Best Arm Identification in Rare Events

Anirban Bhattacharjee[†] **Sushant Vijayan**[†] **Sandeep Juneja**[†, ‡]

[†]School of Technology and Computer Science, Tata Institute of Fundamental Research, Mumbai, India ,
‡ Visiting Researcher, Google Research India

Abstract

We consider the Best Arm Identification (BAI) problem in the stochastic multi-armed bandit framework, where each arm has a small probability of realizing large rewards, while with overwhelming probability, the reward is zero. A key application of this framework is in online advertising, where click rates of advertisements could be a fraction of a single percent and final conversion to sales, while highly profitable, may again be a small fraction of the click rates. Lately, algorithms for BAI problems have been developed that minimise sample complexity while providing statistical guarantees on the correct arm selection. As we observe, these algorithms can be computationally prohibitive. We exploit the fact that the reward process for each arm is well approximated by a Compound Poisson process and arrive at algorithms that are faster, with a small increase in sample complexity. We analyze the problem in an asymptotic regime as rarity of reward occurrence reduces to zero, and reward amounts increase to infinity. This helps illustrate the benefits of the proposed algorithm. It also sheds light on the underlying structure of the optimal BAI algorithms in the rare event setting.

1 INTRODUCTION

Online advertising is ubiquitous in present times. Its users include e-commerce platforms, mobile application developers, marketing agents and online retailers. Typically, an online advertiser has to decide amongst various product advertisements and choose the one with highest expected reward. Advertisers typically have a period of experimentation where they sequentially show competing advertisements to the users to arrive at those that elicit best response from each customer type (customers maybe clustered based on available information).

A key feature of online advertising is that while each advertisement maybe shown to a large number of customers, the click rates on advertisements are usually small. Typically, these maybe of order one in a thousand [1], and a very small percentage [2] of the users who click on an advertisement end up buying the product (known as the conversion rate). The conversion and click rates can vary significantly depending on the product category. For example, high-end products often have higher click rates but much lower conversion rates compared to standard products. Thus, a key characteristic of the problem is that rarer conversion rates often have very high rewards. Further, the seller would have an estimate of the price that their product(s) may sell for, along with an estimate of the volume of sales that may take place, it is also fair to assume that in practice upper bounds on rewards are known.

We study the problem of identifying the best advertisement to show to a customer type as a best arm identification (BAI) problem in the multi-armed bandit framework. The rarity of the reward probabilities, and the fact advertisements are shown to a large number of customers, may make the computational effort of popular existing adaptive algorithms prohibitive. On the other hand, these properties call for sensible aggregation based algorithms. In this paper, we observe that the rewards from large number of pulls from each arm can be well modelled as a compound Poisson process, significantly simplifying and speeding up the existing *optimal* algorithms.

To illustrate the proposed ideas clearly, we consider a simple stochastic BAI problem where agent is given a set of K unknown probability distributions (arms) that can be sampled sequentially. The agent's objective is to declare the arm with the highest mean with a pre-specified

[1] https://cxl.com/guides/click-through-rate/benchmarks/

[2] https://localiq.com/blog/search-advertising-benchmarks/.

Accepted for the 39^th^ Conference on Uncertainty in Artificial Intelligence (UAI 2023).

confidence level $1 - \delta$, while minimizing the expected number of samples (sampling complexity). In the literature, this is popularly known as the fixed-confidence setting, and the algorithms that provide $1 - \delta$ confidence guarantees are referred to as δ-correct.

Best arm identification problems are also popular in simulation community where these are better known as ranking and selection problems (for example see Goldsman [1983], Chan and Lai [2006]). Classical problem involves many complex simulation models of practical systems such as supply chain design, traffic network and so on, and the aim is to identify with high probability, the system with the highest expected reward, using minimum computational budget. In many systems, the performance measure of interest may correspond to a rare event, e.g., a manufacturing plant shut down probability, or computer system unavailability fraction. The algorithms that we propose here are also applicable in optimal computational resource allocation in simulating such systems.

Related literature: In the learning theory literature, Even-Dar et al. [2006] were amongst the first to consider the fixed confidence BAI problem. They proposed a successive elimination algorithm (see section F of supplementary material). Upper Confidence Bound (UCB) based algorithms were proposed in Auer et al. [2002], Jamieson et al. [2014], wherein the arm with highest confidence index is sampled. These algorithms usually stop when the difference between arm indices breaches a certain threshold (see Jamieson and Nowak [2014] for more details). The sample complexities of these algorithms was shown to match the lower bound developed by them to within a constant. Motivated by Bayesian approaches in Russo [2016], Jourdan et al. [2022] propose top-two algorithms that sequentially identify a challenger to the current empirical best arm and sample between the two with a pre-defined probability β. Although these algorithms are β-optimal [3], it's not clear how optimal β may be learnt, and thus they are sub-optimal. The sample complexity of these algorithms is typically analyzed in an asymptotic regime where $\delta \to 0$. Garivier and Kaufmann [2016] and Kaufmann et al. [2016] derived a lower bound on the sample complexity through a max-min formulation. Based on this lower bound, a Track-and-Stop algorithm (TS) was proposed for arm distributions restricted to single parameter exponential families (SPEF), and was shown to match the lower bound even to a constant (as $\delta \to 0$).Agrawal et al. [2019, 2020] extended the TS algorithms to more general distributions. The optimal TS algorithms in the literature, proceed sequentially. At each iteration, the observed empirical parameters are plugged into the lower bound max-min problem to arrive at prescriptive optimal sample allocations to each arm, that then guide the sample allocations. As is known, and as we observe, TS algorithms are computationally prohibitive[4], especially since in our rare advertising settings, the informative non-zero reward samples (those instances where users buy products) are rare. This motivates the paper's goal to arrive at computationally efficient algorithms that exploit the compound Poisson structure (see chapter 2 Ross [1995]) of the arm reward process, with a small increase in sample complexity.

