Mitigating Disparity while Maximizing Reward: Tight Anytime Guarantee for Improving Bandits

Vishakha Patil[1] , **Vineet Nair**[2] , **Ganesh Ghalme**[3] and **Arindam Khan**[1]
[1]Indian Institute of Science, Bangalore
[2]Arithmic Labs
[3]Indian Institute of Technology, Hyderabad
patilv@iisc.ac.in, vineet@arithmic.com, ganeshghalme@ai.iith.ac.in, arindamkhan@iisc.ac.in

Abstract

We study the Improving Multi-Armed Bandit (IMAB) problem, where the reward obtained from an arm increases with the number of pulls it receives. This model provides an elegant abstraction for many real-world problems in domains such as education and employment, where decisions about the distribution of opportunities can affect the future capabilities of communities and the disparity between them. A decision-maker in such settings must consider the impact of her decisions on future rewards in addition to the standard objective of maximizing her cumulative reward at any time. We study the tension between two seemingly conflicting objectives in the horizon-unaware setting: a) maximizing the cumulative reward at any time, and b) ensuring that arms with better long-term rewards get sufficient pulls even if they initially have low rewards. We show that, surprisingly, the two objectives are aligned with each other. Our main contribution is an *anytime* algorithm for the IMAB problem that achieves the *best possible cumulative reward* while ensuring that the *arms reach their true potential* given sufficient time. Our algorithm mitigates the initial disparity due to lack of opportunity and continues pulling an arm until it stops improving. We prove the optimality of our algorithm by showing that a) any algorithm for the IMAB problem, no matter how utilitarian, must suffer $\Omega(T)$ policy regret and $\Omega(k)$ competitive ratio with respect to the optimal offline policy, and b) the competitive ratio of our algorithm is $O(k)$.

1 Introduction

Machine Learning (ML) algorithms are increasingly being used to make or assist critical decisions that affect communities (or, people) in areas such as education, employment, and loan lending. The distribution of opportunities in such critical domains can affect the future abilities of the impacted communities. Consequently, ML algorithms can influence the disparity between different communities. Though a significant amount of research has been aimed at ensuring fairness in ML algorithms, most of this work has focused on static settings without considering the long-term impacts of algorithmic decisions over time [Barocas *et al.*, 2019; Hardt *et al.*, 2016].

Multi-Armed Bandits (MAB) is a classical framework that captures sequential decision-making in uncertain environments [Robbins, 1952]. In MAB, a decision-maker pulls one of k arms at each time step and obtains a reward determined by a function or a reward distribution corresponding to that arm. The goal of the decision-maker is to maximize the total reward over multiple time steps when the reward function is not known to her a priori. A variant of MAB, called the Improving MAB (IMAB) model, was introduced by [Heidari *et al.*, 2016] to model the long-term impacts of algorithmic decisions on the underlying population. In IMAB, the arms represent communities (or, people), and pulling an arm corresponds to allocating opportunities to the corresponding community. The instantaneous reward obtained from pulling an arm is the current ability of the community in utilizing the opportunity. This reward *improves* with the number of pulls the arm receives, which imitates the likely improvement in abilities of communities given more opportunities. Inspired by the motivating examples and numerous studies on human learning, the reward functions in IMAB are assumed to be bounded, monotonically increasing, and having decreasing marginal returns (diminishing returns) [Son and Sethi, 2006].

In this work, we initiate the study of IMAB in the horizon-unaware (anytime) setting. The key technical challenge of this problem is the seemingly conflicting set of objectives: we wish to maximize the cumulative reward at any time and ensure that arms with high long-term but low short-term rewards also get pulled sufficiently often. The IMAB problem has been previously studied in the horizon-aware setting with asymptotic regret guarantees [Heidari *et al.*, 2016] (see Section 1.1). However, their non-asymptotic performance guarantee is not good (see Section 1.1). Further, in many practical applications, the time horizon is not known to the algorithm beforehand. In contrast to horizon-aware algorithms, anytime algorithms do not know the time horizon beforehand and hence cannot tailor their decisions to the given time horizon. Thus, an anytime algorithm must perform well for any finite time horizon without having prior knowledge of it, which poses interesting technical challenges. Due to its theoretical and practical significance, the design of anytime algorithms has been of prime interest to the MAB research community

(e.g., the popular UCB1 algorithm for stochastic MAB [Auer *et al.*, 2002a]).

The nature of the IMAB model and its applications also call for an investigation of any IMAB algorithm through the lens of fairness. Fairness through awareness [Dwork *et al.*, 2012] is a well-accepted notion of fairness that enforces that *similar* individuals or communities be treated similarly. In the IMAB model, one could quantify similarity (or equivalently, disparity) based on the arms' current rewards (abilities). However, we argue that such a blind comparison may be fallacious. For instance, historical marginalization could lead to differences in the abilities of different individuals or communities to perform a given task, for example, the racial gap observed in SAT scores [V. Reeves and Halikias, 2017]. One way of mitigating such differences that has been studied in the MAB literature [Li *et al.*, 2019; Patil *et al.*, 2021], is via *affirmative action*, where the decision-maker allocates some opportunities to individuals based on attributes such as their race, gender, caste, etc. Such policies have been in place for decades in some countries (see reservation system in India [Sahoo, 2009]), while they are banned in several US states [J. Baker, 2019]. Another popular notion of fairness in the MAB literature is meritocratic fairness [Joseph *et al.*, 2016], where arms are compared solely based on their current rewards. In IMAB however, meritocracy would identify individuals that are gifted early and provide them more opportunities which would suppress the growth of *late bloomers*, i.e., the individuals that would go on to perform well had they been given more opportunities. This detrimental effect of meritocracy has also been observed in the real world; for example, the education system in Singapore [Staff, 2018].

