Training a Single Bandit Arm

Eren Ozbay **Vijay Kamble**
Department of Information and Decision Sciences
University of Illinois at Chicago
{eozbay3, kamble}@uic.edu

Abstract

In several applications of the stochastic multi-armed bandit problem, the traditional objective of maximizing the expected sum of rewards obtained can be inappropriate. Motivated by the problem of optimizing job assignments to train novice workers of unknown quality in labor platforms, we consider a new objective in the classical setup. Instead of maximizing the expected total reward from T pulls, we consider the vector of cumulative rewards earned from the K arms at the end of T pulls, and aim to maximize the expected value of the highest cumulative reward across the K arms. This corresponds to the objective of training a single, highly skilled worker using a limited supply of training jobs. For this new objective, we show that any policy must incur an instance-dependent asymptotic regret of $\Omega(\log T)$ (with a higher instance-dependent constant compared to the traditional objective) and an instance-independent regret of $\Omega(K^{1/3}T^{2/3})$. We then design an explore-then-commit policy, featuring exploration based on appropriately tuned confidence bounds on the mean reward and an adaptive stopping criterion, which adapts to the problem difficulty and achieves these bounds (up to logarithmic factors). Our numerical experiments demonstrate the efficacy of this policy compared to several natural alternatives in practical parameter regimes.

1 Introduction

The stochastic multi-armed bandit (MAB) problem (Lai and Robbins, 1985; Auer et al., 2002) presents a basic formal framework to study the exploration vs. exploitation tradeoff fundamental to online decision-making in uncertain settings. Given a set of K arms, each of which yields independent and identically distributed (i.i.d.) rewards over successive pulls, the goal is to adaptively choose a sequence of arms to maximize the expected value of the total reward attained at the end of T pulls. The critical assumption here is that the reward distributions of the different arms are a priori unknown. Any good policy must hence, over time, optimize the tradeoff between choosing arms that are known to yield high rewards (exploitation) and choosing arms whose reward distributions are yet relatively unknown (exploration). Over several years of extensive theoretical and algorithmic analysis, this classical problem is now quite well understood (see Lattimore and Szepesvári (2018), Slivkins (2019), and Bubeck and Cesa-Bianchi (2012) for a survey).

In this paper, we address a new objective in this classical setup. We consider the vector of cumulative rewards that have been earned from the different arms at the end of T pulls, and instead of maximizing the expectation of their *sum*, we aim to maximize the expected value of the *maximum* (max) of these cumulative rewards across the arms. This problem is motivated by several practical settings, as we discuss below.

1. **Training workers in online labor platforms.** An important operational objective of online labor platforms is to develop and maintain a reliable pool of high-quality workers to satisfy the demand for jobs. This is a challenging problem since, a) workers continuously leave the platform and hence new talent must be trained on an ongoing basis, b) the number of "training" jobs available to train the incoming talent is limited (for instance, this could result from a limited budget for the incentives offered to the clients for choosing novice workers), and c) the quality of the

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130. Copyright 2021 by the author(s).

workers is unknown: some workers are fast learners while some are slow, and efficient allocation of training jobs entails distinguishing between these types. At the core of this challenging operational question is the following problem. Given a limited number of training jobs, the platform must determine a policy to allocate these jobs to a set of novice workers to maximize some appropriate functional of their terminal skill levels. For a platform that seeks to offer robust service guarantees to its clients, simply maximizing the sum of the terminal skill levels across all workers may not be appropriate. A more appropriate objective is to maximize some q^{th} percentile skill level amongst the workers ordered by their terminal skills, where q is determined by the volume of demand for regular jobs: higher the demand for jobs, higher the q needed. Effectively, the skill levels of the lower skilled workers at the end of training do not matter, since there is not enough demand for regular jobs to assign to them anyway.

To address this problem, we can use the MAB framework: the set of arms is the set of novice workers, the reward of an arm is the random increment in the skill level of the worker after performing a job, and the number of training jobs available is T. Assuming that T is not too large, the random increments may be assumed to be i.i.d. over time. The mean of these increments can be interpreted as the unknown learning rate or the *trainability* of a worker. Given K workers, the goal is to adaptively allocate the jobs to these workers to maximize the smallest terminal skill level amongst the top $m \leq K$ (where $m \approx qK$) most terminally skilled workers. Our objective corresponds to the case where $m = 1$, and is a significant step towards solving this general problem.

2. **Training for external competitions.** Related to the above application, the objective we consider is also relevant to the problem of developing advanced talent within a region for participation in external competitions like Science Olympiads, the Olympic games, etc., with limited training resources. In these settings, only the terminal skill levels of those finally chosen to represent the region matter. The resources spent on others, despite resulting in skill advancement, are effectively wasteful. This feature is not captured by the sum objective, while it is effectively captured by the max objective, particularly in situations where one individual will finally be chosen to represent the region.

3. **Grooming an "attractor" product on e-commerce platforms.** E-commerce platforms typically feature very similar substitutes within a product category. For instance, consider a product like a tablet cover (e.g., for an iPad). Once the utility of a new product of this type becomes established (e.g., the size specifications of a new version of the iPad becomes available), several brands offering close to identical products serving the same purpose proliferate the marketplace. This proliferation is problematic to the platform for two reasons: a) customers are inundated by choices and may unnecessarily delay their purchase decision, thereby increasing the possibility of leaving the platform altogether (Settle and Golden, 1974; Gourville and Soman, 2005), and b) the heterogeneity in the purchase behavior resulting from the lack of a clear choice may complicate the problem of effectively managing inventory and delivery logistics. Given a budget for incentivizing customers to pick different products in the early exploratory phase where the qualities of the different products are being discovered, a natural objective for the platform is to "groom" a product to have the highest volume of positive ratings at the end of this phase. This product then becomes a clear choice for the customers. Our objective effectively captures this goal.

A key assumption we make in the paper is that the rewards for all arms are non-negative; this is motivated by training applications where rewards represent skill increments. Under this assumption, in the *full-information* setting where the reward distributions of the arms are known, we first show that the optimal policy for the max objective is identical to the one for the sum objective: one always pulls the arm with the highest mean reward (Proposition 1). This additionally implies that the optimal rewards under the two objectives are identical.