Contributions: We develop a rarity framework where the reward success probabilities are modelled as a function of γ^α for arm dependent $\alpha > 0$ and γ is > 0 and small. The rewards are modelled to be of order $\gamma^{-\alpha}$ so that the expected rewards across arms are comparable (otherwise, we a-priori know arms with small or large expected rewards). We assume that arm specific upper bounds on rewards are available to us. In this framework, we propose a computationally efficient δ-correct algorithm that is nearly asymptotically optimal for small γ. This algorithm (approximate track and stop) is based on existing track and stop algorithms that are simplified through a Compound Poisson approximation to the bandit reward process. The Poisson approximation can be seen to be tight as $\gamma \to 0$ and we provide bounds on the deviations due to the Poisson approximation. Further, we give an asymptotically valid upper bound on the sample complexity illustrating that the increase in it is marginal compared to the computational benefit of the proposed alggorithm. The rarity structure helps us shed further light on the optimal sample allocations across arms in our BAI problem. We identify five different regimes depending on the rarity differences between the arms. Finally, we compare experimentally with the TS algorithm in Agrawal et al. [2020] for bounded random rewards. We find that for realistic rare event probabilities and reward structure, our algorithm is 6-12 times faster than the TS algorithm with a small increase (1-13 %) in sample complexity.

The rest of the paper is organized as follows: Section 2 formally introduces the problem, rare event setting and provides some background. Section 3 introduces the approximate problem, analyzes its deviations from the exact problem and gives the optimal weight asymptotics. Section 4 outlines the details of the Aproximate Track and Stop (TS(A)) algorithm, δ-correctness, sample complexity guarantee and computational benefits of the algorithm. Section 5 presents some experimental results and we conclude in Section 6. The proofs of various results and further technical details are furnished in the supplementary material.

2 MODELLING FRAMEWORK

Consider a K-armed bandit with each arm's distribution denoted by p_i, $i \in [K]$. We denote such a bandit instance by p. For any distribution η, let $\mu(\eta)$ denote its mean and $\mathrm{supp}(\eta)$

[3] see Jourdan et al. [2022] for definition

[4] UCB based BAI algorithms aren't instance optimal and incur large sample complexity in this setting.

denote its support. Further, let $KL(\eta,\kappa)=\mathbb{E}_\eta\big[\log\big(\frac{d\eta}{d\kappa}\big)\big]$ denote the Kullback-Leibler divergence between two measures η and κ, where E_η denotes the expectation operator under η. The agent's goal is to sequentially sample from these arms using a policy that at any sequential step t, may depend upon all the generated data before time t. The policy then stops at a random stopping time and declares an arm that it considers to have the highest mean. A sampling strategy, a stopping rule and a recommendation rule are together called a best arm bandit algorithm. A best arm bandit algorithm that correctly recommends the arm with the highest mean with probability at least $1-\delta$ (for a pre-specified $\delta\in(0,1)$) is said to be δ-correct.

This BAI problem has been well studied, and lower bounds on sample complexity under δ-correct algorithms have been developed along with algorithms that match the lower bound asymptotically as $\delta\to 0$. Below, we first state the lower bound in Theorem 2, and then briefly outline an algorithm that asymptotically matches it. The lower bounds were developed by Garivier and Kaufmann [2016]) for single parameter exponential family of distributions and were generalized to bounded and heavy-tailed distributions by Agrawal et al. [2020]. Let

$$\mathcal{K}^{L,B}_{inf}(\eta,x) := \min_{\substack{\text{supp}(\kappa)\subseteq[0,B]\\ \mu_\kappa\le x}} KL(\eta,\kappa) \qquad (1)$$

$$\mathcal{K}^{U,B}_{inf}(\eta,x) := \min_{\substack{\text{supp}(\kappa)\subseteq[0,B]\\ \mu_\kappa\ge x}} KL(\eta,\kappa). \qquad (2)$$

Henceforth, we suppress the dependence on B above to ease the presentation. This should not cause confusion in the following discussion. For brevity, we'll denote μ_{p_i} by μ_i for each $i\in[K]$. As is customary in the BAI literature, we assume that best arm is unique and without loss of generality, $\mu_1>\mu_i$ for $i\in[K]\backslash\{1\}$.

Theorem 5 in Agrawal et al. [2020]. *For our bandit problem, any δ-correct algorithm with stopping rule τ_δ, satisfies*

$$\mathbb{E}_p[\tau_\delta]\ge\frac{1}{V^*(p)}\log\Big(\frac{1}{2.4\delta}\Big),$$

where $V^*(p)$ is defined as

$$\max_{w\in\Sigma_K}\min_{i\neq 1}\inf_{x\in[\mu_i,\mu_1]} w_1\mathcal{K}^{L}_{inf}(p_1,x)+w_i\mathcal{K}^{U}_{inf}(p_i,x), \quad (3)$$

Σ_K being the K-dimensional probability simplex.

The lower bound suggests that each arm be sampled in proportion to the optimal weights w^* in (3). This idea guides the optimal Track and Stop (TS) algorithms that match the lower bound asymptotically as $\delta\to 0$. Typically, such algorithms have the following features: (see Garivier and Kaufmann [2016], Agrawal et al. [2020] for further details):

1. Arms are sampled sequentially in batches. At stage t, each arm is sampled at least order $\sqrt{t}$ times (this sub linear exploration ensures that no arm is starved).
2. Empirical distributions $\hat{p}_t$ are plugged into the lower bound 3 and is solved to determine the prescriptive proportions $\hat{w}_t$.
3. The algorithm then samples to closely track these proportions.
4. The algorithm stops when the log-likelihood ratio at stage m exceeds a threshold $\beta(m,\delta)$ (set close to $\log(1/\delta)$). At stage m, the log likelihood ratio equals

   $$\min_{b\neq k^*}\inf_{x\le y} N_{k^*}(m)\mathcal{K}^{L}_{inf}(\hat{p}_{k^*}(m),x) + N_b(m)\mathcal{K}^{U}_{inf}(\hat{p}_b(m),y),$$

   where k^* denotes the arm with the largest sample mean, each $N_a(m)$ denotes the samples of arm a amongst m samples.

As is apparent, the above algorithm involves repeatedly solving the lower bound problem, and this is computationally demanding, particularly when nonzero rewards are rare and occur with very low probabilities.

2.1 THE RARE EVENT SETTING

We now specialize the BAI setting to illustrate our rare event framework where the rewards from each arm take positive values with small probabilities. Further, while the expected rewards across arms are of the same order, the realized rewards and the associated probabilities may be substantially different.