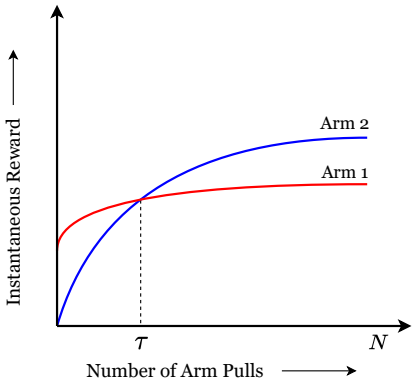

Figure 1: Two-armed IMAB Instance

We give a simple example to highlight the challenges in the anytime IMAB problem. Figure 1 shows a two-armed IMAB instance. We emphasize here that the reward of an arm changes only when the arm is pulled. That is, the x-axis denotes the number of arm pulls of an arm and not the time horizon. As the figure shows, arm 1 is an early gifted arm and arm 2 is a late bloomer. Here, a myopic decision-maker that pulls arms based only on the instantaneous rewards will seldom pull arm 2. An algorithm that majorly plays arm 1 may obtain a high cumulative reward for shorter horizons. However, for larger horizons, this algorithm could perform poorly. Additionally, an algorithm that mostly plays arm 1 will increase the disparity between the two arms. This highlights the challenges faced by an anytime algorithm in balancing exploitation (pull arm 1) and exploration (pull arm 2) such that the total reward is maximized.

Our Results. Our contributions in this paper are twofold. First, we contribute strong theoretical results to the long line of literature on non-stationary bandits, particularly rested bandits, where the rewards obtained from an arm can change when pulled [Besbes *et al.*, 2014; Levine *et al.*, 2017]. We refer the reader to Section 1.1 for a discussion on why existing algorithms for the non-stationary MAB problem do not work for the IMAB problem. Second, we make an important conceptual and technical contribution to the study of fairness in the IMAB problem. We study the IMAB problem in the horizon-unaware setting with the following objective: how does a decision-maker maximize her reward while ensuring the participants (arms) are provided with sufficient opportunities to improve and reach their true potential?

Our first result shows that any algorithm for the IMAB problem, how much ever utilitarian, suffers $\Omega(T)$ regret and has competitive ratio $\Omega(k)$ (Theorem 2). Our main contribution is an efficient anytime algorithm (Algorithm 1) which has a competitive ratio of $O(k)$ for the IMAB problem in the horizon-unaware case (Theorem 4).[1] An interesting and important property of our proposed algorithm is that it continues pulling an arm until it reaches its true potential (Theorem 5), thus mitigating the disparity that existed due to lack of opportunity. We note that this is not accomplished by imposing any fairness constraints but by establishing that it is in the best interest of a decision-maker that wishes to have good anytime regret, to enable arms to achieve their true potential (see example in Figure 1). The proofs of Theorems 4 and 5 require intricate analysis (see Sections 4.2 and 4.3). Our theoretical guarantees rest crucially on several important and non-trivial properties that our algorithm satisfies (e.g., see Lemmas 6, 9, and 10) and are the key technical contributions of our paper. We also analyze the performance of the round-robin (RR) algorithm. We show that while RR gives equal opportunity to all arms, its competitive ratio is $\Theta(k^2)$ and hence, is sub-optimal for the decision maker (Theorem 3). Finally, we further highlight the significance our theoretical results via experiments.

1.1 Related Work

The IMAB problem was introduced by [Heidari *et al.*, 2016], who studied the horizon-aware IMAB problem. Our work differs from theirs in two key aspects. First, they study the horizon-aware setting and their algorithm uses the knowledge of T at every time step. In particular, if their algorithm has run for some time steps with horizon set as T_1, and it is then decided to run it for some more rounds for a horizon $T_2 > T_1$, then the algorithm would need to be restarted from the first time step. Importantly, for two different horizons T_1 and T_2, the sequence of arm pulls by this algorithm could vary significantly. In contrast, our algorithm does not have prior

[1] Informally, the competitive ratio (Definition 2) is the worst-case ratio of the reward of the offline optimal to that of the algorithm.

knowledge of T and must work well for any stopping time T. Second, they provide an *asymptotically* sub-linear regret bound in terms of instance-dependent parameters.[2] On the other hand, our results hold for *any finite time horizon* T. In fact, in Appendix G, we show that for any time horizon T there is an instance for which the algorithm of [Heidari *et al.*, 2016] will have competitive ratio $\Omega(k^2)$ which is similar to RR. We note here that our results are with respect to two standard performance metrics in the MAB literature, policy regret [Arora *et al.*, 2012; Heidari *et al.*, 2016] and competitive ratio [Immorlica *et al.*, 2019; Daniely and Mansour, 2019; Andrew *et al.*, 2013] (see Section 2 for the definitions).

IMAB is a variant of the non-stationary MAB problem. Non-stationary MAB problems have been extensively studied in the literature under various assumptions [Tekin and Liu, 2012; Levine *et al.*, 2017]. The work of [Lindner *et al.*, 2021] shows that existing non-stationary MAB algorithms such as R-EXP3 [Besbes *et al.*, 2014], DUCB, and SW-UCB [Garivier and Moulines, 2011] and popular MAB algorithms such as greedy-selection and EXP3 [Auer *et al.*, 2002b] do not perform well for the IMAB problem even when the time horizon is known a priori. In recent parallel work, [Metelli *et al.*, 2022] have studied the stochastic variant of the IMAB problem. However, they provide only a $O(T)$ regret guarantee for the problem in the horizon-unaware setting and no competitive ratio analysis. We hope that the ideas in our paper and [Metelli *et al.*, 2022] can be leveraged to obtain an anytime competitive ratio guarantee for the stochastic IMAB problem.

The area of fairness in ML has received tremendous attention in recent years [Barocas *et al.*, 2019]. However, much of this attention has been focused on fairness in static, and one-shot settings such as classification [Kleinberg *et al.*, 2017; Hardt *et al.*, 2016; Dwork *et al.*, 2012]. Recent work has studied the fairness aspects of sequential decision-making models, such as MABs [Joseph *et al.*, 2016; Patil *et al.*, 2021; Li *et al.*, 2019; Hossain *et al.*, 2021] and Markov Decision Processes (MDPs) [Jabbari *et al.*, 2017; Wang *et al.*, 2021; Ghalme *et al.*, 2022]. However, these works do not consider the impact of the decisions on the population on which they operate. With a motivation similar to ours, [Lindner *et al.*, 2021] recently studied a problem called the single-peaked bandit model, where the reward functions are monotonically increasing before and decreasing after the peak. This class of reward functions subsumes the class of reward functions considered in [Heidari *et al.*, 2016]. The algorithm and the results in [Lindner *et al.*, 2021], which are again for the horizon-aware case only, match the results in [Heidari *et al.*, 2016] for the class of reward functions considered in IMAB.