A standard approach in MAB problems is to design a policy that minimizes *regret*, i.e., the quantity of loss relative to the optimal full-information policy for a given objective over time. In the classical setting with the sum objective, it is well known that any policy must incur an instance-dependent asymptotic regret of $\Omega(\sum_{i \neq i^*}(\Delta_i \log T)/d_i)$ as $T \to \infty$ (Lai and Robbins, 1985). Here, $\Delta_i = \mu^* - \mu_i$, i.e., it is the difference between the highest mean reward μ^* belonging to the arm i^* and the mean reward μ_i of arm i; and d_i is a quantity that captures an appropriate notion of divergence between the reward distribution of arm i and the "closest" distribution within the space of possible distributions having a mean that is at least μ^*. Additionally, it is also well-known that any policy must incur and instance-independent regret of $\Omega(\sqrt{KT})$ in the worst-case over the set of possible bandit instances (Auer et al., 2002).

Since the optimal full-information reward is the same under the sum and the max objectives, and since the maximum of a set of non-negative numbers is always at most the sum of the numbers, any lower bound on the regret for the sum objective implies the same lower bound on the max objective. However, a key feature of the max objective is that the rewards earned from arms that do not eventually turn out to be the one yielding the highest cumulative reward are effectively a waste. Owing to this feature, we show that any policy must incur a *higher* instance-dependent regret of $\Omega(\sum_{i\neq i^*}(\mu^* \log T)/d_i)$ in this case (Theorem 1). Moreover, we show that an instance-independent regret of $\Omega(K^{1/3}T^{2/3})$ is inevitable in the worst-case (Theorem 2). Both these results rely on novel arguments that are a significant departure from those involved in proving the corresponding lower bounds for the sum objective.

Attaining these lower bounds requires algorithmic innovation. For the sum objective, well-performing policies are typically based on the principle of optimism in the face of uncertainty. A popular policy-class is the Upper Confidence Bound (UCB) class of policies (Agrawal, 1995; Auer et al., 2002; Auer and Ortner, 2010), in which a confidence interval is maintained for the mean reward of each arm, and at each time, the arm with the highest upper confidence bound is chosen. For a standard tuning of these intervals, this policy – termed UCB1 in literature due to (Auer et al., 2002) – guarantees an instance-dependent asymptotic regret of $\mathrm{O}(\log T)$ and a regret of $\mathrm{O}(\sqrt{KT\log T})$ in the worst case. With a more refined tuning, $\mathrm{O}(\sqrt{KT})$ can be achieved (Audibert and Bubeck, 2018; Lattimore, 2018).

For our max objective, directly using one of the above UCB policies can prove to be disastrous. To see this, suppose that all K arms have equal deterministic rewards. Then, UCB1 will pull each of the arms in a round-robin fashion until a total of T pulls, resulting in the highest terminal cumulative reward of $\mathrm{O}(T/K)$; whereas, a reward of $\Theta(T)$ is feasible by simply committing to an arbitrary arm from the start. This results in a $\Omega(T)$ regret in the worst case. Introducing randomness in the rewards doesn't change this observation: the result of a numerical experiment shown in Figure 1 suggests a $\Omega(T)$ regret on using UCB1 for a two-armed bandit problem with both arms having Bernoulli rewards with mean 0.5.

This observation suggests that any good policy must, at some point, stop exploring and permanently commit to a single arm. A natural candidate is the basic explore-then-commit (ETC) strategy, which uniformly explores all arms until some time that is fixed in advance, and then commits to the empirically best arm (Lattimore and Szepesvári, 2018; Slivkins, 2019). When each arm is chosen $(T/K)^{2/3}$ times in the exploration phase, this strategy can be shown to achieve a regret of $\mathrm{O}(K^{1/3}T^{2/3}\sqrt{\log K})$ relative to the sum objective (Slivkins, 2019). It is easy to argue that it achieves the same regret relative to the max objective. However, this policy is excessively optimized for the worst case where the means of all the arms are within $(K/T)^{1/3}$ of each other. When the arms are easier to distinguish, this policy's performance is quite poor due to excessive exploration. For example, consider a two armed bandit problem with Bernoulli rewards and means $(0.5, 0.5+\Delta)$, where $\Delta > 0$. For this fixed instance, ETC will pull both arms $\Omega(T^{2/3})$ times and hence incur a regret of $\Omega(T^{2/3})$ relative to our max objective. However, it is well known that UCB1 will not pull the suboptimal arm more than $\mathrm{O}(\log T/\Delta^2)$ times with high probability (Auer et al., 2002) and hence for this instance, UCB1 will incur an instance-dependent regret of only $\mathrm{O}(\log T)$, which could be much smaller if Δ is large. Thus, although the worst case regret of UCB1 is $\Omega(T)$ due to perpetual exploration, for a fixed bandit instance, its asymptotic performance is significantly better than ETC. This observation motivates us to seek a practical policy with a graceful dependence of performance on the difficulty of the bandit instance, and which will achieve both: the worst-case bound of ETC and the instance-dependent asymptotic bound of $\mathrm{O}(\log T)$.

We propose a new policy with an explore-then-commit structure, in which appropriately defined confidence bounds on the means of the arms are utilized to guide exploration, as well as to decide when to stop exploring. We call this policy Adaptive Explore-then-Commit (ADA-ETC). We show that ADA-ETC adapts to the problem difficulty by exploring less if appropriate, while attaining the same regret guarantee of $\mathrm{O}(K^{1/3}T^{2/3}\sqrt{\log K})$ attained by vanilla ETC in the worst case (Theorem 3). In particular, ADA-ETC guarantees an instance-dependent asymptotic regret of $\mathrm{O}(\log T)$ as $T \to \infty$, matching our instance-dependent lower bound upto a constant factor. Finally, our numerical experiments demonstrate that ADA-ETC results in significant improvements over the performance of vanilla ETC in easier settings, while never performing worse in difficult ones, thus corroborating our theoretical results. Our numerical results also demonstrate that naive ways of introducing adaptive exploration based on upper confidence bounds, e.g., simply using the upper confidence bounds of UCB1, may lead to no improvement over vanilla ETC for practical values of T and K.