Concretely, suppose that γ is a small positive value (say of order 10^{-2} or lower) and corresponding to each arm distribution $p_i,\ i\in[K]$, we have a rarity index $\alpha_i>0$. The support of arm i takes n_i distinct nonzero values, namely, $a_{ij}\gamma^{-\alpha_i}$, each with probability $p_{ij}\gamma^{\alpha_i}>0$ for $j\in[n_i],\ n_i\in\mathbb{N}$. Under each p_i, the realized reward takes value zero with probability close to 1. To summarize,

$$\mathbb{P}_{X\sim p_i}(X=a_{ij}\gamma^{-\alpha_i})=p_{ij}\gamma^{\alpha_i},\ j\in[n_i]$$
$$\mathbb{P}_{X\sim p_i}(X=0)=1-\sum_j p_{ij}\gamma^{\alpha_i}.$$

The arm means are given by $\mu_i=\sum_j a_{ij}p_{ij}$ and are independent of γ. We further assume that an upper bound $B_i\gamma^{-\alpha_i}$ for each arm i is known to the agent. We assume above that when arm i sees a large reward of order $\gamma^{-\alpha_i}$, it takes finitely many values. This keeps our analysis somewhat simpler and we deal with compound Poisson process for cumulative reward from each arm. It is easy to extend this to general distributions. However, the cumulative rewards from each arm would follow a Poisson random measure and Proposition 2.2 would generalize accordingly.

The above rarity framework brings out the benefits of the proposed approximations cleanly for small γ in our theoretical analysis. However, in executing the associated algorithm, we don't need to separately know the values of γ and each α_i.

2.2 THE POISSON APPROXIMATION OF KL DIVERGENCE

We motivate in this section the approximate form of KL divergence that we shall use. The following well-known result, shown in section A.5 of the supplementary material for completeness, is used to motivate our approximation.

Proposition 1. *Let $\tau_{ij}^{(1)}$ denote the minimum number of samples of arm i needed to see the reward $a_{ij}\gamma^{-\alpha_i}$, i.e. the first arrival time of the support point j. Similarly, let $\tau_{ij}^{(k)}$ be the k-th arrival time of support point j.*

Let $N_{ij}(t)$ be the number of times the reward $a_{ij}\gamma^{-\alpha_i}$ is returned by arm i in $\lceil t\gamma^{-\alpha_i}\rceil$ trials ($t \in \mathbb{R}$). Then as $\gamma \to 0$,

(a) $\mathbb{P}(\tau_{ij}^{(k)} > t\gamma^{-\alpha_i}) \to e^{-p_{ij}t}$,

(b) $N_{ij}(t) \xrightarrow{D} \mathrm{Poisson}(p_{ij}t)$.

Further for all support points, $\{\mathrm{Poisson}(p_{ij}t)\}_j$ is a collection of mutually independent random variables.

This implies that in rare event setting, the distribution of the counting process $N_{ij}(t)$ for each support point $a_{ij}\gamma^{-\alpha_i}$ is well-approximated by a Poisson process. We now argue that when γ is small enough, the KL divergence between arm distributions p_i and $\tilde{p}_i$ of same rarity can be approximated by a sum of KL divergences between independent Poisson variables.

Let $X_{1:m}$ and $\tilde{X}_{1:m}$ be two sets of i.i.d samples of size m from p_i and $\tilde{p}_i$ respectively. The corresponding measures are the product measures $p_i^{\otimes m}$ and $\tilde{p}_i^{\otimes m}$ respectively. By the tensorization property of KL-divergence, we have that

$$KL\big(p_i^{\otimes m}, \tilde{p}_i^{\otimes m}\big) = mKL(p_i, \tilde{p}_i) \tag{4}$$

In the following discussion we set $m = \lceil t\gamma^{-\alpha_i}\rceil$. Consider the vector-valued random variable $(N_{ij}(t))_{j\in[n_i]}$ and its counterpart $(\tilde{N}_{ij}(t))_{j\in[n_i]}$ under $\tilde{p}_i$. Note that they are functions of the samples $X_{1:\lceil t\gamma^{-\alpha_i}\rceil}, \tilde{X}_{1:\lceil t\gamma^{-\alpha_i}\rceil}$. Since we can also reconstruct a permutation of these samples from $(N_{ij}(t))_j$,$(\tilde{N}_{ij}(t))_j$, we have that

$$\begin{aligned}&KL\big(p_i^{\otimes\lceil t\gamma^{-\alpha_i}\rceil}, \tilde{p}_i^{\otimes\lceil t\gamma^{-\alpha_i}\rceil}\big)\\ &\qquad =KL\big(\nu((N_{ij}(t))_j), \nu((\tilde{N}_{ij}(t))_j)\big)\end{aligned}$$

where $\nu(A)$ is the measure of a random variable A. Now, by continuity of KL in γ and weak convergence of Proposition 1, it follows that for γ small enough:

$$\begin{aligned}&KL\big(p_i^{\otimes\lceil t\gamma^{-\alpha_i}\rceil}, \tilde{p}_i^{\otimes\lceil t\gamma^{-\alpha_i}\rceil}\big)\\ &\approx \sum_j KL(\mathrm{Poisson}(p_{ij}t), \mathrm{Poisson}(\tilde{p}_{ij}t))\\ &=t\left[\sum_j p_{ij}\log\left(\frac{p_{ij}}{\tilde{p}_{ij}}\right) + (\tilde{p}_{ij} - p_{ij})\right].\end{aligned}$$

for γ small enough. Then, combining the approximation above with the relation (4) gives

$$KL(p_i, \tilde{p}_i) \approx \gamma^{\alpha_i}\left[\sum_j p_{ij}\log\left(\frac{p_{ij}}{\tilde{p}_{ij}}\right) + (\tilde{p}_{ij} - p_{ij})\right]. \tag{5}$$

This approximation is used to motivate the approximate lower bound problem in the next section.

3 APPROXIMATE LOWER BOUND PROBLEM

For each i, if $B_i \notin \mathrm{supp}(p_i)$, let $\tilde{n}_i = n_i + 1$ and set $a_{i\tilde{n}_i} = B_i$, else $\tilde{n}_i = n_i$. The Poisson approximation of the KL divergence (see Section 2.2) suggests that in lieu of Equation (3), which is computationally expensive to solve, one could consider the following approximate problem when the rarity γ is small (the summations over j below correspond to $j \in [\tilde{n}_i]$).

$$\begin{aligned}V_a^*(p) := \max_{w\in\Sigma_K}\min_{i\neq 1}\inf_{\substack{\sum_j a_{ij}\tilde{p}_{ij}\geq\\ \sum_j ua_{1j}\tilde{p}_{1j}}}\Bigg\{&w_1\gamma^{\alpha_1}\left[\sum_j p_{1j}\log\left(\frac{p_{1j}}{\tilde{p}_{1j}}\right)\right.\\ &\left.+ (\tilde{p}_{1j} - p_{1j})\right] + w_i\gamma^{\alpha_i}\left[\sum_j p_{ij}\log\left(\frac{p_{ij}}{\tilde{p}_{ij}}\right) + (\tilde{p}_{ij} - p_{ij})\right]\Bigg\}.\end{aligned} \tag{6}$$

The minimization in 3 will now be replaced with the approximation in 5. Above, instead of allowing $\tilde{p}_i$ to have the support $[0, B_i\gamma^{-\alpha_i}]$, we limited its support to that of p_i extended to allow point $B_i\gamma^{-\alpha_i}$. This is justified in Sections A.1-A.2 of the supplementary material. The above representation suggest that in TS algorithm we do not need estimate γ and α's separately, instead the equation above suggests that only the relative rarity $\gamma^{\alpha_i-\alpha_k}$ for some fixed k is sufficient.

