My Fair Bandit: Distributed Learning of Max-Min Fairness with Multi-player Bandits

Ilai Bistritz [1] **Tavor Z. Baharav** [1] **Amir Leshem** [2] **Nicholas Bambos** [1]

Abstract

Consider N cooperative but non-communicating players where each plays one out of M arms for T turns. Players have different utilities for each arm, representable as an $N \times M$ matrix. These utilities are unknown to the players. In each turn players select an arm and receive a noisy observation of their utility for it. However, if any other players selected the same arm that turn, all colliding players will receive zero utility due to the conflict. No other communication or coordination between the players is possible. Our goal is to design a distributed algorithm that learns the matching between players and arms that achieves max-min fairness while minimizing the regret. We present an algorithm and prove that it is regret optimal up to a $\log \log T$ factor. This is the first max-min fairness multi-player bandit algorithm with (near) order optimal regret.

1. Introduction

In online learning problems, an agent sequentially makes decisions and receives an associated reward. When this reward is stochastic, the problem takes the form of a stochastic multi-armed bandit problem (Bubeck et al., 2012). However, stochastic bandits assume stationary reward processes that are rarely the case in practice; they are too optimistic about the environment. To deal with this shortcoming, one can model the rewards as being determined by an adversary, which leads to a formulation known as non-stochastic or adversarial bandits (Bubeck et al., 2012). Naturally, the performance guarantees against such a powerful adversary are much weaker than in the stochastic case, as adversarial bandits are usually overly pessimistic about the environment. Is there an alternative that lies in the gap between the two?

Multi-player bandits is a promising answer, which has seen a surge of interest in recent years (Anandkumar et al., 2011; Avner & Mannor, 2014; 2016; Cohen et al., 2017; Evirgen & Kose, 2017; Lai et al., 2008; Liu et al., 2013; Liu & Zhao, 2010; Liu et al., 2019; Magesh & Veeravalli, 2019; Sankararaman et al., 2019; Vakili et al., 2013). One primary motivation for studying multi-player bandits is distributed resource allocation. Examples include channels in communication networks, computation resources on servers, consumers and items, etc. In most applications of interest, the reward of a player is a stochastic function of the decisions of other players that operate in the same environment. Thinking of arms as resources, we see that while these players are not adversaries, conflicts still arise due to the players' preferences among the limited resources. To model that, we assign zero reward to players that choose the same arm. The goal of multi-player bandit algorithms is to provide a distributed way to learn how to share these resources optimally in an online manner. This is useful in applications where agents (players) follow a standard or protocol, like in wireless networks, autonomous vehicles, or with a team of robots.

A common network performance objective is the sum of rewards of the players over time. As such, maximizing the sum of rewards has received the vast majority of the attention in the multi-player bandit literature (Besson & Kaufmann, 2018; Bistritz & Leshem, 2018; Boursier & Perchet, 2019; Boursier et al., 2019; Hanawal & Darak, 2018; Kalathil et al., 2014; Nayyar et al., 2016; Tibrewal et al., 2019). However, in the broader literature of network optimization, the sum of rewards is only one possible objective. One severe drawback of this objective is that it has no fairness guarantees. As such, the maximal sum of rewards assignment might starve some users. In many applications, the designer wants to make sure that all users will enjoy at least minimal target Quality of Service (QoS).

Ensuring fairness has recently been recognized by the machine learning community as a problem of key importance. In addition to interest in fair classifiers (Zemel et al., 2013), fairness has been recognized as a major design parameter

[1] Department of Electrical Engineering, Stanford University [2] Faculty of Engineering, Bar-Ilan University. Correspondence to: Ilai Bistritz <bistritz@stanford.edu>, Tavor Z. Baharav <tavorb@stanford.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

in reinforcement learning and single-player bandits as well (Jabbari et al., 2017; Joseph et al., 2016; Wei et al., 2015). Our work addresses this major concern for the emerging field of multi-player bandits.

In the context of maximizing the sum of rewards, many works on multi-player bandits have considered a model where all players have the same vector of expected rewards (Alatur et al., 2019; Boursier & Perchet, 2019; Bubeck et al., 2019; Rosenski et al., 2016). While being relevant in some applications this model is not rich enough to study fairness, as then the worst off player is simply the one that was allocated the worst resource. To study fairness a heterogeneous model is necessary, where players have different expected rewards for the arms (a matrix of expected rewards). In this case, a fair allocation may prevent some players from getting their best arm in order to significantly improve the allocation for less fortunate players.