1.1 Related literature

To the best of our knowledge, the objective we consider in this paper has not been studied before. We nevertheless note that buried in our objective is the goal of quickly identifying the arm with approximately the highest mean reward so that a substantial amount of time can be spent earning rewards from that arm (e.g., "training" a worker). This goal is related to the *pure exploration* problem in multi-armed bandits. Several variants of this problem have been studied, where the goal of the decision-maker is to either minimize the probability of misidentification of the optimal arm given a fixed budget of pulls (Audibert et al., 2010; Kaufmann et al., 2016; Carpentier and Locatelli, 2016); or minimize the expected number of pulls to attain a fixed probability of misidentification, possibly within an approximation error (Even-Dar et al., 2002; Mannor and Tsitsiklis, 2004; Even-Dar et al., 2006; Karnin et al., 2013; Vaidhiyan and Sundaresan, 2017; Jamieson et al., 2014; Kaufmann et al., 2016); or to minimize the expected suboptimality (called "simple regret") of a recommended arm after a fixed budget of pulls (Bubeck et al., 2009, 2011; Carpentier and Valko, 2015). Extensions to settings where multiple good arms are needed to be identified have also been considered (Bubeck et al., 2013; Kalyanakrishnan et al., 2012; Zhou et al., 2014; Kaufmann and Kalyanakrishnan, 2013). The critical difference from these approaches is that in our scenario, the budget of T pulls must not only be spent on identifying an approximately optimal arm but also on earning rewards on that arm. Hence any choice of apportionment of the budget to the identification problem, or a choice for a target for the approximation error or probability of misidentification within a candidate policy, is a priori unclear and must arise *endogenously* from our primary objective.

The fact that focusing on one arm in the long-run is prudent for our objective makes it seem related to the line of work on bandits with switching costs, where there is a cost incurred for switching from one arm to another (Cesa-Bianchi et al., 2013; Dekel et al., 2014). Another related line of work is on *batched* bandits, which imposes a constraint that the policy must split the arm pulls into a small number of batches (Perchet et al., 2016; Gao et al., 2019). However, we note that our objective does not simply amount to keeping the number of switches or batches low; it also matters how "spread apart" the switches are. To enforce this point, we note that the algorithm of Cesa-Bianchi et al. (2013), which restricts the number of switches/batches to $\mathrm{O}(\log\log T)$ while attaining $\tilde{\mathrm{O}}(\sqrt{T})$ regret for the sum objective, incurs a worst-case regret of $\Theta(T)$ for our max objective: Figure 1 shows this for $K = 2$ arms with Bernoulli(0.5) rewards.

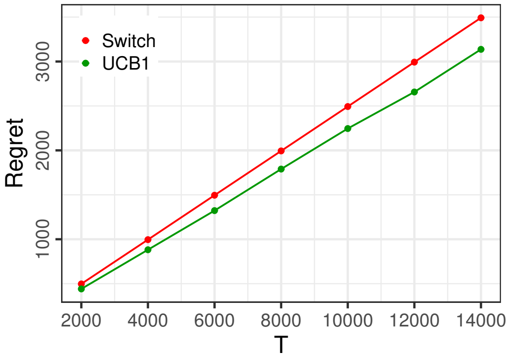


Figure 1: Expected regret for the max objective under UCB1 and the algorithm of Cesa-Bianchi et al. (2013), referred to here as *Switch*, for $K = 2$ arms, each with Bernoulli(0.5) rewards.

2 Problem Setup

Consider the stochastic multi-armed bandit (MAB) problem parameterized by the number of arms, which we denote by K; the length of the decision-making horizon (the number of discrete times/stages), which we denote by T; and the probability distributions for arms $1, \ldots, K$, denoted by $\nu_1, \ldots, \nu_K$, respectively. We assume that the rewards are non-negative and their distributions have a bounded support, assumed to be $[0, 1]$ without loss of generality (although, this latter assumption can be easily relaxed to allow, for instance, σ-Sub-Gaussian distributions with bounded σ). We define $\mathcal{V}$ to be the set of all K-tuples of distributions for the K arms having support in $[0, 1]$. Let $\mu_1, \ldots, \mu_K$ be the means of the distributions. Without loss of generality, unless specified otherwise, we assume that $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_K$ for the remainder of the discussion. The distributions of the rewards from the arms are unknown to the decision-maker. We denote $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$ and $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K)$. We also define $\Delta_i = \mu_1 - \mu_i$ for $i \in \{1, \ldots, K\}$.

At each time, the decision-maker chooses an arm to play and observes a reward. Let the arm played at time t be denoted as I_t and the reward be denoted as X_t, where X_t is drawn from the distribution ν_{I_t}, independent from the previous actions and observations. The history of actions and observations at any time $t \geq 2$ is denoted as $\mathcal{H}_t = (I_1, X_1, I_2, X_2, \ldots, I_{t-1}, X_{t-1})$, and $\mathcal{H}_1$ is defined to be the empty set ϕ. A *policy* π of the decision-maker is a sequence of mappings $(\pi_1, \pi_2, \ldots, \pi_T)$, where π_t maps every possible history $\mathcal{H}_t$ to an arm I_t to be played at time t. Let Π_T denote the set of all such policies.

For an arm i, we denote n_t^i to be the number of times this arm is played until and including time t, i.e., $n_t^i = \sum_{s=1}^{t} \mathbb{1}_{\{I_s = i\}}$. We also denote U_n^i to be the reward observed from the n^{th} pull of arm i. $(U_n^i)_{n\in\mathbb{N}}$ is thus a sequence of i.i.d. random variables, each distributed as ν_i. Note that the definition of U_n^i implies that we have $X_t = U^{I_t}_{n^{I_t}_t}$. We further define $\overline{U}^i_t \triangleq \sum_{n=1}^{n^i_t} U^i_n$ to be the cumulative reward obtained from arm i until time t.

Once a policy π is fixed, then for all $t = 1, \ldots, T$, I_t, X_t, and n^i_t for all $i \in \{1, \ldots, K\}$, become well-defined random variables. We consider the following notion of reward for a policy π:

$$\mathcal{R}_T(\pi, \boldsymbol{\nu}) = \mathrm{E}\big(\max\big(\overline{U}^1_T, \overline{U}^2_T, \ldots, \overline{U}^K_T\big)\big). \qquad (1)$$

In words, the objective value attained by the policy is the expected value of the largest cumulative reward across all arms at the end of the decision making horizon. When the reward distributions $\nu_1, \ldots, \nu_K$ are known to the decision-maker, then for a large T, the best reward that the decision-maker can achieve is

$$\sup_{\pi \in \Pi_T} \mathcal{R}_T(\pi, \boldsymbol{\nu}).$$

A natural candidate for a "good" policy when the reward distributions are known is the one where the decision-maker exclusively plays arm 1 (the arm with the with the highest mean), attaining an expected reward of $\mu_1 T$. Let us denote $\mathcal{R}^*_T(\boldsymbol{\nu}) \triangleq \mu_1 T$. One can show that, in fact, this is the best reward that one can achieve in our problem.