2 Model and Preliminaries

Let $\mathbb{R}$ and $\mathbb{N}$ denote the set of real and natural numbers, respectively, and $[k]$ denote the set $\{1, 2, \ldots, k\}$ for $k \in \mathbb{N}$.

Model and Problem Definition. The IMAB model was introduced in [Heidari *et al.*, 2016]. Formally, an instance I of IMAB is defined by a tuple $\langle k, (f_i)_{i\in[k]}\rangle$ where k is the number of arms. Each arm $i \in [k]$ is associated with a *fixed* underlying reward function denoted by $f_i(\cdot)$. When the decision-maker pulls arm i for the n-th time, it obtains an instantaneous reward $f_i(n)$. Further, $\texttt{Rew}_i(N)$ denotes the cumulative reward obtained from arm i after it has been pulled N times, i.e., $\texttt{Rew}_i(N) = f_i(1) + f_i(2) + \ldots + f_i(N)$. We assume that the reward functions f_i, $i \in [k]$ are bounded in $[0, 1]$[3], i.e., $f_i : \mathbb{N} \to [0, 1]$. In our motivating examples, the reward functions f_i correspond to the ability of communities to utilize an opportunity. In the IMAB model, f_i's are assumed to be monotonically increasing with decreasing marginal returns (aka diminishing returns).[4] This assumption about the progression of human abilities is well-supported by literature in cognitive sciences [Son and Sethi, 2006] and microeconomics [Jovanovic and Nyarko, 1995]. The decreasing marginal returns property states that for all $i \in [k]$

$$f_i(n+1) - f_i(n) \le f_i(n) - f_i(n-1) \quad \text{for all } n \ge 1.$$

It is to be noted that the main technical hurdle for an anytime algorithm is the lack of knowledge of the time horizon, which is in no way alleviated by the non-stochastic nature of the reward functions. Next, let a_i denote the asymptote of f_i, i.e., $a_i = \lim_{n\to\infty} f_i(n)$. Since f_i's are monotonically increasing and bounded, this asymptote exists and is finite. We call a_i the true potential of the arm. It represents the ability of communities if they are given enough opportunities.

Let ALG be a deterministic algorithm for the IMAB problem and T be the time horizon (unknown to ALG). Let $i_t \in [k]$ denote the arm pulled by ALG at time step $t \in [T]$. We use $N_i(t)$ to denote the number of pulls of arm i made by ALG until (not including) time step t, and $\texttt{ALG}(I, T)$ to denote the cumulative reward of ALG on instance I at the end of T time steps. Then, $\texttt{ALG}(I, T) = \sum_{t=1}^{T} f_{i_t}(N_{i_t}(t) + 1)$. We note that the cumulative reward of ALG after T time steps only depends on the number of pulls of each arm and not on the order of pulls. Hence, we can write $\texttt{ALG}(I, T) = \sum_{i\in[k]} \texttt{Rew}_i(N_i(T + 1))$. For brevity, we write $N_i(T + 1)$ as N_i. Hence, $\texttt{ALG}(I, T) = \sum_{i\in[k]} \texttt{Rew}_i(N_i)$. When I is clear from context, we use $\texttt{ALG}(T)$ to denote $\texttt{ALG}(I, T)$.

Offline Optimal Algorithm for IMAB. Let $I = \langle k, (f_i)_{i\in[k]}\rangle$ be an IMAB instance. We use $\texttt{OPT}(I, T)$ to denote the offline algorithm maximizing the cumulative reward for instance I and horizon T. Here, offline means that $\texttt{OPT}(I, T)$ knows the IMAB instance I and the time horizon T beforehand. With slight abuse of notation, we also denote the cumulative reward of this algorithm by $\texttt{OPT}(I, T)$. When I is clear from context, we use $\texttt{OPT}(T)$ instead of $\texttt{OPT}(I, T)$. The following proposition shows that for the IMAB problem, $\texttt{OPT}(I, T)$ corresponds to pulling a single arm for all the T rounds. We give an alternate proof in Appendix B.

Proposition 1. *[[Heidari* et al.*, 2016]] Suppose $I = \langle k, (f_i)_{i\in[k]}\rangle$ is an instance of the IMAB problem and T is the*

[2] Asymptotically sub-linear regret: as the horizon tends to infinity, the ratio of the algorithm's regret to the horizon is zero.

[3] We only need that the reward functions are bounded in some interval $[0, c]$, $c \in \mathbb{R}_+$ but we work with $[0, 1]$ following prior work.

[4] If think of f_i's as being continuous functions, in which case monotonically increasing and diminishing returns imply concavity.

time horizon. Then there exists an arm j_T^* such that the optimal offline algorithm consists of pulling arm j_T^* for T time steps.

We emphasize that j_T^* defined above depends on the time horizon T, and may differ for different values of T. We compare the performance of an online algorithm at any time T with $\texttt{OPT}(T)$ using the performance metrics defined next.

Performance Metrics. In this work, our objective is to minimize the stronger notion of regret, viz. policy regret, as opposed to external regret [Arora *et al.*, 2012]. We refer the reader to Example 1 in [Heidari *et al.*, 2016] or Appendix A.2 for insight into how the two regret notions differ in IMAB. We next define policy regret for the IMAB problem. Henceforth, we use regret to denote policy regret unless stated otherwise.

Definition 1. *Let $\mathcal{I}$ denote the set of all problem instances for the IMAB problem with k arms. The policy regret of an algorithm $\texttt{ALG}$ for time horizon T, is defined as*

$$\texttt{Regret}_{\texttt{ALG}}(T) = \sup_{I \in \mathcal{I}} \left[\texttt{OPT}(I,T) - \mathbb{E}[\texttt{ALG}(I,T)]\right] \quad (1)$$

where the expectation is over any randomness in $\texttt{ALG}$.

In Section 3, we show that any algorithm for the IMAB problem must suffer regret that is linear in T. This motivates our choice to study the competitive ratio of an algorithm with respect to the offline optimal algorithm. We note that competitive ratio is a well-studied notion used to evaluate performance of online algorithms [Borodin and El-Yaniv, 2005; Buchbinder *et al.*, 2012] and is also studied in the MAB literature [Immorlica *et al.*, 2019; Kesselheim and Singla, 2020; Andrew *et al.*, 2013; Daniely and Mansour, 2019].