Let

$$\mathcal{P}_i := \inf_{x\in[\mu_i,\mu_1]} w_1\mathcal{K}_{inf}^L(p_1, x) + w_i\mathcal{K}_{inf}^U(p_i, x) \tag{7}$$

denote the inner minimisation problem in 3 and let

$$\begin{aligned}\mathcal{P}_{i,a} := \inf_{\substack{\sum_j a_{ij}\tilde{p}_{ij}\geq\\ \sum_j a_{1j}\tilde{p}_{1j}}} &w_1\gamma^{\alpha_1}\left[\sum_j p_{1j}\log\left(\frac{p_{1j}}{\tilde{p}_{1j}}\right) + (\tilde{p}_{1j} - p_{1j})\right]\\ &+w_i\gamma^{\alpha_i}\left[\sum_j p_{ij}\log\left(\frac{p_{ij}}{\tilde{p}_{ij}}\right) + (\tilde{p}_{ij} - p_{ij})\right]\end{aligned} \tag{8}$$

denote its approximation (above, we suppress the dependence on w_1 and w_i of $\mathcal{P}_i$ and $\mathcal{P}_{i,a}$). By approximating a reformulated version of $\mathcal{P}_i$ that uses the dual representations of $\mathcal{K}^L_{inf}$ and $\mathcal{K}^U_{inf}$ (following the approach used in Honda and Takemura [2010], Agrawal et al. [2020]), we can show that

$$\begin{aligned}\mathcal{P}_{i,a} =& w_1\gamma^{\alpha_1}\big[\sum_j p_{1j}\log(1+C^a_{1i}a_{1j}) - C^a_{1i}x^*_{i,a}\big] \\ &+ w_i\gamma^{\alpha_i}\big[\sum_j p_{ij}\log(1-C^a_i a_{ij}) + C^a_i x^*_{i,a}\big],\end{aligned} \tag{9}$$

where the quantities $x^*_{i,a}, C^a_{1i}, C^a_i$ (the qualifier 'a' reminds us these are for the approximate problem) are defined by the relations:

$$\begin{aligned}C^a_{1i}w_1\gamma^{\alpha_1} &= C^a_i w_i\gamma^{\alpha_i}, \\ x^*_{i,a} &= \sum_j \frac{a_{1j}p_{1j}}{1+a_{1j}C^a_{1i}}, \text{ and} \\ x^*_{i,a} &= \sum_j \frac{a_{ij}p_{ij}}{1-a_{ij}C^a_i}.\end{aligned} \tag{10}$$

Section A.4 of the supplementary material provides the step-by-step reformulation, as well as the results that have been used for it (Sections A.1-A.3 and A.5). The advantage of our reformulation is that the quantities C^a_{1i} and C^a_i have bounded well-defined limits and using (10), we can eliminate the dependence on x^*_i (whose behaviour is not as easy to analyze when $\gamma \to 0$).

The discussion in Section 2.2 also suggests that $\mathcal{P}_{i,a} \approx \mathcal{P}_i$ and hence, $V^*(p) \approx V^*_a(p)$. This is shown in the following theorem:

Theorem 1. *For each $i \in [K]$ and $w \in \Sigma_K$, $\mathcal{P}_i$, $\mathcal{P}_{i,a}$ are $\mathcal{O}(\gamma^{\max(\alpha_1,\alpha_i)})$. Furthermore, $\lim_{\gamma\to 0}\frac{\mathcal{P}_i}{\mathcal{P}_{i,a}} = 1$. In addition, there exist constants L_{1i} and L_i, independent of w, such that*

$$|\mathcal{P}_i - \mathcal{P}_{i,a}| \le L_{1i}w_1\gamma^{\min(2\alpha_1,\alpha_1+\alpha_i)} + L_i w_i \gamma^{\min(2\alpha_i,\alpha_i+\alpha_1)}.$$

Furthermore,

$$\begin{aligned}|V^*(p) - V^*_a(p)| \le \max_{i\ne 1}\max\big(&L_{1i}\gamma^{\min(2\alpha_1,\alpha_1+\alpha_i)}, \\ &L_i\gamma^{\min(2\alpha_i,\alpha_i+\alpha_1)}\big).\end{aligned}$$

The proof involves simplifying $\mathcal{P}_i$, $\mathcal{P}_{i,a}$ through Taylor expansions for small γ. It is given in the Sections A.4 and B of the supplementary material.

3.1 SOLVING THE APPROXIMATE LOWER BOUND

By definition we have that

$$V^*_a(p) = \max_{w\in\Sigma_K}\min_{i\ne 1}\mathcal{P}_{i,a}.$$

Further, we note that $\mathcal{P}_{i,a}$ is a concave function of w (infimum of linear function of w). Maxmin problems with this specific structure were studied in Glynn and Juneja [2004] (the caveat being that in our $\mathcal{K}_{inf}$ definitions in the underlying KL term, the first argument is fixed while we optimize over the second argument, while in Glynn and Juneja [2004], these orders are reversed. However, all the steps carry out identically). The optimal weights w^* are characterized in the following theorem:

Theorem 1 in Glynn and Juneja [2004]. *The optimal w^* of the maxmin problem 6 satisfies:*

$$\sum_{i=2}^{K}\frac{\partial\mathcal{P}_{i,a}(w^*)}{\partial w_1}\Big/\frac{\partial\mathcal{P}_{i,a}(w^*)}{\partial w_i} = 1, \tag{11}$$

and $\forall i \ne j$, $i, j \ne 1$,

$$\mathcal{P}_{i,a}(w^*) = \mathcal{P}_{j,a}(w^*). \tag{12}$$

These conditions are also sufficient.