Proposition 1. *For any bandit instance $\boldsymbol{\nu} \in \mathcal{V}$, $\sup_{\pi \in \Pi_T} \mathcal{R}_T(\pi, \boldsymbol{\nu}) = \mathcal{R}^*_T(\boldsymbol{\nu})$.*

The proof is presented in Section A in the Appendix. This shows that the simple policy of always picking the arm with the highest mean is optimal for our problem. Next, we denote the *regret* of any policy π to be

$$\mathrm{Reg}_T(\pi, \boldsymbol{\nu}) = \sup_{\pi \in \Pi_T} \mathcal{R}_T(\pi, \boldsymbol{\nu}) - \mathcal{R}_T(\pi, \boldsymbol{\nu}).$$

In the rest of the paper, we focus on two objectives. The first is to design a policy $\pi_T \in \Pi_T$, which attains an asymptotically optimal instance-dependent (i.e., $\boldsymbol{\nu}$ dependent) bound on $\mathrm{Reg}_T(\pi_T, \boldsymbol{\nu})$, simultaneously for (almost) all instances $\boldsymbol{\nu} \in \mathcal{V}$ as $T \to \infty$. The second objective is to design a policy $\pi_T \in \Pi_T$, which achieves the smallest regret in the worst-case over all distributions $\boldsymbol{\nu} \in \mathcal{V}$, i.e., the one that solves the optimization problem:

$$\mathrm{Reg}^*_T \triangleq \inf_{\pi \in \Pi_T} \sup_{\boldsymbol{\nu} \in \mathcal{V}} \mathrm{Reg}_T(\pi, \boldsymbol{\nu}),$$

where Reg^*_T denotes the *minmax (or the best worst-case) regret*. In the remainder of the paper, we design a *single* policy that attains the first objective to within a constant factor and the second objective to within a logarithmic factor.

3 Lower Bounds

We first provide an instance-dependent $\Omega(\log T)$ asymptotic lower bound on the regret. We let $\mathcal{M}$ be the set of distributions with support in $[0, 1]$. For $\nu \in \mathcal{M}$, and $\mu \in [0, 1]$, define $d_{\inf}(\nu, \mu, \mathcal{M}) = \inf_{\nu' \in \mathcal{M}} \{\mathrm{D}(\nu, \nu') : \mu(\nu') > \mu\}$, where $\mu(\nu)$ denotes the mean of distribution ν, and $\mathrm{D}(\nu, \nu')$ is the Kullback-Leibler (KL) divergence between the distributions ν and ν'. $d_{\inf}(\nu, \mu, \mathcal{M})$ is thus the smallest KL divergence between the distribution ν and any other distribution in $\mathcal{M}$ whose mean is at least μ.

We say that a sequence of policies $(\pi_T)_{T \in \mathbb{N}}$, where $\pi_T \in \Pi_T$ for all $T \in \mathbb{N}$, is *consistent* for a class $\mathcal{V} = \mathcal{M}^K$ of stochastic bandits, if for all $\boldsymbol{\nu} \in \mathcal{V}$ such that there is a unique arm with the highest mean reward, and for any $p > 0$, we have that $\lim_{T \to \infty} \mathrm{Reg}_T(\pi_T, \boldsymbol{\nu})/T^p = 0$. We then have the following result.

Theorem 1. *Consider a class $\mathcal{V} = \mathcal{M}^K$ of K-armed stochastic bandits and let $(\pi_T)_{T \in \mathbb{N}}$ be a consistent sequence of policies for $\mathcal{V}$. Then, for all $\boldsymbol{\nu} \in \mathcal{V}$ such that the optimal arm is unique,*

$$\liminf_{T \to \infty} \frac{\mathrm{Reg}_T(\pi_T, \boldsymbol{\nu})}{\log(T)} \geq \sum_{i \neq k^*} \frac{\mu^*}{d_{\inf}(\nu_i, \mu^*, \mathcal{M})},$$

where k^* is the optimal arm with the highest mean μ^*.

The proof of Theorem 1 is presented in Section B in the Appendix. The result has an intuitive explanation. For convenience, we denote $d_i = d_{\inf}(\nu_i, \mu^*, \mathcal{M})$. Similar to the proof of the lower bound for the sum objective (Lai and Robbins, 1985), we can show that for any consistent sequence of policies, each suboptimal arm i must be pulled $\Omega(\log T/d_i)$ number of times in expectation. However, unlike the sum objective where each such pull yields a mean reward of μ_i and results in an expected regret of Δ_i, for the max objective, each such pull is wasteful and results in an expected regret of μ^*.

Despite this intuitive explanation of the result, the proof is not straightforward. In particular, showing that each suboptimal arm i must be pulled $\log T/d_i$ times in expectation doesn't directly allow us to account for a regret contribution of $\mu^* \log T/d_i$ from arm i. This is because, in the full-information setting, with a (relatively high) probability of $\log T/(Td_i)$, one can choose to pull a suboptimal arm i for all the T time periods, thus ensuring that it gets pulled $\log T/d_i$ times in expectation and at the same time resulting in an expected reward contribution of $\mu_i \log T/d_i$ and hence a regret contribution of $(\mu^* - \mu_i) \log T/d_i = \Delta_i \log T/d_i$. To get around this difficulty, we prove a stronger result: we show that for each $\alpha \in (0, 1]$, a suboptimal arm i must be pulled $\alpha \log T/d_i$ times in expectation until time T^α (Proposition 2 in the Appendix). We then argue that the probability of a suboptimal arm being the one with the highest cumulative reward cannot be too high for any consistent sequence of policies, and thus the best way to satisfy the stronger set of lower bounds on the number of pulls for the suboptimal arms in terms of minimizing regret is to chalk these pulls as wasted.

We next show that for our objective, a regret of $\Omega(K^{1/3}T^{2/3})$ is inevitable in the worst case.

Theorem 2. *Suppose that $K < T$. Then,* $\mathrm{Reg}^*_T \geq \Omega((K-1)^{1/3}T^{2/3})$.

The proof is presented in Section C in the Appendix. Informally, the argument for the case of $K = 2$ arms is as follows. Consider two bandits with Bernoulli rewards, one with the mean rewards $(1/2+1/T^{1/3}, 1/2)$, and the other with mean rewards $(1/2+1/T^{1/3}, 1/2+2/T^{1/3})$. Then until time $\approx T^{2/3}$, no algorithm can reliably distinguish between the two bandits. Hence, until this time, either $\Omega(T^{2/3})$ pulls are spent on arm 1 irrespective of the underlying bandit, or $\Omega(T^{2/3})$ pulls are spent on arm 2 irrespective of the underlying bandit. In both cases, the algorithm incurs a regret of $\Omega(T^{2/3})$, essentially because of wasting $\Omega(T^{2/3})$ pulls on a suboptimal arm that could have been spent on earning reward on the optimal arm. This latter argument is not entirely complete, however, since it ignores the possibility always picking a suboptimal arm until time T, in which case spending time on the suboptimal arm in the first $\approx T^{2/3}$ time periods was not wasteful. However, even in this case, one incurs a regret of $\approx T \times (1/T^{1/3}) = \Omega(T^{2/3})$. Thus a regret of $\Omega(T^{2/3})$ is unavoidable. Our formal proof builds on this basic argument to additionally determine the optimal dependence on K.