Definition 2. *Let $\mathcal{I}$ denote the set of all problem instances for the IMAB problem with k arms. and $\texttt{ALG}$ be an algorithm for the IMAB problem. Then the (strict) competitive ratio of $\texttt{ALG}$ for time horizon T is defined as*

$$\texttt{CR}_{\texttt{ALG}}(T) = \inf_{\alpha \geq 1} \{\forall I \in \mathcal{I},\ \ \alpha \cdot \texttt{ALG}(I,T) \geq \texttt{OPT}(I,T)\} \quad (2)$$

We will henceforth refer to this as the competitive ratio of an algorithm. To lower bound the competitive ratio of an algorithm $\texttt{ALG}$ by α, it is sufficient to provide an instance I such that $\frac{\texttt{OPT}(I,T)}{\texttt{ALG}(I,T)} \geq \alpha$. Naturally, the goal of the decision-maker is to design an algorithm with a small competitive ratio. Finally, we note that although we have defined $\texttt{Rew}_i(N)$ and $\texttt{ALG}(T)$ as a discrete sum, in some of our proofs, we use definite integrals (area under the curves defined by f_i) to approximate the value of the discrete sum. This approximation does not affect our results (see Appendix A.1).

3 Lower Bound and Round Robin

In this section, we begin by proving the hardness of IMAB. In particular, we show that for any finite time horizon T, there is an instance such that any algorithm for IMAB suffers a regret that is linear in T and has a competitive ratio $\Omega(k)$. This implies that even an algorithm that solely wants to maximize its cumulative reward, without any fairness consideration towards the arms, must suffer linear regret. We also show that the competitive ratio of simple round-robin ($\texttt{RR}$) is $\Theta(k^2)$, and hence $\texttt{RR}$, even though it equally distributes pulls to the arms, is sub-optimal for the decision-maker.

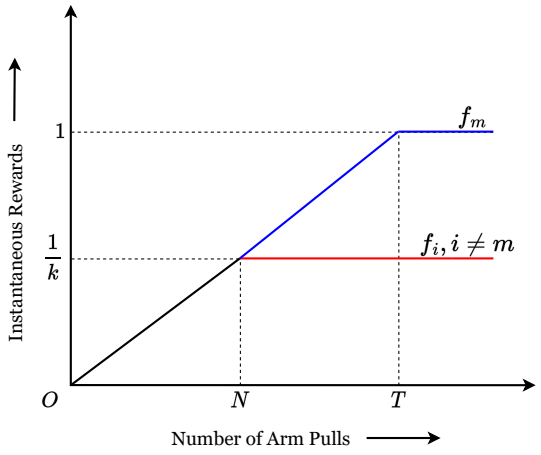

Figure 2: Instance I_m for Lower Bound

Lower Bound: In Theorem 2, we show that for any algorithm $\texttt{ALG}$ and any time horizon T, there exists a problem instance $I = \langle k, (f_i)_{i \in [k]} \rangle$, such that the competitive ratio $\texttt{CR}_{\texttt{ALG}}(T) \geq k/2$. The lower bound shows that we cannot give anytime guarantees with competitive ratio $o(k)$ for IMAB.[5]

Theorem 2. *Let $\texttt{ALG}$ be an algorithm for the IMAB problem with k arms. Then, for any finite time horizon T, there exists a problem instance defined by the reward functions $(f_1, f_2, \ldots, f_k)$ such that*

(a) $\texttt{Regret}_{\texttt{ALG}}(T) \geq c \cdot T$, *for some constant c,*

(b) $\texttt{CR}_{\texttt{ALG}}(T) \geq k/2$.

The proof of our lower bound also shows that even when T is known to the algorithm, an instance-independent sublinear regret is not possible. Note that [Heidari *et al.*, 2016] give an instance-dependent asymptotically sublinear regret. In contrast, we seek an instance-independent anytime regret guarantee (i.e., one that does not depend on the parameters of problem instance). We prove this lower bound by constructing a family of k IMAB instances and showing that no algorithm can achieve sub-linear regret and $o(k)$ competitive ratio on all k instances. We briefly provide intuition about the construction of these problem instances and defer the detailed proof to Appendix C.1. Let $N = \lceil T/k \rceil$. The k problem instances are as follows: For $m \in [k]$, instance I_m is such that

$$\forall i \neq m, f_i(n) = \begin{cases} \frac{n}{kN} & \text{If } n \leq N \\ \frac{1}{k} & \text{If } n > N \end{cases}$$

$$f_m(n) = \begin{cases} \frac{n}{kN} & \text{If } n \leq kN \\ 1 & \text{If } n > kN \end{cases}$$

See Figure 2 for a depiction of instance I_m. An algorithm cannot differentiate between the arms in instance I_m until the

[5] In Appendix C.2, we show that the regret lower bound holds even if the algorithm knows the functions f_i and only T is unknown.

optimal arm m has been pulled at least N times. Given the construction of the reward functions, at least one arm, say arm j, would be pulled $< N$ times by any algorithm. Hence, the algorithm will suffer linear regret on instance I_j.

Round Robin: The RR algorithm pulls arms 1 to k in a cyclic manner, and at the end of T rounds, for any $T \in \mathbb{N}$ ensures that each arm is pulled at least $\lfloor T/k \rfloor$ times irrespective of the reward obtained from the arm. Although this ensures equal distribution of arm pulls, in Theorem 3, we show that RR is sub-optimal in terms of its competitive ratio.

Theorem 3. *Let* RR *denote the round-robin algorithm. Then,* $8k^2 \geq \mathtt{CR}_{\mathtt{RR}}(T) \geq \frac{k^2}{2}$.

The proof is in Appendix C.3. The first inequality above says that the competitive ratio of RR is at most $8k^2$, whereas the second inequality shows that our analysis for RR is tight (up to constants). Theorems 2 and 3 together show a gap of factor k. In the next section, we propose an algorithm whose competitive ratio is optimal up to constants and which allocates the pulls to arms in a manner that ensures each arm attains its true potential given sufficient time.