We can use the above theorem to find closed form expressions (in terms of w^*) for $\mathcal{P}_{i,a}$ and $\frac{\partial\mathcal{P}_{i,a}(w^*)}{\partial w_j}$ using (9). As a starting point, we identify certain monotonicities present in (10), (11) and (12) to ease up the process of root-finding via bisection methods.

The equations defining C^a_{1i} and C^a_i imply that C^a_i is a decreasing function of C^a_{1i}. Mathematically, the implicit functions $g_i(r)$, defined for all $i \ne 1$ as

$$\sum_j \frac{a_{1j}p_{1j}}{1+g_i(r)a_{1j}} = \sum_j \frac{a_{ij}p_{ij}}{1-ra_{ij}}$$

are decreasing in r. The domain of g_i is chosen such that the RHS in the above equation is positive and finite.
The optimality equation (12) implies at the optimal weight w^*, each C^a_{1i}, $i > 2$, is an increasing function of C^a_{12}. More formally, the functions $\xi_i(s)$, $\forall i > 2$, implicitly defined through the equation:

$$\begin{aligned}&\sum_j p_{1j}\log(1+g_i(\xi_i)a_{1j}) + \frac{g_i(\xi_i)}{\xi_i}\sum_j p_{ij}\log(1-\xi_i a_{ij}) \\ =&\sum_j p_{1j}\log(1+g_2(s)a_{1j}) + \frac{g_2(s)}{s}\sum_j p_{2j}\log(1-sa_{2j}),\end{aligned}$$

are increasing in s. The domain of ξ_i is such that the RHS is well-defined. Finally, as a function of C^a_{12}, the LHS in the optimality equation 11 is also increasing. Mathematically this means that the functions , $\forall i \ne 1$,

$$\begin{aligned}h_i(s) :=& \Big(\sum_j p_{1j}\log(1+\xi_i a_{1j}) - \xi_i. \\ &\big[\sum_j \frac{a_{1j}p_{1j}}{1+a_{1j}\xi_i}\big]\Big)\Big(\sum_j p_{ij}\log(1-g_i(\xi_i)a_{ij}) \\ &+ g_i(\xi_i)\sum_j\big[\frac{a_{ij}p_{ij}}{1-a_{ij}g_i(\xi_i)}\big]\Big)^{-1}\end{aligned}$$

are increasing in s. These monotonicities enable one to solve for optimal weights in (6) through simple bisection methods. This is the source of computational benefit of solving (6) vis-a-vis (3). In (3), one has to solve either convex programs ($\mathcal{P}_i$) or a nonlinear system of four equations to arrive at the solution (see Section C of supplementary material).

This enables us to study the behaviour of w^* as $\gamma \to 0$. We set up some notation first.

Definition 1. Two positive valued functions of γ, $A(\gamma)$ and $B(\gamma)$, are said to be *asymptotically equivalent* if $0 < \liminf\limits_{\gamma\to 0}\frac{A(\gamma)}{B(\gamma)} \leq \limsup\limits_{\gamma\to 0}\frac{A(\gamma)}{B(\gamma)} < \infty$. We denote this by $A(\gamma) = \Theta(B(\gamma))$.

Let $\alpha_{\max} = \max_i \alpha_i$. The quantity $\zeta := \sum_{\substack{i\neq 1,\\ \alpha_i=\alpha_{max}}} h_i(\xi_i(0))$ also plays a role in governing the asymptotic behaviour of w^*.

Theorem (2) provides insight into the optimal weights in the lower bound problem as $\gamma \to 0$. We discuss its conclusions further in the next subsection.

Theorem 2. *The behaviour of w^* as $\gamma \to 0$ is described by the following five cases:*

Case 1: The best arm is not the rarest, $\alpha_{max} \neq \alpha_1$.

$$w_1^* = \Theta(\gamma^{\frac{\alpha_{max}-\alpha_1}{2}}),$$
$$w_i^* = \Theta(\gamma^{\alpha_{max}-\alpha_i}) \quad \textit{for all } i \neq 1.$$

Case 2: The best arm is uniquely the rarest, $\alpha_1 = \alpha_{max} > \alpha_i, i \neq 1$.

$$w_2^* = \Theta(\gamma^{\frac{\alpha_{max}-\alpha_2}{2}}),$$
$$w_i^* = \Theta(\gamma^{\alpha_{max}-\alpha_i}) \quad \textit{for all } i \neq 2.$$

Case 3: The best and second best arm only are the rarest, $\alpha_1 = \alpha_2 = \alpha_{max} > \alpha_i,\ \forall i \neq 1, 2$.

$$w_i^* = \Theta(\gamma^{\alpha_{max}-\alpha_i}), \textit{ for all } i.$$

Case 4: The best arm is the rarest but not uniquely, $\alpha_1 = \alpha_k = \alpha_{max} \geq \alpha_i,\ i \notin \{1, 2, k\}$, $\alpha_{max} > \alpha_2$ and $\zeta \leq 1$.

$$w_1^* = \Theta(\gamma^{\alpha_{max}-\alpha_1}),$$
$$w_i^* = \Theta(\gamma^{\alpha_{max}-\alpha_i}) \quad \textit{for all } i \neq 1.$$

Case 5: The best arm is the rarest but not uniquely, $\alpha_1 = \alpha_k = \alpha_{max} \geq \alpha_i,\ i \notin \{1, 2, k\}$, $\alpha_{max} > \alpha_2$ and $\zeta > 1$.

$$w_2^* = \Theta(\gamma^{\frac{\alpha_{max}-\alpha_2}{2}}),$$
$$w_i^* = \Theta(\gamma^{\alpha_{max}-\alpha_i}) \quad \textit{for all } i \neq 2.$$

Further, the asymptotic equivalence can be expressed by limits that are functions of parameters of the bandit problem.

Proof. See section C of supplementary material. □

Theorem 2 gives us insight into the behavior of the optimal weights w^* in (6). The results from the theorem rely on Lemma 1 below and are discussed further in Section 3.2.

By the fact that $V^*(p) \approx V_a^*(p)$ (Theorem 1) the optimal weights of actual maxmin problem also will show the same asymptotic behaviour. It is easy to see that substituting these optimal weights in $V^*(p)$ gives us an overall lower bound on the sample complexity as a scalar multiple of $\gamma^{\alpha_{max}}$.

3.2 DISCUSSION ON THEOREM 2

Without loss of generality let arm 2 be the one with the second highest mean. We further assume that $\mu_2 > \mu_i$ for $i \geq 3$.