4 Adaptive Explore-then-Commit (ADA-ETC)

We now define an algorithm that we call Adaptive Explore-then-Commit (ADA-ETC), specifically designed for our problem. It is formally defined in Algorithm 1. The algorithm can be simply described as follows. After choosing each arm once, choose the arm with the highest upper confidence bound, until there is an arm such that (a) it has been played at least $\tau = \lceil T^{2/3}/K^{2/3} \rceil$ times, and (b) its empirical mean is higher than the upper confidence bounds on the means of all other arms. Once such an arm is found, commit to this arm until the end of the decision horizon.

The upper confidence bound is defined in Equation 2. In contrast to its definition in UCB1, it is tuned to eliminate wasteful exploration and to allow stopping early if appropriate. We enforce the requirement that an arm is played at least τ times before committing to it by defining a trivial "lower confidence bound" (Equation 3), which takes value 0 until the arm is played less than τ times, after which both the upper and lower confidence bounds are defined to be the empirical mean of the arm. The stopping criterion can then be simply stated in terms of these upper and lower confidence bounds (Equation 4): stop and commit to an arm when its lower confidence bound is strictly higher than the upper confidence bounds of all other arms (this can never happen before τ pulls since the rewards are non-negative).

Algorithm 1: Adaptive Explore-then-Commit (ADA-ETC)

Input: K arms with horizon T.

Define: Let $\tau = \lceil \frac{T^{2/3}}{K^{2/3}} \rceil$. For $n \geq 1$, let $\bar{\mu}^i_n$ be the empirical average reward from arm i after n pulls, i.e., $\bar{\mu}^i_n = \frac{1}{n}\sum_{s=1}^n U^i_s$. Also, for $n \geq 1$, define,

$$\mathrm{UCB}^i_n = \bar{\mu}^i_n + \sqrt{\frac{4}{n}\log\left(\frac{T}{Kn^{3/2}}\right)}\mathbb{1}_{\{n<\tau\}}. \tag{2}$$

$$\mathrm{LCB}^i_n = \bar{\mu}^i_n - \bar{\mu}^i_n \mathbb{1}_{\{n<\tau\}}. \tag{3}$$

Also, for $t \geq 1$, let n^i_t be the number of times arm i is pulled until and including time t.

Procedure:

```
• Explore Phase: From time $t = 1$ until $t = K$, pull each arm once. For $K < t \leq T$:
    1. Identify $L_t \in \arg\max_{i \in [K]} \mathrm{LCB}^i_{n^i_{t-1}}$, breaking ties arbitrarily. If
           $\mathrm{LCB}^{L_t}_{n^{L_t}_{t-1}} > \max_{i \in [K]: i \neq L_t} \mathrm{UCB}^i_{n^i_{t-1}}$,    (4)
       then define $i^* \triangleq L_t$, break, and enter the Commit phase. Else, continue to Step 2.
    2. Identify $E_t \in \arg\max_{i \in [K]} \mathrm{UCB}^i_{n^i_{t-1}}$, breaking ties arbitrarily. Pull arm $E_t$.
• Commit Phase: Pull arm $i^*$ until time $t = T$.
```

Note that the collapse of the upper and lower confidence bounds to the empirical mean after τ pulls ensures that each arm is not pulled more than τ times during the Explore phase. This is because choosing this arm to explore after τ pulls would imply that its upper confidence bound = lower confidence bound is higher than the upper confidence bounds for all other arms, which means that the stopping criterion has been met and the algorithm has committed to the arm.

Remark 1. *A heuristic rationale behind the choice of the upper confidence bound is as follows. Consider a suboptimal arm whose mean is smaller than the highest mean by* Δ*. Let* P_e *be the probability that this arm is misidentified and committed to in the Commit phase. Then the expected regret resulting from this misidentification is approximately* $P_e \Delta T$*. Since we want to ensure that the regret is at most* $\mathrm{O}(T^{2/3}K^{1/3})$ *in the worst-case, we can tolerate a* P_e *of at most* $\approx K^{1/3}/(\Delta T^{1/3})$*. Unfortunately,* Δ *is not known to the algorithm. However, a reasonable proxy for* Δ *is* $1/\sqrt{n}$*, where* n *is the number of times the arm has been pulled. This is because it is right around* $n \approx 1/\Delta^2$*, when the distinction between this arm and the optimal arm is expected to occur. Thus a good (moving) target for the probability of misidentification is* $\delta_n \approx (K^{1/3}n^{1/2})/T^{1/3}$*. This necessitates the* $\sqrt{\log(1/\delta_n)} \approx \sqrt{\log(T/(Kn^{3/2}))}$ *scaling of the confidence interval in Equation 2. In contrast, our numerical experiments show that utilizing the traditional scaling of* $\sqrt{\log T}$ *as in UCB1 results in significant performance deterioration. Our tuning is reminiscent of similar tuning of confidence bounds under the "sum" objective to improve the performance of UCB1; see Audibert and Bubeck (2018); Lattimore (2018); Auer and Ortner (2010).*

Remark 2. *Instead of defining the lower confidence bound to be* 0 *until an arm is pulled* τ *times, one may define a non-trivial lower confidence bound to accelerate commitment, perhaps in a symmetric fashion as the upper confidence bound. However, this doesn't lead to an improvement in the regret bound. The reason is that if an arm looks promising during exploration, then eagerness to commit to it is imprudent, since if it is indeed optimal then it is expected to be chosen frequently during exploration anyway; whereas, if it is suboptimal then we preserve the option of eliminating it by choosing to not commit until after* τ *pulls. Thus, to summarize, ADA-ETC eliminates wasteful exploration primarily by reducing the number of times suboptimal arms are pulled during exploration through the choice of appropriately aggressive upper confidence bounds, rather than by being hasty in commitment.*

Let ADA-ETC$_{K,T}$ denote the implementation of ADA-ETC using K and T as the input for the number of arms and the time horizon, respectively. We characterize the regret guarantees achieved by ADA-ETC$_{K,T}$ in the following result.

