| 0.0030222 | 0.0032011 | 0.0036364 | 0.0036137 | 0.0018426 |
| 0.0017718 | 0.0023036 | 0.0028038 | 0.0025506 | 0.0024710 |
| 0.0019308 | 0.0021782 | 0.0016784 | 0.0037885 | 0.0015287 |
| 0.0045120 | 0.0041963 | 0.0036784 | 0.0032292 | 0.0055569 |
| 0.0055678 | 0.0028800 | 0.0035584 | 0.0044478 | 0.0053337 |
| 0.0026211 | 0.0055760 | 0.0035852 | 0.0048702 | 0.0024826 |
| 0.0051337 | 0.0039318 | 0.0055106 | 0.0044275 | 0.0057023 |
| 0.0034024 | 0.0056714 | 0.0049135 | 0.0028941 | 0.0026866 |
| 0.0038009 | 0.0026913 | 0.0037623 | 0.0049876 | 0.0055036 |
| 0.0048012 | 0.0059725 | 0.0044809 | 0.0056396 | 0.0033993 |
| 0.0041044 | 0.0038471 | 0.0019121 | 0.0018957 | 0.0035998 |
| 0.0022913 | 0.0030215 | 0.0027332 | 0.0025879 | 0.0020447 |
| 0.0026221 | 0.0036932 | 0.0024460 | 0.0052332 | 0.0056697 |

**Open Bandit Dataset:** Table 5 shows the CTR values extracted from Open Bandit dataset ($K = 80$). Based on the CTR values, we derive the instance $P$ by normalizing the mean standard deviation ($= 0.057774753125$). We then modeled each draw as a result of Milli-impressions (1,000 impressions) and calculated the instance $\boldsymbol{P}$ by multiplying the normalized CTRs by $\sqrt{10^3}$. We set $T = 3000$ and run each simulation for 10,000 times.

Table 6: MovieLens 1M: Normalized ratings of movies with $> 2000$ ratings.

| | | | | |
|---|---|---|---|---|
| 0.86074 | 0.79806 | 0.90208 | 0.79304 | 0.88125 |
| 0.82937 | 0.89074 | 0.86747 | 0.85094 | 0.68196 |
| 0.80458 | 0.84699 | 0.81170 | 0.86348 | 0.75277 |
| 0.79061 | 0.85860 | 0.89554 | 0.87036 | 0.86317 |
| 0.82550 | 0.91091 | 0.81759 | 0.82508 | 0.74799 |
| 0.83192 | 0.83041 | 0.85564 | 0.84388 | 0.78111 |
| 0.90499 | | | | |

**Movielens 1M Dataset:** Table 6 shows the normalized ratings of movies with $> 2000$ ratings ($K = 31$). We normalize the ratings by the mean standard deviation of the ratings ($= 0.17820006619699696$) to obtain instance $\boldsymbol{P}$. We set $T = 10000$ and run each simulation for 10,000 times.

Note that the simulation results with these two instances derived from real-world data are shown in Figure 2 in the main paper.

### L.5 Suboptimality of Two-two Algorithms

In our simulation, we have tested two top-two algorithms, Thompson Sampling - Transportation Costs (TS-TC) and Empirical Best - Transportation Costs (EB-TC). These algorithms are designed for the fixed-confidence setting and have a zero rate in the fixed-budget setting. To empirically demonstrate the suboptimality of the Two-two algorithms, we consider a version of Instance with different $T$. Table 7 shows that the rate ($= (H(\boldsymbol{P})/T)\log(1/\text{PoE})$) for Top-Two keeps decreasing as $T$ increases, implying that its error probability only decays sub-exponentially. In contrast, the PoE of other good algorithms (SR, CR, and Tracking) decays exponentially fast until it reaches below $1/10,000$.