Lemma 1. *In the maxmin problem (3), let $x_{i,e}^*(w^*)$ denote the minimizer of each $\mathcal{P}_i$ for the optimal weights w^*. Then, we have $x_i^*(w^*) \in [\mu_2, \mu_1]\ \ \forall i$.*

Remark 1. *t is well known that $x_{i,e}^*(w^*)$ lies within $[\mu_i, \mu_1]$. The Lemma 1 shows that $x_{i,e}^*(w^*) \geq \mu_2$.*

Proof of Lemma 1: We shall show this by contradiction. Suppose $x_{i,e}^*(w^*) < \mu_2$. Then, from the optimality conditions of w^* (similar to (11), (12)) we have, $\forall i \neq j,\ i, j \neq 1$:

$$\inf_{\mu_i' \geq \mu_1'} w_1^* KL(\mu_1, \mu_1') + w_i^* KL(\mu_i, \mu_i')$$
$$= \inf_{\mu_j' \geq \mu_1'} w_1^* KL(\mu_1, \mu_1') + w_j^* KL(\mu_j, \mu_j').$$

But we know that this minimization, for each $i \neq 1$, is attained uniquely by a bandit instance p' where the rest of the arms, except 1 and i, are the same as the original bandit instance in consideration, namely, p. Both the arms i and 1 have means $x_{i,e}^*(w^*)$ under p'. But the assumed hypothesis then implies that $x_{i,e}^*(w^*) = \mu_1' < \mu_2' = \mu_2$. That means p' is also in the set $\{\mu_2' \geq \mu_1'\}$ and hence

$$\inf_{\mu_i' \geq \mu_1'} w_1^* KL(\mu_1, \mu_1') + w_i^* KL(\mu_i, \mu_i')$$
$$> \inf_{\mu_2' \geq \mu_1'} w_1^* KL(\mu_1, \mu_1') + w_2^* KL(\mu_2, \mu_2').$$

However, this contradicts the necessary optimality conditions for w^*. Thus, $x_{i,e}^*(w^*) \geq \mu_2$. □

A similar result can also be shown for the approximate problem (6) (see Section D of supplementary material).

On Theorem 2. In the rare event setting, the non-zero samples from an arm are the informative samples, but they are quite rare. Any algorithm needs to sufficient see non-zero (informative) samples from at least some arms before it decides to stop.

As is well known, the lower bound problem in this testing of hypothesis setting corresponds to gaining enough evidence for each arm so that we can rule out alternative hypotheses. Specifically, the agent sees the data coming from more-or-less underlying distribution and is concerned that the true distribution comes from an alternative set, and what is seen is a large deviation from the data. Now, all else being equal, an arm with a rarer event has larger tendency towards large deviations. $x^*_{i,e}(w^*)$ in Lemma 1 corresponds to the mean of the most likely distribution under the alternate hypothesis.

In Case 1, $x^*_{i,e}(w^*)$ is close to μ_1 for each arm. The suboptimal arms see $O(1)$ informative samples while the best arm sees $O(\gamma^{-(\alpha_{max}-\alpha_1)})$ informative samples.

In Case 2, $x^*_{i,e}(w^*)$ is close to μ_2 for each arm. This necessitates additional samples for the second best arm in this case. The suboptimal arms see $O(1)$ informative samples just as the best arm while the second best sees $O(\gamma^{-(\alpha_{max}-\alpha_2)})$ informative samples.

In case 3, the noisiest distributions are arm 1 and arm 2 and bulk of the samples are given to them.

In case 4, $x^*_{i,e}(w^*)$ corresponding to the noisiest two arms lies strictly in (μ_2, μ_1) as $\gamma \to 0$ and the remaining suboptimal arms see $O(1)$ informative samples. This changes in Case 5, where now the noisiest arms are forced to meet at μ_2 due to Lemma 1. Arm 2 is given extra samples and sees $O(\gamma^{-(\alpha_{max}-\alpha_2)})$ informative samples.

4 TRACK AND STOP ALGORITHM

Our algorithm builds upon the Track and Stop (TS) algorithm proposed in Agrawal et al. [2019], Kaufmann et al. [2016]. We call it Track and Stop (A), to emphasize thatwe are solving an approximate problem. The algorithm solves the approximate maxmin problem 6, and samples according to the weights obtained. The calculation of the sampling weights happen in batches of size m. Let l denote the batch index. Within each batch we ensure that each arm gets at least $\sqrt{lm}$ samples. This is done in the same manner as Agrawal et al. [2019]. At the end of l-th batch, TS(A) evaluates the maximum likelihood ratio $Z_{k^*}(l)$ for the empirical best arm $k^*(l)$ and decides whether to stop or not. The likelihood ratio is given by:

$$Z_{k^*}(l) := \min_{b \neq k^*} \inf_{x \leq y} N_{k^*}(lm) \mathcal{K}^L_{inf}(\hat{p}_{k^*}(lm), x) + N_b(lm) \mathcal{K}^U_{inf}(\hat{p}_b(lm), y),$$

the same way as in Garivier and Kaufmann [2016] and Agrawal et al. [2019]. $\hat{p}(t)$ refers to the empirical bandit instance after t samples. $N_i(t)$ denotes to number of pulls of arm i after t samples. TS(A) stops when $Z_{k^*}(l) > \beta(lm, \delta)$, where $\beta(t, \delta)$ is a stopping threshold defined as

$$\beta(t, \delta) := \log\left(\frac{K-1}{\delta}\right) + 5\log(t+1) + 2.$$

Note that we are computing the maximum likelihood ratio by solving the $\mathcal{K}_{inf}$ problems exactly, and not approximately. Although it is relatively expensive to compute these quantities exactly, such computations occur only once for each l. The number of samples $N_i(t)$ for each arm i is influenced by the optimal weights that are obtained as solution to the approximate maxmin problem. The precise algorithmic details of TS(A) are given below.