Theorem 3 (ADA-ETC)**.** *Let* $K < T$*. Consider a* $\boldsymbol{\nu} \in \mathcal{V}$ *such that the optimal arm is unique and relabel arms so that* $\mu_1 > \mu_2 \geq \cdots \geq \mu_K$*. Then the expected regret of* ADA-ETC$_{K,T}$ *is upper bounded as:*[1]

$$\begin{aligned}
&\mathrm{Reg}_T(\text{ADA-ETC}_{K,T}, \boldsymbol{\nu}) \\
&\leq \underbrace{\mu_1 \sum_{i=2}^{K} \min\left(\kappa_i, \tau\right) + \mu_1 \tau \sum_{i=2}^{K} \min(2, \frac{648K}{T\Delta_i^3})}_{\text{Regret contribution from wasted pulls in the Explore phase}} \\
&+ \underbrace{\sum_{i=2}^{K} \exp(-\frac{\tau\Delta_i^2}{2}) T\Delta_i + \sum_{i=2}^{K} \min(1, \frac{320K}{T\Delta_i^3}) T\tilde{\Delta}_i}_{\text{Regret contribution from misidentification in the Commit phase}} \ ,
\end{aligned}$$

where $\tau = \lceil \frac{T^{2/3}}{K^{2/3}} \rceil$*,* $\tilde{\Delta}_i = \Delta_i - \Delta_{i-1}$ *and* $\kappa_i = \frac{10}{\Delta_i^2} + \frac{16}{\Delta_i^2} \log^+\left(\frac{T\Delta_i^3}{K}\right) + \frac{24}{\Delta_i^2}\sqrt{\log^+\left(\frac{T\Delta_i^3}{K}\right)}$*. In the worst case, we have*

$$\sup_{\boldsymbol{\nu} \in \mathcal{V}} \mathrm{Reg}_T(\text{ADA-ETC}_{K,T}, \boldsymbol{\nu}) \leq \mathrm{O}(K^{1/3}T^{2/3}\sqrt{\log K}).$$

The proof of Theorem 3 is presented in Section D in the Appendix. Theorem 3 features an instance-dependent regret bound and a worst-case bound of $\mathrm{O}(K^{1/3}T^{2/3}\sqrt{\log K})$. The first two terms in the instance-dependent bound arise from the wasted pulls during the Explore phase. Under vanilla Explore-then-Commit, to obtain near-optimality in the worst case, every arm must be pulled τ times in the Explore phase (Slivkins, 2019). Hence, the expected regret from the Explore phase is $\Omega(K\tau) = \Omega(T^{2/3}K^{1/3})$ irrespective of the instance. On the other hand, our bound on this regret depends on the instance and can be significantly smaller than $K\tau$ if the arms are easier to distinguish. In particular, for a fixed K and ν (with $\Delta_2 > 0$), the regret from exploration (and the overall regret) is $\mathrm{O}(\sum_{i \geq 2} 16\mu_1 \log T/\Delta_i^2)$ under ADA-ETC as opposed to $\Omega(T^{2/3}K^{1/3})$ under ETC as $T \to \infty$. This shows that ADA-ETC attains the instance-dependent lower bound on regret of Theorem 1 up to a constant factor.

The next two terms in our instance-dependent bound arise from the regret incurred due to committing to a suboptimal arm, which can be shown to be $\mathrm{O}(K^{1/3}T^{2/3}\sqrt{\log K})$ in the worst case, thus matching the guarantee of ETC. The first of these terms is not problematic since it is the same as the regret arising under ETC. The second term arises due to the inevitably increased misidentifications occurring due to stopping early in adaptive versions of ETC. If the confidence bounds are aggressively small, then this term increases. In ADA-ETC, the upper confidence bounds used in exploration are tuned to be as small as possible while ensuring that this term is no larger than $\mathrm{O}(K^{1/3}T^{2/3})$ in the worst case (see Remark 1). Thus,

[1] We define $\log^+(a) = \log(\max(a, 1))$ for $a > 0$.

our tuning of the Explore phase ensures that the performance gains during exploration does not come at the cost of higher worst-case regret (in the leading-order) due to misidentification.

Remark 3. *It is possible to show that using the confidence bounds of UCB1 under ADA-ETC results in the same asymptotic instance-dependent regret bound of* $\mathrm{O}(\log T)$ *and an instance-independent regret bound of* $\mathrm{O}(K^{1/3}T^{2/3}\sqrt{\log K})$ *in the worst case. However, for fixed* T *and* K*, the bounds derived for ADA-ETC, as defined, have an improved dependence on the instance owing to the reasons mentioned in Remark 1. As we shall see in Section 5, this results in significant performance gains for practical values of* T *and* K*. Optimizing finite* T *performance is particularly important, since in training applications, the assumption of skill increments being i.i.d. is not expected to hold when* T *is large.*

5 Experiments

Benchmark Algorithms. We compare the performance of ADA-ETC with five algorithms described in Table 1. UCB1 never stops exploring and pulls the arm with the highest upper confidence bound at each time step, while ETC pulls arms in a round-robin fashion and commits to the arm with the highest empirical mean after each arm has been pulled τ times. NADA-ETC and UCB1-s have the same algorithmic structure as ADA-ETC: they explore based on upper confidence bounds and commit if the lower confidence bound of an arm rises above upper confidence bounds for all other arms. They differ from ADA-ETC in how the upper and lower confidence bounds are defined. Both NADA-ETC and UCB1-s use UCB1's upper confidence bound, but they differ in their lower confidence bounds. These definitions are presented in Table 1.

SUCC is an adaptation of the well-known "successive elimination" algorithm of Even-Dar et al. (2006) for best-arm identification, which finds the best arm within a probability of error of δ in a sample efficient manner. This algorithm proceeds in rounds and samples every active arm once in each round, eliminating arms based on their empirical performance. In our adaptation, we set $\delta = (K/T)^{1/3}$ so that the expected regret in case of failure is at most $\delta T = T^{2/3}K^{1/3}$ in the worst-case. We further force the algorithm to commit to the an active arm with the highest empirical mean after τ rounds have elapsed.

Instances. We let $\nu_i \sim \mathrm{Bernoulli}(\mu_i)$, where μ_i is uniformly sampled from $[\alpha, 1-\alpha]$ for each arm in each instance. We sample two sets of instances, each of size 500, with $\alpha \in \{0, 0.4\}$. The regret for an algorithm for each instance is averaged over 500 runs to estimate the expected regret. We vary $K \in \{2, 5, 10, 15, 20, 25\}$ and $T \in \{100, 200, 300, 400, 500\}$. The average regret over the 500 instances under different algorithms and settings is presented in Figure 2 and Figure 3.