Despite being a widely-applied objective in the broader resource allocation literature (Asadpour & Saberi, 2010; Radunovic & Le Boudec, 2007; Zehavi et al., 2013), max-min fairness in multi-player bandits has yet to be studied. Some bandit works have studied alternative objectives that can potentially exhibit some level of fairness (Bar-On & Mansour, 2019; Darak & Hanawal, 2019). In the networking literature, a celebrated notion of fairness is α*-fairness* (Mo & Walrand, 2000) where $\alpha = 1$ yields proportional fairness and $\alpha = 2$ yields sum of utilities. While for constant α, α-fairness can be maximized in a similar manner to (Bistritz & Leshem, 2018), the case of max-min fairness corresponds to $\alpha \to \infty$ and is fundamentally different.

Learning to play the max-min fairness allocation involves major technical challenges that do not arise in the case of maximizing the sum of rewards (or in the case of α- fairness). The sum-rewards optimal allocation is unique for "almost all" scenarios (randomizing the expected rewards). This is not the case with max-min fairness, as there will typically be multiple optimal allocations. This complicates the distributed learning process, since players will have to agree on a specific optimal allocation to play, which is difficult to do without communication. Specifically, this rules out using similar techniques to those used in (Bistritz & Leshem, 2018) to solve the sum of rewards case under a similar multi-player bandit setting (i.e., matrix of expected rewards, no communication, and a collision model).

Trivially, any matching where all players achieve reward greater than γ has max-min value of at least γ. We observe that finding these matchings, called γ-matchings in this paper, can be done via simple dynamics that introduce an absorbing Markov chain with the γ-matchings as the absorbing states. This insight allows for a more robust algorithm than that of (Bistritz & Leshem, 2018) which relies on ergodic Markov chains that have an exploration parameter ε that has to be tuned.

In this work we provide an algorithm that has provably order optimal regret up to a $\log\log T$ factor (that can be arbitrarily improved to any factor that increases with the horizon T). We adopt the challenging model with heterogeneous arms (a matrix of expected rewards) and no communication between the players. The only information players receive regarding their peers is through the collisions that occur when two or more players pick the same arm.

1.1. Outline

In Section 2 we formulate our multi-player bandit problem of learning the max-min matching under the collision model with no communication between players. In Section 3 we present our distributed max-min fairness algorithm and state our regret bound (proved in Section 8). Section 4 analyzes the exploration phase of our algorithm. Section 5 analyzes the matching phase of our algorithm and bounds the probability of the exploitation error event. Section 6 presents simulation results that corroborate our theoretical findings and demonstrate that our algorithm learns the max-min optimal matching faster than our analytical bounds suggest. Finally, Section 7 concludes the paper.

2. Problem Formulation

We consider a stochastic game played by a set of N players $\mathcal{N} = \{1, ..., N\}$ over a finite time horizon T. The strategy space of each player is the set of M arms with indices denoted by $i, j \in \{1, ..., M\}$. We assume that $M \geq N$, since otherwise the max-min utility is trivially zero. The horizon T is not known by any of the players, and is considered to be much larger than M and N since we assume that the game is played for a long time. Let t be the discrete turn index. At each turn t, all players simultaneously pick one arm each. The arm that player n chooses at turn t is $a_n(t)$ and the strategy profile (vector of arms selected) at turn t is $\boldsymbol{a}(t)$. Players do not know which arms the other players chose, and need not even know the number of players N.

Define the no-collision indicator of arm i in strategy profile $\boldsymbol{a}$ to be

$$\eta_i(\boldsymbol{a}) = \begin{cases} 0 & \left|\mathcal{N}_i(\boldsymbol{a})\right| > 1 \\ 1 & otherwise. \end{cases} \tag{1}$$

where $\mathcal{N}_i(\boldsymbol{a}) = \{n \mid a_n = i\}$ is the set of players that chose arm i in strategy profile $\boldsymbol{a}$. The instantaneous utility of player n at time t with strategy profile $\boldsymbol{a}(t)$ is

$$\upsilon_n(\boldsymbol{a}(t)) = r_{n,a_n(t)}(t)\,\eta_{a_n(t)}(\boldsymbol{a}(t)) \tag{2}$$

where $r_{n,a_n(t)}(t)$ is a random reward which is assumed to have a continuous distribution on $[0, 1]$. The sequence of rewards of arm i for player n, $\{r_{n,i}(t)\}_{t=1}^{T}$, is i.i.d. with expectation $\mu_{n,i}$.