Algorithm 1 TS(A) algorithm

```
Input: Confidence level $\delta$, Upper bounds $[B_i \gamma^{-\alpha_i}]_{i \in [K]}$.
Output: Arm recommendation $k^*$.

Generate $\lfloor \frac{m}{K} \rfloor$ samples for each arm.
$l \leftarrow 1$.
Compute the empirical bandit $\hat{p} = (\hat{p})_{i \in [K]}$.
$\hat{w}(\hat{p}) \leftarrow$ Compute weights according to (6).
$k^* \leftarrow \arg\max_{i \in [K]} \mathbb{E}[\hat{p}_i]$.
Compute $Z_{k^*}(l)$, $\beta(lm, \delta)$.
while $Z_{k^*}(l) \geq \beta(lm, \delta)$ do
    $s_i \leftarrow (\sqrt{(l+1)m} - N_i(lm))^+$.
    if $m \geq \sum_i s_i$ then
        Generate $s_i$ many samples for each arm $i$.
        Generate $(m - \sum_i s_i)^+$ i.i.d. samples from $\hat{w}(\hat{p})$.
        Let $Count(i)$ be occurrence of $i$ in these samples.
        Generate $Count(i)$ samples from each arm $i$.
    else
        $\hat{s}^* \leftarrow \arg\min_{\hat{s}, s_i \geq \hat{s}_i \geq 0} \max_i (s_i - \hat{s}_i)$.
        Generate $\hat{s}^*_i$ samples from each arm $i$.
    end if
    $l \leftarrow l + 1$
    Update empirical bandit $\hat{p}$.
    $k^* \leftarrow \arg\max_{i \in [K]} \mathbb{E}[\hat{p}_i]$.
    Update $Z_{k^*}(l)$, $\beta(lm, \delta)$.
    $\hat{w}(\hat{p}) \leftarrow$ Compute weights according to (6).
end while
return $k^*$.
```

Observe that in Algorithm 1 when we solve (6) we need an estimate of γ^{α_i}. Typically, this can be estimated either as the ratio of the known upper bounds of the support of each arm. Alternatively, this maybe estimated from the past sales data.

4.1 δ-CORRECTNESS AND SAMPLE COMPLEXITY OF TS(A)

The following theorem guarantees the δ-correctness and gives asymtptotic sample complexity bound for TS(A):

***Theorem 3.*.** *The TS(A) is a δ-correct algorithm with the* *following asymptotic sample complexity bound:*

$$\limsup_{\delta\to 0} \frac{\mathbb{E}_p[\tau_\delta]}{\log(1/\delta)} \le \frac{1}{V_{TS(A)}(p)} \qquad (13)$$

where $V_{TS(A)}(p) := \min_{i\neq 1}\mathcal{P}_i(\hat{w}^*(p))$. $\hat{w}^*(p))$ *denotes the optimal weights for the approx lower bound problem* $V_a^*(p)$.

See sections E and F in the supplementary material for a proof of Theorem 3. Note that by definition we have $V^*(p) \le V_{TS(A)}$ and hence we do suffer some loss in sample complexity vis-a-vis the TS algorithm. However, when γ is small, the difference is negligible as $w^*(p) \approx \hat{w}^*(p)$.

4.2 COMPUTATIONAL BENEFIT OF POISSON APPROXIMATION

The computational benefit of TS(A) vis-a-vis the exact algorithm, call it TS (E), is in how the approximate and exact lower bound problems are solved.

Let us first examine the number of operations required in finding the exact lower bound. In our implementation, we used Brent's method for one-dimensional optimization and the bisection method for root finding. To get a relative error of ϵ in Brent's method (see Chapter 4 in Brent [2013]) we require $\mathcal{O}\big(\log^2\big(\frac{1}{\epsilon}\big)\big)$ operations. The bisection method takes $\mathcal{O}\big(\log\big(\frac{1}{\epsilon}\big)\big)$ for a relative accuracy of ϵ. Lemma 2 (see Section A of the supplementary material) reduces the process of computing $\mathcal{K}_{inf}^L$ and $\mathcal{K}_{inf}^U$ to a root-finding procedure, causing said computations to take about $\mathcal{O}\big(\log\big(\frac{1}{\epsilon}\big)\big)$ operations. The inner optimization $\mathcal{P}_i$ is a convex optimization that requires $\mathcal{O}\big(\log^2\big(\frac{1}{\epsilon}\big)\big)$ operations. The outer optimization in (3) can be reduced to solving two sets of simultaneous root finding procedures and hence would take $\mathcal{O}\big(\log^2\big(\frac{1}{\epsilon}\big)\big)$. Thus, the total number of operations to solve the exact lower bound (3) is $\mathcal{O}\big(\log^5\big(\frac{1}{\epsilon}\big)\big)$.

In the approximate problem C_i, C_{1i}'s are the unknown variables, whose behaviour we analyze. Using g_i (section 3.1) to write C_i as a function of C_{1i} requires about $\mathcal{O}\big(\log\big(\frac{1}{\epsilon}\big)\big)$ operations for each such conversion using the bisection method. Then, each of the C_{1i} $(i \neq 2)$, are written as function of C_{12} through ξ_i. This again requires about $\mathcal{O}\big(\log\big(\frac{1}{\epsilon}\big)\big)$ operations for each such conversion. Finally the solution of C_{12} through h_i requires another factor of $\mathcal{O}\big(\log\big(\frac{1}{\epsilon}\big)\big)$. This gives the total required number of operations to be $\mathcal{O}\big(\log^3\big(\frac{1}{\epsilon}\big)\big)$. Thus, we are saving about $\mathcal{O}\big(\log^2\big(\frac{1}{\epsilon}\big)\big)$ by solving the approximate problem vis-a-vis the exact one.

5 NUMERICAL EXPERIMENTS

We compare the sample complexity and computational time between TS(A) and TS(E) algorithm proposed in Agrawal et al. [2020]. We make the comparison across different arms, γ and α structures at a confidence level $\delta = 0.01$. We choose the parameter $\gamma = 10^{-2}, 10^{-3}$ to reflect the typical rarities seen in the online ads scenario. We choose different configuration of the relative rarities α's to reflect some of the different regimes seen in Theorem 2. The tested configuration are given below:

(γ, α) configuration	Config. name
$(\gamma = 10^{-3}, \alpha = (1, 1, 1))$	Expt 1.
$(\gamma = 10^{-2}, \alpha = (1, 1.5, 2))$	Expt 2.
$(\gamma = 10^{-3}, \alpha = (1, 1, 1, 1, 1))$	Expt 3.
$(\gamma = 10^{-2}, \alpha = (2, 1.5, 2, 2.5, 1))$	Expt 4.

We run each algorithm for 100 sample paths and their average sample complexity and average computational time are reported in the Table 1 below. The algorithm for both TS(E) and TS(A) proceeds in batches of size $\gamma^{-\alpha_{\max}}$. Table 1 shows that for all experiments, TS(A) takes slightly more samples (1-13%) to stop and recommend an arm compared to TS. The computational savings of TS(A) is about $6-12$ times the TS algorithm. These simple experiments underscore the trade-off between sample complexity and computational time.