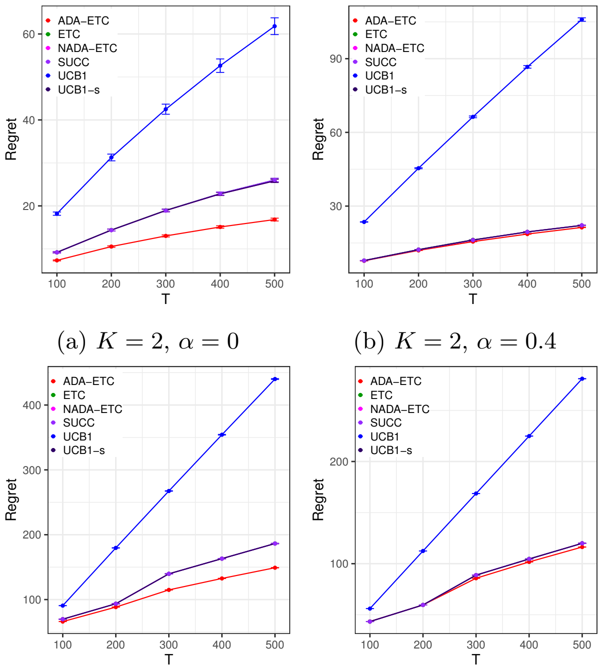
Table 1: Benchmark Algorithms

ADA-ETC	$\mathrm{UCB}_n^i = \bar{\mu}_n^i + \sqrt{\frac{4}{n}\log\left(\frac{T}{Kn^{3/2}}\right)}\mathbb{1}_{\{n<\tau\}}$ $\mathrm{LCB}_n^i = \bar{\mu}_n^i - \bar{\mu}_n^i\mathbb{1}_{\{n<\tau\}}$
NADA-ETC	$\mathrm{UCB}_n^i = \bar{\mu}_n^i + \sqrt{\frac{4}{n}\log(T)}\mathbb{1}_{\{n<\tau\}}$ $\mathrm{LCB}_n^i = \bar{\mu}_n^i - \bar{\mu}_n^i\mathbb{1}_{\{n<\tau\}}$
SUCC	$\mathrm{UCB}_n^i = *$ $\mathrm{LCB}_n^i = *$
ETC	$\mathrm{UCB}_n^i = *$ $\mathrm{LCB}_n^i = *$
UCB1	$\mathrm{UCB}_n^i = \bar{\mu}_n^i + \sqrt{\frac{4}{n}\log(T)}$ $\mathrm{LCB}_n^i = *$
UCB1-s	$\mathrm{UCB}_n^i = \bar{\mu}_n^i + \sqrt{\frac{4}{n}\log(T)}\mathbb{1}_{\{n<\tau\}}$ $\mathrm{LCB}_n^i = \bar{\mu}_n^i - \sqrt{\frac{4}{n}\log(T)}\mathbb{1}_{\{n<\tau\}}$

Discussion. ADA-ETC shows the best performance uniformly across all settings, although there are settings where its performance is similar to ETC. As anticipated, these are settings where either (a) $\alpha = 0.4$, in which case, the arms are expected to be close to each other and hence adaptivity in exploring has little benefits, or (b) T/K is relatively small, due to which τ is small. In these latter situations, the exploration budget of τ is expected to be exhausted for almost all arms under ADA-ETC, yielding in performance similar to ETC, e.g., if $K = 25$ and $T = 100$, then $\tau = \lceil 4^{2/3} \rceil = 3$, i.e., a maximum of only three pulls can be used per arm for exploring. When α is smaller, i.e., when arms are easier to distinguish, or when τ is large, the performance of ADA-ETC is significantly better than that of ETC. This illustrates the gains from the adaptivity of exploration under ADA-ETC.

Furthermore, we observe that the performances of SUCC, UCB1-s and NADA-ETC are essentially the same as ETC for the ranges of T and K we consider.[2] This important observation suggests that naively adding adaptivity to exploration, e.g., based on UCB1's upper confidence bounds, may not improve upon the performance of ETC in finite parameter settings, and appropriate refinement of the confidence bounds is crucial to the gains of ADA-ETC in these settings. Finally, we note that UCB1 performs quite poorly, thus demonstrating the importance of introducing an appropriate stopping criterion for exploration.

[2] Further experiments show that increasing T results in UCB1-s and NADA-ETC eventually outperforming ETC as well as SUCC.



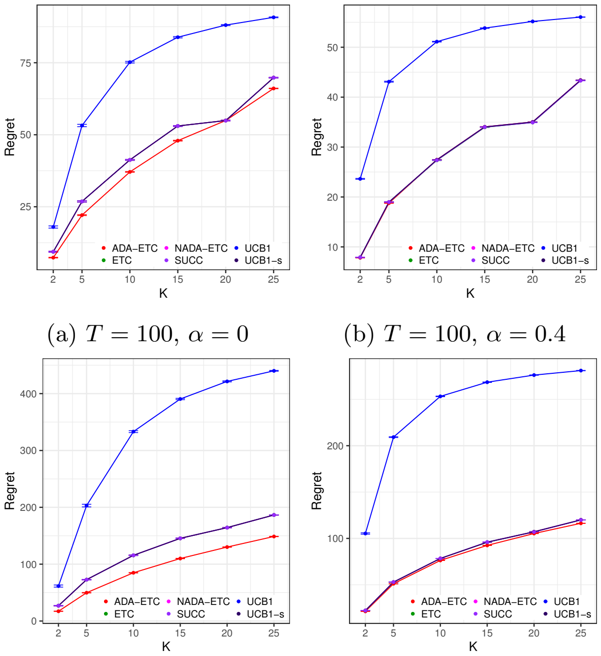
(a) $K = 2$, $\alpha = 0$

(b) $K = 2$, $\alpha = 0.4$

(c) $K = 25$, $\alpha = 0$

(d) $K = 25$, $\alpha = 0.4$

Figure 2: Performance comparison of ADA-ETC for varying values of T. The performances of SUCC, UCB1-s and NADA-ETC are identical to ETC.



(a) $T = 100$, $\alpha = 0$

(b) $T = 100$, $\alpha = 0.4$

(c) $T = 500$, $\alpha = 0$

(d) $T = 500$, $\alpha = 0.4$

Figure 3: Performance comparison of ADA-ETC for varying values of K. The performances of SUCC, UCB1-s and NADA-ETC are identical to ETC.