Algorithm 1: Horizon-Unaware Improving Bandits ALG

1 **Initialize:**
2 $N_i(0) = 0$ for all arms $i \in [k]$ Number of arm pulls
3 Pull each arm twice, $N_i(t) = 2$ for all $i \in [k]$
4 $t = 2k + 1$ Current time step
5 **for** $t = 2k+1, \ldots, T$ **do**
6 $\quad \Delta_i(N_i(t)) = f_i(N_i(t)) - f_i(N_i(t)-1)$
7 $\quad i_t^* \in \arg\max_{i \in [k]} N_i(t)$
8 $\quad$ **for** $i = 1, 2, \ldots, k$ **do**
9 $\quad\quad p_i(t) = \mathtt{Rew}_i(N_i(t)) + \sum_{n=1}^{N_{i^*}(t) - N_i(t)} [f_i(N_i(t)) + n \cdot \Delta_i(N_i(t))]$
10 $\quad$ **end**
11 $\quad C = \arg\max_{i \in [k]} p_i(t)$
12 $\quad$ Pull arm $i_t = \arg\min_{i \in C} N_i(t)$
13 $\quad$ **for** $i = 1, 2, \ldots, k$ **do**
14 $\quad\quad N_i(t+1) = N_i(t) + \mathbb{1}\{i_t = i\}$
15 $\quad$ **end**
16 **end**

4 Optimal Algorithm for Improving Bandits

In this section, we propose an algorithm that *mitigates disparity* due to lack of opportunities while achieving the *best possible cumulative reward* at any time. We first give an intuitive idea about how our algorithm works. Then, we formally state the two-way guarantee that our algorithm provides; first, the tight anytime guarantee for the cumulative reward, and second, the mitigation of disparity by helping arms reach their true potential given sufficient time. In Section 4.1, we state our algorithm along with the main results (Theorems 4 and 5). In Section 4.2, we provide a proof sketch of Theorem 4 along with the supporting lemmas. Finally, in Section 4.3, we give a proof sketch of Theorem 5.

4.1 Algorithm and its Guarantees

Our proposed algorithm is Algorithm 1. Throughout this section, we use ALG to denote Algorithm 1 unless stated otherwise. In addition to having strong performance guarantees, ALG is simple (in terms of the operations used) and efficient (in terms of time complexity). Before stating the theoretical guarantees of our algorithm, we provide an intuitive explanation of how it works. For each arm $i \in [k]$, recall $N_i(t)$ denotes the number of times arm i has been pulled until (not including) time step t, and for notational convenience, we use N_i to denote the number of times ALG pulls arm i in T time steps. Initialization is done by pulling each arm twice (Step 3). This lets us compute the rate of change of the reward function between the first and second pulls of each arm, i.e., $\Delta_i(2) = f_i(2) - f_i(1)$ (as defined in Step 6). This takes $2k$ time steps. At each time step $t > 2k$, let i_t^* denote the arm that has been pulled the maximum number of times so far, i.e., $i_t^* \in \arg\max_{i \in [k]} N_i(t)$. Then, for every arm $i \in [k]$, we compute an optimistic estimate of its cumulative reward had it been pulled $N_{i_t^*}(t)$ times, denoted by $p_i(t)$. The optimistic estimate $p_i(t)$ is computed by adding the actual cumulative reward obtained from arm i in $N_i(t)$ pulls, denoted $\mathtt{Rew}_i(N_i(t))$, and the maximum cumulative reward that can be obtained from the arm in additional $N_{i_t^*}(t) - N_i(t)$ pulls, i.e., if it continues to increase at the current rate, $\Delta_i(N_i(t))$.We then pull an arm with the largest value of $p_i(t)$. Ties are first broken based on the minimum value of $N_i(t)$ and then arbitrarily.

Our goal is to provide an upper bound on $\mathtt{CR}_{\mathtt{ALG}}(T)$. The following theorem, one of our key technical contributions, proves that Algorithm 1 has $O(k)$ competitive ratio. From Theorem 2 in Section 3 it follows that the competitive ratio of our algorithm is optimal (up to constants).

Theorem 4. *The competitive ratio of* ALG *is* $O(k)$. *Further, the time complexity of* ALG *is* $O(k \log k)$ *per time step.*

The per-round time complexity of ALG follows from the $\arg\max$ operation (step 11) performed at each time step, which is standard in MAB literature. This shows that our algorithm, in addition to being simple, is also efficient. The proof of the above theorem relies on some elegant attributes of our algorithm and the class of reward functions. We discuss some of these in Section 4.2. We next show in Theorem 5 that ALG ensures that each arm reaches its true potential given sufficient time.

Theorem 5. *For an arm* $i \in [k]$, *let* $a_i = \lim_{N \to \infty} f_i(N)$. *Then, for every* $\varepsilon \in (0, a_i]$, *there exists* $T \in \mathbb{N}$ *such that* ALG *ensures that* $a_i - f_i(N_i(T)) \leq \varepsilon$.

Theorem 5 shows that all arms reach arbitrarily close to their true potential given sufficient time. In particular, this shows that our algorithm mitigates the initial disparities in the arms due to a lack of opportunities by enabling the arms to reach their true potential given sufficient time. We give a proof sketch in Section 4.3. We recognize that Theorem 5 is not very interesting in and of itself. In particular, even RR satisfies this property. However, what *is* interesting is that our algorithm satisfies this while Theorem 4 also holds. This further underlines our observation that the arm-pull decisions of our algorithm are delicately balanced at each time step.

4.2 Proof Sketch of Theorem 4

The proof of the theorem hinges upon Lemmas 6, 7, and 9 and Corollary 8 stated below. We elaborate upon these lemmas along with the proof sketches for a few of them and then explain how the proof is completed using these lemmas. The complete proof of Theorem 4 along with the proofs of the lemmas and corollaries is in Appendix D.

Let $I = \langle k, (f_i)_{i\in[k]} \rangle$ be an arbitrary instance of the IMAB problem and let T be the time horizon. To upper bound the competitive ratio of our algorithm at T, it is sufficient to upper bound $\frac{\texttt{OPT}(I,T)}{\texttt{ALG}(I,T)}$ (since instance I has been chosen arbitrarily). Throughout, and without loss of generality, assume $N_1 \geq N_2 \geq \ldots \geq N_k$. We begin with a crucial lemma which captures an important feature of our algorithm: the first arm to cross N pulls has to be the optimal arm for the time horizon N, that is, as per Proposition 1, it has to be the arm that maximizes the cumulative reward for horizon N. This property is of key importance in proving the optimality of our algorithm.