An immediate motivation for the collision model above is channel allocation in wireless networks, where the transmission of one user creates interference for other users on the same channel and causes their transmission to fail. Since coordinating a large number of devices in a centralized manner is infeasible, distributed channel allocation algorithms are desirable in practice. In this context, our distributed algorithm learns over time how to assign the channels (arms) such that the maximal QoS guarantee is maintained for all users. However, the collision model is relevant to many other resource allocation scenarios where the resources are discrete items that cannot be shared.

Next we define the total expected regret for this problem. This is the expected regret that the cooperating but non-communicating players accumulate over time from not playing the optimal max-min allocation.

Definition 1. The total expected regret is defined as

$$R(T) = \sum_{t=1}^{T} \left(\gamma^* - \min_n \mathbb{E}\left\{ \upsilon_n \left(\boldsymbol{a}(t) \right) \right\} \right) \tag{3}$$

where $\gamma^* = \max_{\boldsymbol{a}} \min_n \mathbb{E}\left\{ \upsilon_n \left(\boldsymbol{a} \right) \right\}$. The expectation is over the randomness of the rewards $\left\{ r_{n,i}(t) \right\}_t$, that dictate the random actions $\left\{ a_n(t) \right\}_t$.

Note that replacing the minimum in (3) with a sum over the players yields the regret for the sum-reward objective case, after redefining γ^* to be the optimal sum-reward (Bistritz & Leshem, 2018; Boursier & Perchet, 2019; Kalathil et al., 2014; Nayyar et al., 2016; Tibrewal et al., 2019).

Rewards with a continuous distribution are natural in many applications (e.g., SNR in wireless networks). However, this assumption is only used to argue that since the probability for zero reward in a non-collision is zero, players can properly estimate their expected rewards. In the case where the probability of receiving zero reward is not zero, we can assume instead that each player can observe their no-collision indicator in addition to their reward. This alternative assumption requires no modifications to our algorithm or analysis. Observing one bit of feedback signifying whether other players chose the same arm is significantly less demanding than observing the actions of other players.

According to the seminal work in (Lai & Robbins, 1985), the optimal regret of the single-player case is $O(\log T)$. The next proposition shows that $\Omega(\log T)$ is a lower bound for our multi-player bandit case, since any multi-player bandit algorithm can be used as a single-player algorithm by simulating other players.

Proposition 1. *The total expected regret as defined in* (3) *of any algorithm is at least* $\Omega(\log T)$.

Proof. For $N = 1$, the result directly follows from (Lai & Robbins, 1985). Assume that for $N > 1$ there is a policy that results in total expected regret better than $\Omega(\log T)$. Then any single player, denoted player n, can simulate $N-1$ other players such that all their expected rewards are larger than her maximal expected reward. Player n can also generate the other players' random rewards, that are independent of the actual rewards she receives. Player n also simulates the policies for other players, and even knows when a collision occurred for herself and can assign zero reward in that case. In this scenario, the expected reward of player n is the minimal expected reward among the non-colliding players. This implies that γ^* is the largest expected reward of player n. Hence, in every turn t without a collision, the t-th term in (3) is equal to the t-th term of the single-player regret of player n. If there is a collision in turn t, then the t-th term in (3) is γ^*, which bounds from above the t-th term of the single-player regret of player n. Thus, the total expected regret upper bounds the single-player regret of player n. Hence, simulating $N-1$ fictitious players is a valid single player algorithm that violates the $\Omega(\log T)$ bound, which is a contradiction. We conclude that the $\Omega(\log T)$ bound is also valid for $N > 1$. □

3. My Fair Bandit Algorithm

In this section we describe our distributed multi-player bandit algorithm that achieves near order optimal regret for the max-min fairness problem. The key idea behind our algorithm is a global search parameter γ that all players track together (with no communication required). We define a γ-matching, which is a matching of players to arms such that the expected reward of each player is at least γ.

Definition 2. An allocation of arms $\boldsymbol{a}$ is a γ-matching if and only if $\min_n \mathbb{E}\left\{ \upsilon_n \left(\boldsymbol{a} \right) \right\} \geq \gamma$.