6 Conclusion and Future directions

In this paper, we proposed and offered a near-tight analysis of a new objective in the classical stochastic MAB setting, of optimizing the expected value of the maximum of cumulative rewards across arms. From a theoretical perspective, although the current analysis of ADA-ETC is tight, it is unclear whether the extraneous (compared to the lower bound) $\sqrt{\log K}$ factor from the upper bound can be eliminated via a more refined algorithm design. Additionally, our assumption that the rewards are i.i.d. over time, while appropriate for the application of qualifying an attractor product for e-commerce platforms, may be a limitation in the context of worker training, especially in settings where the number of training jobs available is large. It would be interesting to study our objective in settings that allow rewards to decrease over time; such models, broadly termed as *rotting bandits* (Heidari et al., 2016; Levine et al., 2017; Seznec et al., 2019), have attracted recent focus in literature as a part of the study of the more general class of MAB problems with non-stationary rewards (see, for instance, Besbes et al. (2014, 2019)). This literature has so far only focused on the traditional sum objective.

More importantly, our paper presents the possibility of studying a wide variety of new objectives under existing online learning setups motivated by training applications, where the traditional objective of maximizing the total rewards is inappropriate. A natural generalization of our objective is the optimization of other functionals of the vector of cumulative rewards, e.g., maximizing the m^{th} highest cumulative reward, which is relevant to online labor platforms as we mentioned in the Section 1, or the optimization of $\mathcal{L}^p$ norm of the vector of cumulative rewards for $p > 0$, which has natural fairness interpretations in the context of human training (the traditional objective corresponds to the $\mathcal{L}^1$ norm, while our objective corresponds to the $\mathcal{L}^\infty$ norm). More generally, one may consider multiple skill dimensions, with job types that differ in their impact on these dimensions. In such settings, a similar variety of objectives may be considered driven by considerations such as fairness, diversity, and focus.

References

Agrawal, R. (1995). Sample mean based index policies with O(log n) regret for the multi-armed bandit problem. *Advances in Applied Probability*, pages 1054–1078.

Audibert, J.-Y. and Bubeck, S. (2018). Minimax policies for adversarial and stochastic bandits.

Audibert, J.-Y., Bubeck, S., and Munos, R. (2010).

Best arm identification in multi-armed bandits. In *COLT*, pages 41–53.

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256.

Auer, P. and Ortner, R. (2010). Ucb revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*, 61(1-2):55–65.

Besbes, O., Gur, Y., and Zeevi, A. (2014). Stochastic multi-armed-bandit problem with non-stationary rewards. In *Advances in neural information processing systems*, pages 199–207.

Besbes, O., Gur, Y., and Zeevi, A. (2019). Optimal exploration–exploitation in a multi-armed bandit problem with non-stationary rewards. *Stochastic Systems*, 9(4):319–337.

Bubeck, S. and Cesa-Bianchi, N. (2012). Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *arXiv preprint arXiv:1204.5721*.

Bubeck, S., Munos, R., and Stoltz, G. (2009). Pure exploration in multi-armed bandits problems. In *International conference on Algorithmic learning theory*, pages 23–37. Springer.

Bubeck, S., Munos, R., and Stoltz, G. (2011). Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412(19):1832–1852.

Bubeck, S., Wang, T., and Viswanathan, N. (2013). Multiple identifications in multi-armed bandits. In *International Conference on Machine Learning*, pages 258–265.

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.

Carpentier, A. and Valko, M. (2015). Simple regret for infinitely many armed bandits. In *International Conference on Machine Learning*, pages 1133–1141.

Cesa-Bianchi, N., Dekel, O., and Shamir, O. (2013). Online learning with switching costs and other adaptive adversaries. In *Advances in Neural Information Processing Systems*, pages 1160–1168.

Dekel, O., Ding, J., Koren, T., and Peres, Y. (2014). Bandits with switching costs: T 2/3 regret. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 459–467. ACM.

Even-Dar, E., Mannor, S., and Mansour, Y. (2002). Pac bounds for multi-armed bandit and markov decision processes. In *International Conference on Computational Learning Theory*, pages 255–270. Springer.

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

Gao, Z., Han, Y., Ren, Z., and Zhou, Z. (2019). Batched multi-armed bandits problem. *arXiv preprint arXiv:1904.01763*.

Gourville, J. T. and Soman, D. (2005). Overchoice and assortment type: When and why variety backfires. *Marketing science*, 24(3):382–395.

Heidari, H., Kearns, M. J., and Roth, A. (2016). Tight policy regret bounds for improving and decaying bandits.

Jamieson, K., Malloy, M., Nowak, R., and Bubeck, S. (2014). lil'ucb: An optimal exploration algorithm for multi-armed bandits. In *Conference on Learning Theory*, pages 423–439.

Kalyanakrishnan, S., Tewari, A., Auer, P., and Stone, P. (2012). Pac subset selection in stochastic multi-armed bandits. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 227–234.

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

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

Kaufmann, E. and Kalyanakrishnan, S. (2013). Information complexity in bandit subset selection. In *Conference on Learning Theory*, pages 228–251.

Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22.

Lattimore, T. (2018). Refining the confidence level for optimistic bandit strategies. *The Journal of Machine Learning Research*, 19(1):765–796.

Lattimore, T. and Szepesvári, C. (2018). Bandit algorithms. *preprint*.

Levine, N., Crammer, K., and Mannor, S. (2017). Rotting bandits. In *Advances in neural information processing systems*, pages 3074–3083.

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

Perchet, V., Rigollet, P., Chassang, S., Snowberg, E., et al. (2016). Batched bandit problems. *Annals of Statistics*, 44(2):660–681.

Settle, R. B. and Golden, L. L. (1974). Consumer perceptions: Overchoice in the market place. *ACR North American Advances*.

Seznec, J., Locatelli, A., Carpentier, A., Lazaric, A., and Valko, M. (2019). Rotting bandits are no harder than stochastic ones. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2564–2572.

Slivkins, A. (2019). Introduction to multi-armed bandits. *arXiv preprint arXiv:1904.07272*.

Vaidhiyan, N. K. and Sundaresan, R. (2017). Learning to detect an oddball target. *IEEE Transactions on Information Theory*, 64(2):831–852.

Zhou, Y., Chen, X., and Li, J. (2014). Optimal pac multiple arm identification with applications to crowdsourcing. In *International Conference on Machine Learning*, pages 217–225.