Lemma 6. *If arm $i \in [k]$ is the first arm to cross N pulls, i.e., to be pulled $N+1$-th time, then*

$$\texttt{Rew}_i(N) = \texttt{OPT}(I, N).$$

We note that if our algorithm runs for T time steps then the above lemma holds for any N between 1 and T. The proof of the above lemma relies on how we compute $p_i(t)$, the optimistic estimate of the cumulative reward of arm i, at each time step. We remark here that previous works that study the IMAB problem assume that the horizon T is known to the algorithm beforehand. This significantly simplifies the problem of estimating the optimistic estimate of the cumulative reward of any arm using a linear extrapolation. This also allows certain arms to be eliminated based on these estimates. However, such an approach is not possible in the anytime setting. In fact, in Theorem 5, we show that our algorithm keeps pulling an arm until it is improving.

Next, we state Lemma 7 that lower bounds the ratio $\texttt{Rew}_i(N)/\texttt{Rew}_i(T)$, which is the ratio of the cumulative reward of pulling arm i for N pulls to that of pulling it for T pulls, for each arm $i \in [k]$. Part (a) of the following lemma considers the case when $N > T/2$ and part (b) looks at the case when $N \leq T/2$.

Lemma 7. *For each arm $i \in [k]$,*

(a) $\dfrac{\texttt{Rew}_i(\alpha T)}{\texttt{Rew}_i(T)} \geq \dfrac{1}{5} \qquad \text{for } \alpha \geq \dfrac{1}{2}\,,$

(b) $\dfrac{\texttt{Rew}_i(\alpha T/k)}{\texttt{Rew}_i(T)} \geq \dfrac{16\alpha^2}{25k^2} \quad \text{for } 0 \leq \alpha \leq \dfrac{k}{2}.$

The proof of the above lemma crucially relies on the properties of f_i, in particular, the properties that f_i's are monotonically increasing, bounded in $[0,1]$, and have decreasing marginal returns. To prove part (a), we first show that $\texttt{Rew}_i(\alpha T)$ can be lower bounded by the area of triangle defined by O, E, and B in Figure 3 which is equal to $\frac{\alpha T f_i(\alpha T)}{2}$. Further, for $\alpha \geq 1/2$ we show that $\texttt{Rew}_i(T) \leq \frac{5T}{4} f_i(\alpha T)$ (see Claim 4 in Appendix D). This gives us part (a) of the lemma. To prove part (b), we show that $\texttt{Rew}_i(\alpha T/k)$ is lower bounded by the area of the triangle defined by O, E, and B in Figure 4 which is equal to $\frac{\alpha^2 T^2 m_{OE}}{2k^2}$, where m_{OE} is the slope of the line segment passing through points O and E in Figure 4. Using arguments leveraging certain geometric properties satisfied by f_i, we show that $\texttt{Rew}_i(T) \leq \frac{25T^2 m_{OE}}{32}$ (see Claim 5 in Appendix D). This gives us part (b) of the lemma.

Next, we have the following interesting corollary to Lemma 7 which compares the optimal rewards at T and N where N spans values in $\{1, \ldots, T\}$ depending on the value of α.

Corollary 8. *For any finite time horizon T, we have*

(a) $\dfrac{\texttt{OPT}(I, \alpha T)}{\texttt{OPT}(I, T)} \geq \dfrac{1}{5} \text{ for } \dfrac{1}{2} \leq \alpha \leq 1\,,$

(b) $\dfrac{\texttt{OPT}(I, \alpha T/k)}{\texttt{OPT}(I, T)} \geq \dfrac{16\alpha^2}{25k^2} \text{ for } 0 < \alpha \leq \dfrac{k}{2}.$

The proof of the above corollary uses Proposition 1 and Lemma 7. From Proposition 1, we know that the optimal policy for time horizon T pulls a single arm. Let $j_T^* \in [k]$ denote this arm. Further, note that $\texttt{OPT}(I, \alpha T) \geq \texttt{Rew}_{j_T^*}(\alpha T)$, by definition of $\texttt{OPT}(I, \alpha T)$. Since, $\texttt{OPT}(I, T) = \texttt{Rew}_{j_T^*}(T)$, part (a) of the corollary follows from part (a) of Lemma 7. Part (b) is also proved using a similar argument.

Now, observe that $\texttt{ALG}(I, T)$, i.e., the cumulative reward of our algorithm after T time steps can be written as the sum of the rewards obtained from each arm. In particular, $\texttt{ALG}(I, T) = \sum_{i\in[k]} \texttt{Rew}_i(N_i)$, where N_i is the number of times arm i has been pulled in T time steps. Recall that, our goal is to provide an upper bound on $\frac{\texttt{OPT}(I,T)}{\texttt{ALG}(I,T)}$, or equivalently, $\frac{\texttt{OPT}(I,T)}{\sum_{i\in[k]} \texttt{Rew}_i(N_i)}$. The following lemma provides an upper bound on $\frac{\texttt{OPT}(I,T)}{\texttt{Rew}_i(N_i)}$ in terms of only N_i and the time horizon T, when $N_1 \leq T/2$. We handle the (easier) case of $N_1 > T/2$ separately (see proof of Thm. 4 in Appendix D).

Lemma 9. *If $N_1 \leq T/2$ then for any arm $i \in [k]$,*

$$\frac{\texttt{OPT}(I, T)}{\texttt{Rew}_i(N_i)} \leq \frac{200T^2}{N_i^2}.$$

The proof of the above lemma requires intricate case analysis using different properties of our algorithm and the reward functions, and crucially uses Lemmas 7 and 9, and Corollary 8. Finally, with all the components in place, we provide a brief proof sketch of Theorem 4.

Proof Sketch of Theorem 4. Consider the following two cases: 1) $N_1 > T/2$, and 2) $N_1 \leq T/2$.

Case 1 implies that arm 1 is the first, and hence the only arm to cross $T/2$ pulls. From Lemma 6, we get, $\texttt{Rew}_1(T/2) = \texttt{OPT}(I, T/2)$. Therefore, $\texttt{ALG}(I, T) \geq \texttt{Rew}_1(T/2) = \texttt{OPT}(I, T/2)$. Hence, we obtain, $\texttt{OPT}(I, T)/\texttt{ALG}(I, T) \leq \texttt{OPT}(I, T)/\texttt{OPT}(I, T/2) \leq 5 \leq 200k$. Here, the second inequality follows from Corollary 8. Since the above inequality holds for any instance I, we get $\texttt{CR}_{\texttt{ALG}}(T) \leq 200k$.[6]

[6]For ease of exposition, we have not optimized the constants.