Experiment:	**Samples (m)**		**Runtime (s)**	
(γ,α)	**TS(E)**	**TS(A)**	**TS(E)**	**TS(A)**
Expt 1.	0.28	0.37	269.49	27.36
Expt 2.	0.45	0.47	45.47	2.74
Expt 3.	0.81	0.92	1016.29	144.08
Expt 4.	7.87	8.88	109.61	15.17

Table 1: *Comparison between the TS and TS(A) algorithms. Sample complexity is reported in million (m) samples. The computational runtime is reported in seconds (s).*

Experiment:	**Samples (m)**		**Runtime (s)**	
(γ,α)	**lilUCB**	**LUCB**	**lilUCB**	**LUCB**
Expt 1.	38.8	171.7*	8870	28200*
Expt 2.	162.2	137.7*	28230	34250*
Expt 3.	85.8	141*	17340	28590*
Expt 4.	204.8*	134.2*	37300*	34850*

Table 2: *Further comparison of TS(A) with lil-UCB and LUCB1.The superscript* $*$ *denotes those runs which took a long time (>10 hours) to stop. We report the stopped values for these runs.*

We conduct further comparisons with LilUCB (see Jamieson et al. [2014]) and LUCB (see Kalyanakrishnan et al. [2012]). Both these algorithms are well known in the BAI literature. These additional results are reported in Table 2. We see that both the TS algorithms are much better. The issue with UCB-index based algorithms like LilUCB, LUCB is that they have a dependence of σ^2(where σ is the sub-gaussianity parameter, see Appendix H) in the sample complexity upper bound. This translates to a dependence of $\gamma^{-2\alpha_{\max}}$ (since the upper bounds on rewards scale with $\gamma^{-\alpha_{\max}}$) while the TS(E) and TS(A) have an order dependence of only $\gamma^{-\alpha_{\max}}$, which is a significantly better sample complexity dependence.

Experiment:	**Samples (m)**		**Runtime (s)**	
(γ,α)	**m=200**	**m=500**	**m=200**	**m=500**
Expt 1.	0.39	0.38	4.15	6.51
Expt 2.	0.18	0.14	14.59	16.21
Expt 3.	0.28	0.25	62.93	84.13
Expt 4.	4.29	4.38	11.98	21.30

Table 3: *We increase the support size m of each bandit arm while holding the means fixed. The sample complexity increases with increasing support points.*

As noted in Section 2.1 the theory can be extended to continuum support. The experimental results reported in Table 1 were with a support size $m = 25$ per arm. Now, we increase the support size to $m = 200$ and $m = 500$, while the mean of the arms are held fixed. The results are presented in Table 3. We observe the sample complexity increases with increasing support points but the marginal increase is diminishing. This hints that the sample complexity is tending towards the one suggested by theory for the continuum support.

Experiment:	**Samples (m)**		**Runtime (s)**	
(γ,α)	$\gamma' = 0.95\gamma$	$\gamma' = 0.1\gamma$	$\gamma' = 0.95\gamma$	$\gamma' = 0.1\gamma$
Expt 1.	0.52	33.13	2.88	8.58
Expt 2.	0.72	31.9	4.12	8.5
Expt 3.	1.09	9.28	186.84	233.7
Expt 4.	8.23	1124.6	15.21	47.34

Table 4: Misspecified γ. Sample complexity is stable wrt the mis-specification.

The TS(A) algorithm requires (see Section 4.1) an estimate of the rarity γ^{α}. As the rarity can only be known only approximately we study the scenario where the parameter γ is mis-specified and hence the rarities are too. The results are presented in Table 4. We observe that the sample complexity is stable wrt mis-specification, with larger estimation errors leading to an increase in sample complexity.

6 CONCLUSION

The paper proposes a rarity framework to study the fixed confidence BAI problem relevant to online ad placement. In this framework the positive reward probabilities are small while the corresponding rewards are quite large. Consequently, the mean rewards are $\mathcal{O}(1)$.
We introduce a Poisson approximation to the standard lower bound problem and use it to motivate an algorithm that is computationally faster than the optimal TS algorithm at the cost of a small increase sample complexity. We also use this approximation to derive asymptotic optimal weights which give insight into the lower bound behaviour in the rare event setting. We observe this trade-off between sample complexity and computational time in our numerical experiments.

References

Shubhada Agrawal, Sandeep Juneja, and Peter Glynn. Optimal δ-correct best-arm selection for general distributions. *arXiv preprint arXiv:1908.09094*, 2019.

Shubhada Agrawal, Sandeep Juneja, and Peter Glynn. Optimal δ-correct best-arm selection for heavy-tailed distributions. In *Algorithmic Learning Theory*, pages 61–110. PMLR, 2020.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2):235–256, 2002.

Richard P Brent. *Algorithms for minimization without derivatives*. Courier Corporation, 2013.

Hock Peng Chan and Tze Leung Lai. Sequential generalized likelihood ratios and adaptive treatment allocation for optimal sequential selection. *Sequential Analysis*, 25(2): 179–201, 2006.

Eyal Even-Dar, Shie Mannor, Yishay Mansour, and Sridhar Mahadevan. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7(6), 2006.

A. Garivier and E. Kaufmann. Optimal best arm identification with fixed confidence. In *Conference on Learning Theory*, pages 998–1027. PMLR, 2016.

Peter Glynn and Sandeep Juneja. A large deviations perspective on ordinal optimization. In *Proceedings of the 2004 Winter Simulation Conference, 2004.*, volume 1. IEEE, 2004.

David Goldsman. Ranking and selection in simulation. Technical report, Institute of Electrical and Electronics Engineers (IEEE), 1983.

Junya Honda and Akimichi Takemura. An asymptotically optimal bandit algorithm for bounded support models. In *COLT*, pages 67–79. Citeseer, 2010.

Kevin Jamieson and Robert Nowak. Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting. In *2014 48th Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6. IEEE, 2014.

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

Marc Jourdan, Rémy Degenne, Dorian Baudry, Rianne de Heide, and Emilie Kaufmann. Top two algorithms revisited. *arXiv preprint arXiv:2206.05979*, 2022.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. Pac subset selection in stochastic multi-armed bandits. In *ICML*, volume 12, pages 655–662, 2012.

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

Sheldon M Ross. *Stochastic processes*. John Wiley & Sons, 1995.

Daniel Russo. Simple bayesian algorithms for best arm identification. In *Conference on Learning Theory*, pages 1417–1418. PMLR, 2016.