Table 7: Estimated minimax rates of algorithms on instance 9 with different value of $T$ (larger better). Here, $\infty$ means that all 10,000 runs correctly identified the best arm.

| T | SR | SH | CR-A | CR-C | DSR | DSH | S.Track | A.Track | TS-TC | EB-TC |
|---|---|---|---|---|---|---|---|---|---|---|
| 2000 | 0.637 | 0.321 | 0.915 | 0.671 | 0.366 | 0.200 | 0.785 | 0.773 | 0.721 | 0.415 |
| 10000 | $\infty$ | 0.172 | $\infty$ | $\infty$ | 0.206 | 0.087 | $\infty$ | $\infty$ | 0.237 | 0.094 |
| 20000 | $\infty$ | 0.132 | $\infty$ | $\infty$ | $\infty$ | 0.072 | $\infty$ | $\infty$ | 0.136 | 0.050 |

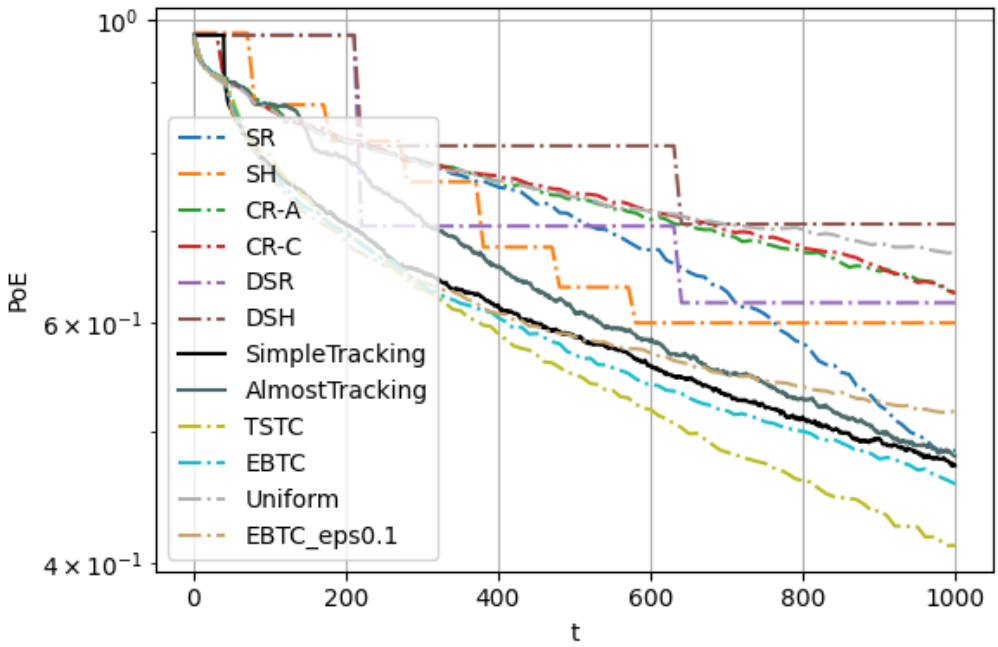
### L.6 An Instance Demonstrating the Worst Performance of Continuous Rejection Sampling

Continuous Rejection (CR, Wang et al. (2023)) outperformed Successive Rejection (SR) in our main simulation (Table 1). Still, we found the performance of CR is more sensitive to the number of rounds $T$. In particular, CR tends to have high PoE when **there are many similarly suboptimal arms** compared to the total number of samples. CR eliminates arms at each time step based on a significant gap between either the worst active arm and the second worst (in CR-C), or between the worst arm and the average of the remaining arms (in CR-A). It essentially performs uniform sampling across active arms. However, this strategy becomes inefficient when the second worst arm is similarly bad (CR-C), or when there are so many arms close in performance to the worst arm that the average does not differ significantly (CR-A). In such cases, CR struggles to remove suboptimal arms from the active set in a timely manner, leading to excessive and inefficient sampling.