Essentially, the players want to find the maximal γ for which there still exists a γ-matching. However, even for a given achievable γ, distributedly converging to a γ-matching is a challenge. Players do not know their expected rewards, and their coordination is extremely limited. To address these issues we divide the unknown horizon of T turns into epochs, one starting immediately after the other. Each epoch is further divided into four phases. In the k-th epoch we have:

1. **Exploration Phase** - this phase has a length of $\lceil c_1 \log(k+1) \rceil$ turns for some $c_1 \geq 4$. It is used for estimating the expectation of the arms. As shown in Section 4, the exploration phase contributes $O(\log \log T \log T)$ to the total expected regret.

2. **Matching Phase** - this phase has a length of $\lceil c_2 \log(k+1) \rceil$ turns for some $c_2 \geq 1$. In this phase, players attempt to converge to a γ_k-matching, where each player plays an arm that is at least as good as γ_k,

up to the confidence intervals of the exploration phase. To find the matching, the players follow distributed dynamics that induce an absorbing Markov chain with the strategy profiles as states. The absorbing states of this chain are the desired matchings. When the matching phase is long enough, the probability that a matching exists but is not found is small. If a matching does not exist, the matching phase naturally does not converge. As shown in Section 5, the matching phase adds $O(\log\log T \log T)$ to the total expected regret.

3. **Consensus Phase** - this phase has a length of M turns. The goal of this phase is to let all players know whether the matching phase ended with a matching, using the collisions for signaling. During this phase, every player who did not end the matching phase with a collision plays their matched arm, while the players that ended the matching phase with a collision sequentially play all the arms for the next M turns. If players deduce that they collectively converged to a matching, they note this for future reference in the matching indicator S_k. If the matching phase succeeded, the search parameter is updated as $\gamma_{k+1} = \gamma_k + \varepsilon_k$. The step size ε_k is decreasing such that if even a slightly better matching exists, it will eventually be found. However, it might be that no γ_k-matching exists. Hence once in a while, with decreasing frequency, the players reset $\gamma_{k+1} = 0$ in order to allow themselves to keep finding new matchings. This phase adds $O(M \log T)$ to the total expected regret.

4. **Exploitation Phase** - this phase has a length of $\left\lceil c_3 \left(\frac{4}{3}\right)^k \right\rceil$ turns for some $c_3 \geq 1$. During this phase, players play the best recently found matching $\tilde{\boldsymbol{a}}_{k^*}$, where k^* is the epoch within the last $\frac{k}{2}$ epochs with the largest γ that resulted in a matching. This phase adds a vanishing term (with T) to the total expected regret since players eventually play an optimal matching with exponentially small error probability.

The Fair Bandit Algorithm is detailed in Algorithm 1. Our main result is given next, and is proved in Section 8.

Theorem 1 (Main Theorem)**.** *Assume that the rewards $\{r_{n,i}(t)\}_t$ are independent in n and i.i.d. with t, with continuous distributions on $[0,1]$ with expectations $\{\mu_{n,i}\}$. Let T be the finite deterministic horizon of the game, which is unknown to the players. Let each player play according to Algorithm 1 with any constants $c_1, c_2, c_3 \geq 4$. Then the total expected regret satisfies*

$$R(T) \leq C_0 + \left(M + 2(c_1 + c_2)\log\log_{\frac{4}{3}}\frac{T}{c_3}\right)\log_{\frac{4}{3}}\frac{T}{c_3}$$
$$= O\left((M + \log\log T)\log T\right) \qquad (4)$$

where C_0 is a constant independent of T and the $\log\log T$ can be improved to any increasing function of T by changing the lengths of the exploration and matching phases.

The purpose of the epoch structure is to address the main challenge arising from having multiple players: coordinating between players without communication. To this end, the players in our algorithm try together to find a γ-matching, where γ is a mutual parameter that they can all update simultaneously but independently, obviating the need for a central entity. Then the main measure of multi-player "problem hardness" is the absorption time $\bar{\tau}$ of the matching Markov chain (see Lemma 4), which is unknown. To find a matching we then need a matching phase with increasing length (to eventually surpass $\bar{\tau}$), taken to be of length $\lceil c_2 \log(k+1) \rceil$ for the k-th phase, which alone contributes $O(\log\log T \log T)$ to the total expected regret. Hence, the coordination challenge dominates the regret.