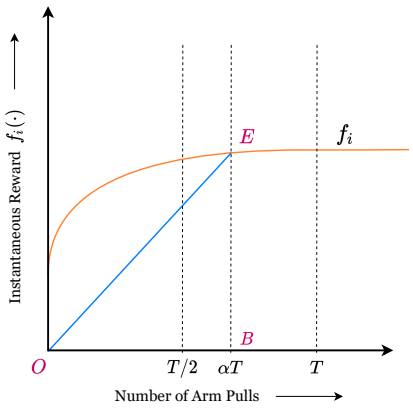

Figure 3: $\alpha > 1/2$

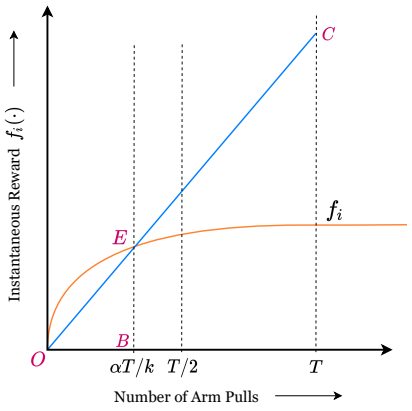

Figure 4: $0 < \alpha \leq k/2$

For Case 2, we use Lemma 9 and obtain

$$\frac{\sum_{i\in[k]} \texttt{Rew}_i(N_i)}{\texttt{OPT}(I,T)} \geq \frac{\sum_{i\in[k]} N_i^2}{200T^2} \geq \frac{(\sum_{i\in[k]} N_i/\sqrt{k})^2}{200T^2} = \frac{T^2}{200kT^2} = \frac{1}{200k}.$$

Here, the second inequality follows from Cauchy-Schwarz inequality (see Observation 3 in Appendix D). This implies $\frac{\texttt{OPT}(I,T)}{\texttt{ALG}(I,T)} \leq 200k$. Since this holds for an arbitrary instance $I \in \mathcal{I}$, we have $\texttt{CR}_{\texttt{ALG}}(T) \leq 200k$, i.e., $O(k)$. □

4.3 Proof Sketch for Theorem 5

The theorem is proved using Lemma 10 stated below. See Appendix E for the detailed proofs of Lemma 10 and Theorem 5). In Lemma 10, we show that ALG pulls an arm finitely many times only if the arm stops improving.

Lemma 10. *Let $L_i = \max_{t\in\mathbb{N}}\{N_i(t)\}$ for all $i \in [k]$. Then for any $i \in [k]$, L_i is finite implies that $\Delta_i(L_i) = 0$.*

L_i as defined above captures the number of times the algorithm pulls arm i as $T \to \infty$. It is easy to see that there is at least one arm $i \in [k]$ such that L_i is not finite, and hence, the property holds for this arm vacuously. The proof of the lemma argues via contradiction that the property has to be satisfied for all the arms. Suppose there is an arm $j \in [k]$ such that L_j is finite but $\Delta_j(L_j) \neq 0$. Then we consider a time horizon larger than when arm j was pulled for the L_j-th time and use the definition of $p_j(t)$ to show that such an arm is indeed pulled again, contradicting the assumption.

Using this lemma, the theorem is proved as follows. Suppose L_i is finite for an arm $i \in [k]$. Then the diminishing returns property of f_i ensures that arm i has reached its true potential, i.e, $f_i(L_i) = a_i$. Further, if L_i is not finite then the arm is pulled infinitely many times, and hence from the properties of f_i we have that for every $\varepsilon \in (0, a_i]$, there exists $T \in \mathbb{N}$ such that ALG ensures that $a_i - f_i(N_i(T)) \leq \varepsilon$.

5 Experiments

In this paper, we focused on a self-contained theoretical study of the IMAB problem. Our algorithm provides $O(k)$ competitive ratio, whereas HKR can only achieve $\Theta(k^2)$ competitive ratio (see Appendix G). In fact, in Appendix G we show that the ratio in the reward obtained by of our algorithm and that of HKR is $\Omega(k^2)$ on some instances. Thus, our analysis of the worst-case performance guarantee is quite revealing in and of itself. Given this and the space constraint, we have provided some experimental analysis in Appendix F that further underlines our results in the main body of the paper. We compared the performance of the two algorithms on the instances in [Heidari *et al.*, 2016] (see Figure 10). We observe that for finite time horizons, our algorithm can match or even beat the performance of horizon-aware HKR on many instances, despite not knowing T. We also compared the performance of our algorithm, HKR, and Round Robin on some randomly generated IMAB instances (see Figure 11). In Figure 12, we also show some additional experiments that provide a visual depiction of Theorem 5 and compare it with the HKR algorithm. We refer the reader to Appendix F for details regarding the experiments and the figures referred to above.

6 Conclusion and Future Work

We studied the IMAB problem in the horizon-unaware setting. A direction that is of immediate future interest is to study the Single-Peaked MAB model when the horizon is not known. This model, where the reward function is monotonically increasing before and decreasing after the peak has been studied when the time horizon is known [Lindner *et al.*, 2021]. It would be interesting to see if our ideas can be extended to the Single-Peaked model to obtain anytime guarantees. Another question of immediate interest is to define alternate regret notions for IMAB that are natural and provide better regret bounds. As mentioned in Section 1.1, it would also be interesting to see if our results extend to the stochastic setting where the expected rewards of the arms change as a function of pulls.

Acknowledgements

Vishakha is grateful for the support of a Google PhD Fellowship. Ganesh's research is supported by SERB research grant CRG/2022/007927. Arindam's research is supported

by IUSSTF virtual center on "Polynomials as an Algorithmic Paradigm', Pratiksha Trust Young Investigator Award, Google India Research Award, Google ExploreCS Award, and SERB Core Research Grant (CRG/2022/001176) on "Optimization under Intractability and Uncertainty".

Contribution Statement

Vishakha Patil and Vineet Nair have contributed equally to this work and are the joint first authors. Their names are ordered randomly in the author list.

References

[Andrew *et al.*, 2013] Lachlan Andrew, Siddharth Barman, Katrina Ligett, Minghong Lin, Adam Meyerson, Alan Roytman, and Adam Wierman. A tale of two metrics: Simultaneous bounds on competitiveness and regret. In *Conference on Learning Theory*, pages 741–763. PMLR, 2013.