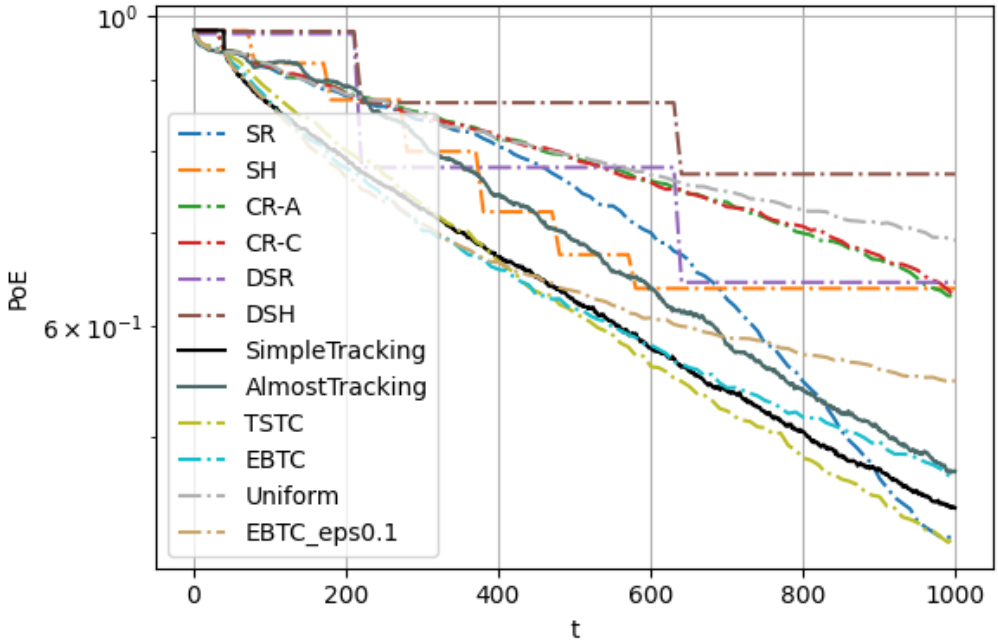
The key bottleneck lies in the **timing of the first elimination** of a group of similarly suboptimal arms. If this elimination is delayed due to limited remaining time, CR ends up with uniform sampling, which leads to a high PoE.

The following experiments allow us to observe the limitations of CR more clearly. To highlight the limitation of CR, we set $\theta_0 = 1/\overline{\log} K$, which makes the timing of the first elimination at least the same or later than SR. The results in Figure 5 show the outcome of best arm identification performed on the same bandit instance with a much shorter time horizon of $T = 1000$. One can check that CR-A and CR-C exhibit significantly higher error probabilities compared to the other algorithms, which can be largely attributed to the structural issue of CR mentioned earlier—namely, the delayed initiation of arm elimination.

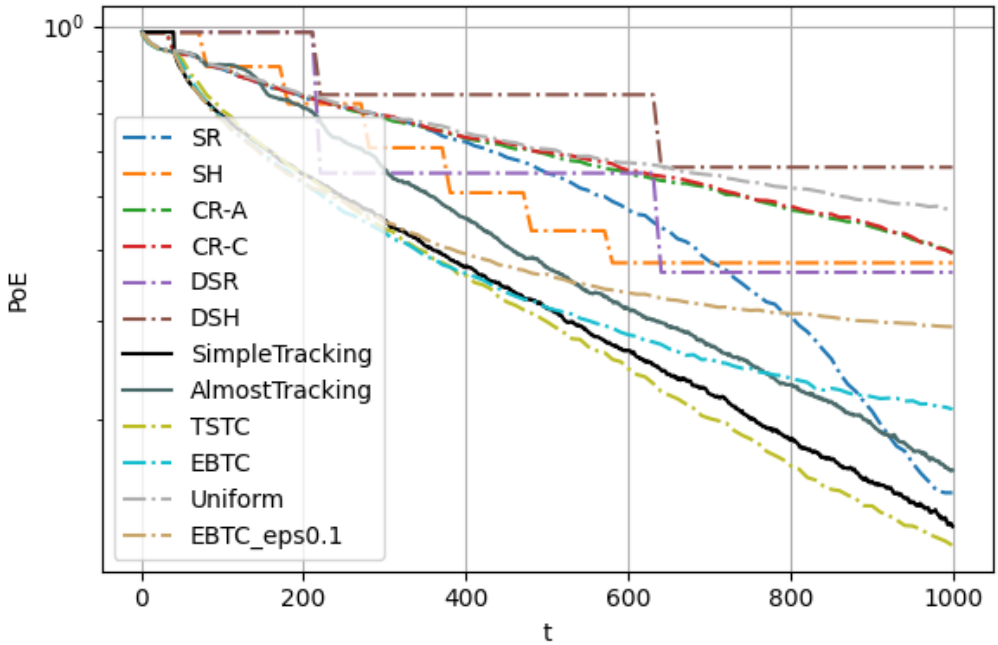
In contrast, SR and Tracking are more robust against the choice of $T$. SR follows a predetermined schedule in which one arm is eliminated at each checkpoint based on its index $i$ and the total time $T$, ensuring that less promising arms are eliminated in a timely manner, maintaining reasonable sampling efficiency. Our Tracking algorithm, meanwhile, dynamically updates the sampling distribution based on the empirical means at each time step. Even when many poor arms are present, they are quickly assigned low distribution weights around the same time, allowing the algorithm to focus resources efficiently.



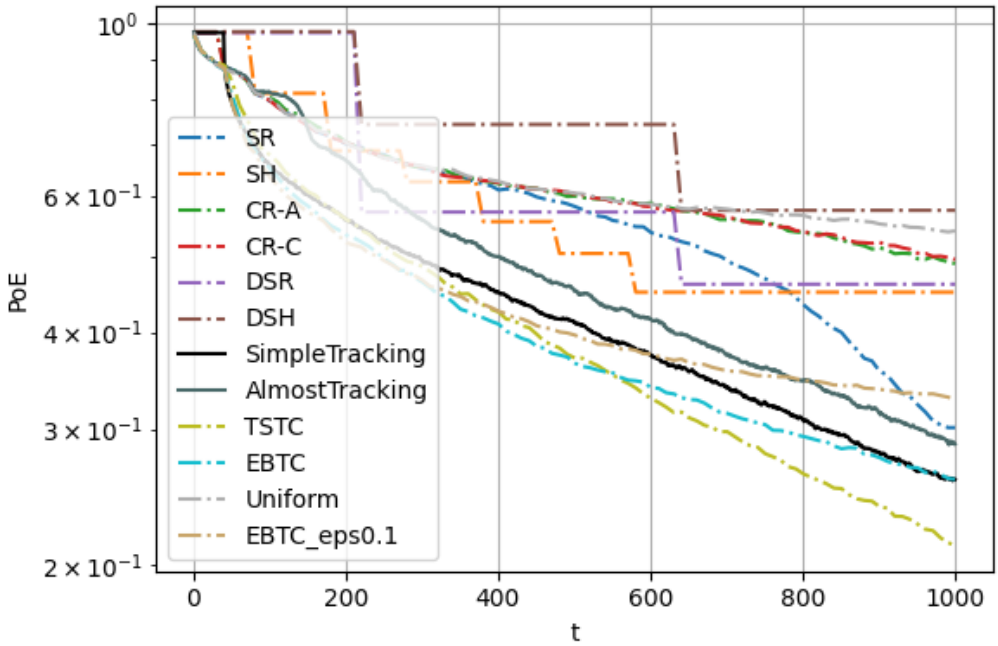

Modified Instance 1: $\boldsymbol{P} = \{1 - (i-1) \times 0.05\}$ with $T = 1000$




Modified Instance 3: $\boldsymbol{P} = \{1 - \sqrt{i-1}/10\}$ with $T = 1000$.




Modified Instance 8: $\boldsymbol{P} = \{1, \underbrace{0.8, 0.8, 0.8}_{3 \text{ arms}}, \underbrace{0.8, 0.8, \ldots, 0.8, 0.8}_{6 \text{ arms}}, \underbrace{0.2, 0.2, \ldots, 0.2, 0.2}_{10 \text{ arms}}, \underbrace{0, 0, \ldots, 0, 0}_{20 \text{ arms}}\}$ with $T = 1000$.




Modified Instance 10: $\boldsymbol{P} = \{1.0, 0.9, 0.85, 0.8, \underbrace{0, 0, \ldots, 0, 0}_{36 \text{ arms}}\}$ with $T = 1000$.

Figure 5: Probability of Error (PoE) as a function of rounds $t$, under the same setting as Figure 4 but with a shorter time horizon ($T = 1000$).