[Arora *et al.*, 2012] Raman Arora, Ofer Dekel, and Ambuj Tewari. Online bandit learning against an adaptive adversary: from regret to policy regret. *arXiv preprint arXiv:1206.6400*, 2012.

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

[Auer *et al.*, 2002b] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

[Barocas *et al.*, 2019] Solon Barocas, Moritz Hardt, and Arvind Narayanan. Fairness and machine learning. http://www.fairmlbook.org, 2019. Accessed: 2023-05-29.

[Besbes *et al.*, 2014] Omar Besbes, Yonatan Gur, and Assaf Zeevi. Stochastic multi-armed-bandit problem with non-stationary rewards. *Advances in neural information processing systems*, 27, 2014.

[Borodin and El-Yaniv, 2005] Allan Borodin and Ran El-Yaniv. *Online computation and competitive analysis*. cambridge university press, 2005.

[Buchbinder *et al.*, 2012] Niv Buchbinder, Shahar Chen, Joshep Seffi Naor, and Ohad Shamir. Unified algorithms for online learning and competitive analysis. In *Conference on Learning Theory*, pages 5–1. JMLR Workshop and Conference Proceedings, 2012.

[Daniely and Mansour, 2019] Amit Daniely and Yishay Mansour. Competitive ratio vs regret minimization: achieving the best of both worlds. In *Algorithmic Learning Theory*, pages 333–368. PMLR, 2019.

[Dwork *et al.*, 2012] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.

[Garivier and Moulines, 2011] Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In *International Conference on Algorithmic Learning Theory*, pages 174–188. Springer, 2011.

[Ghalme *et al.*, 2022] Ganesh Ghalme, Vineet Nair, Vishakha Patil, and Yilun Zhou. Long-term resource allocation fairness in average markov decision process (amdp) environment. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pages 525–533, 2022.

[Hardt *et al.*, 2016] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016.

[Heidari *et al.*, 2016] Hoda Heidari, Michael J Kearns, and Aaron Roth. Tight policy regret bounds for improving and decaying bandits. In *IJCAI*, pages 1562–1570, 2016.

[Hossain *et al.*, 2021] Safwan Hossain, Evi Micha, and Nisarg Shah. Fair algorithms for multi-agent multi-armed bandits. *Advances in Neural Information Processing Systems*, 34, 2021.

[Immorlica *et al.*, 2019] Nicole Immorlica, Karthik Abinav Sankararaman, Robert Schapire, and Aleksandrs Slivkins. Adversarial bandits with knapsacks. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 202–219. IEEE, 2019.

[J. Baker, 2019] Dominique J. Baker. Why might states ban affirmative action? https://www.brookings.edu/blog/brown-center-chalkboard/2019/04/12/why-might-states-ban-affirmative-action/, 2019. Accessed: 2023-05-29.

[Jabbari *et al.*, 2017] Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, and Aaron Roth. Fairness in reinforcement learning. In *International conference on machine learning*, pages 1617–1626. PMLR, 2017.

[Joseph *et al.*, 2016] Matthew Joseph, Michael Kearns, Jamie Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. *arXiv preprint arXiv:1605.07139*, 2016.

[Jovanovic and Nyarko, 1995] Boyan Jovanovic and Yaw Nyarko. A bayesian learning model fitted to a variety of empirical learning curves. *Brookings Papers on Economic Activity. Microeconomics*, 1995:247–305, 1995.

[Kesselheim and Singla, 2020] Thomas Kesselheim and Sahil Singla. Online learning with vector costs and bandits with knapsacks. In *Conference on Learning Theory*, pages 2286–2305. PMLR, 2020.

[Kleinberg *et al.*, 2017] Jon M. Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. In Christos H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference, ITCS 2017, January 9-11, 2017, Berkeley, CA, USA*, volume 67 of *LIPIcs*, pages 43:1–43:23. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

[Levine *et al.*, 2017] Nir Levine, Koby Crammer, and Shie Mannor. Rotting bandits. *Advances in neural information processing systems*, 30, 2017.

[Li *et al.*, 2019] Fengjiao Li, Jia Liu, and Bo Ji. Combinatorial sleeping bandits with fairness constraints. *IEEE Transactions on Network Science and Engineering*, 7(3):1799–1813, 2019.

[Lindner *et al.*, 2021] David Lindner, Hoda Heidari, and Andreas Krause. Addressing the long-term impact of ml decisions via policy regret. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 537–544. International Joint Conferences on Artificial Intelligence, 2021.

[Metelli *et al.*, 2022] Alberto Maria Metelli, Francesco Trovo, Matteo Pirola, and Marcello Restelli. Stochastic rising bandits. In *International Conference on Machine Learning*, pages 15421–15457. PMLR, 2022.

[Patil *et al.*, 2021] Vishakha Patil, Ganesh Ghalme, Vineet Nair, and Y. Narahari. Achieving fairness in the stochastic multi-armed bandit problem. *J. Mach. Learn. Res.*, 22:174:1–174:31, 2021.

[Robbins, 1952] Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

[Sahoo, 2009] Niranjan Sahoo. Reservation policy and its implementation across domains in india. *Academic Foundation in association with Observer Research Foundation*, 2009.

[Son and Sethi, 2006] Lisa K Son and Rajiv Sethi. Metacognitive control and optimal learning. *Cognitive Science*, 30(4):759–774, 2006.

[Staff, 2018] GlobalIsAsian Staff. Meritocracy in singapore: Solution or problem? https://lkyspp.nus.edu.sg/gia/article/meritocracy-in-singapore-solution-or-problem, 2018. Accessed: 2023-05-29.

[Tekin and Liu, 2012] Cem Tekin and Mingyan Liu. Online learning of rested and restless bandits. *IEEE Transactions on Information Theory*, 58(8):5588–5611, 2012.

[V. Reeves and Halikias, 2017] Richard V. Reeves and Dimitrios Halikias. Race gaps in sat scores highlight inequality and hinder upward mobility. https://www.brookings.edu/research/race-gaps-in-sat-scores-highlight-inequality-and-hinder-upward-mobility/, 2017. Accessed: 2023-05-29.

[Wang *et al.*, 2021] Lequn Wang, Yiwei Bai, Wen Sun, and Thorsten Joachims. Fairness of exposure in stochastic bandits. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10686–10696. PMLR, 18–24 Jul 2021.