The additive constant C_0 (see (25)) is essentially the total regret accumulated during the initial epochs when the confidence intervals were still not small enough compared to the gap $\Delta = \min\limits_{n} \min\limits_{i \neq j} |\mu_{n,i} - \mu_{n,j}|$ (formalized in (11)) or when the length of the matching phase was not long enough compared to $\bar{\tau}$ (formalized in (17)). Hence, C_0 depends on Δ and $\bar{\tau}$. In a simplified scenario when Δ and $\bar{\tau}$ are known or can be bounded, the lengths of the exploration and matching phase can be made constant (c_1 and c_2) and the confidence intervals in (10) can be set to $\frac{\Delta}{4}$. Note that with constant length phases, the $\log\frac{T_e(k)}{5M}$ in (13) would be replaced with $\frac{M}{\Delta^2}$ and the $\log(\frac{k}{2}+1)$ in (17) replaced with a constant. Then c_1, c_2 can be chosen such that $C_0 = 0$, by making the probability in (13) vanish faster than $\left(\frac{3}{4}\right)^k$ and satisfying (17). This amounts to choosing $c_1 = O\left(\frac{M}{\Delta^2} + M^2\right)$ and $c_2 = O(\bar{\tau})$, which makes our regret bound in (4) become $O\left(\left(\frac{M}{\Delta^2} + M^2 + \bar{\tau}\right)\log T\right)$, since the $\log\log T$ becomes 1 with constant length phases. Nevertheless, this issue is mainly theoretical since in practice, it is easy to choose large enough c_1, c_2 such that C_0 is very small across various experiments, as can be seen in our simulations in Section 6.

4. Exploration Phase

Over time, players receive stochastic rewards from different arms and average them to estimate their expected reward for each arm. In each epoch, only $\lceil c_1 \log(k+1) \rceil$ turns are dedicated to exploration. However, the estimation of the expected rewards uses all the previous exploration phases, so the number of samples for estimation at epoch k is $\Theta(k \log k)$. Since players only have estimates of the expected rewards, they can never be sure if a matching is a γ-matching. The purpose of the exploration phase is to help the players become more confident over time that the matchings they converge to in the matching phase are indeed γ-matchings.

Algorithm 1 My Fair Bandit Algorithm

```
Initialization: Set $V_{n,i} = 0$ and $s_{n,i} = 0$ for all $i$. Set reset counter $w = 0$ with expiration $e_w = 1$. Let $\varepsilon_0 = 1$.
For each epoch $k = 1, 2, \ldots$
  1. Exploration Phase:
     (a) For the next $\lceil c_1 \log(k+1) \rceil$ turns:
         i. Play an arm $i$ uniformly at random from all $M$ arms.
         ii. Receive $r_{n,i}(t)$ and set $\eta_i(\boldsymbol{a}(t)) = 0$ if $r_{n,i}(t) = 0$ and $\eta_i(\boldsymbol{a}(t)) = 1$ otherwise.
         iii. If $\eta_i(\boldsymbol{a}(t)) = 1$ then update $V_{n,i} = V_{n,i} + 1$ and $s_{n,i} = s_{n,i} + r_{n,i}(t)$.
     (b) Estimate the expectation of arm $i$ as $\mu^k_{n,i} = \frac{s_{n,i}}{V_{n,i}}$ for each $i = 1, ..., M$.
     (c) Construct confidence intervals for each $\mu^k_{n,i}$ as $C^k_{n,i} = \sqrt{\frac{M}{\log V_{n,i}}}$.
  2. Matching Phase:
     (a) Update $w \leftarrow w + 1$. If $w = e_w$ then set $\gamma_k = 0$, $w = 0$, $e_w = \left\lceil \frac{k}{3} \right\rceil$ and update $\varepsilon_k = \frac{1}{1+\log k}$. If $w < e_w$ then $\varepsilon_k = \varepsilon_{k-1}$.
     (b) Let $\mathcal{E}^k_n = \left\{ i \,|\, \mu^k_{n,i} \geq \gamma_k - C^k_{n,i} \right\}$.
     (c) Pick $a_n(t)$ uniformly at random from $\mathcal{E}^k_n$.
     (d) For the next $\lceil c_2 \log(k+1) \rceil$ turns:
         i. If $\eta_{a_n(t)}(\boldsymbol{a}(t)) = 1$ then keep playing the same arm, that is $a_n(t+1) = a_n(t)$.
         ii. If $\eta_{a_n(t)}(\boldsymbol{a}(t)) = 0$ then pick $a_n(t+1)$ uniformly at random from $\mathcal{E}^k_n$.
     (e) Set $\widetilde{a}_{k,n} = a_n(t)$.
  3. Consensus Phase:
     (a) If $\eta_{\widetilde{a}_{k,n}}(\widetilde{\boldsymbol{a}}_k) = 1$ then play $\widetilde{a}_{k,n}$ for $M$ turns.
     (b) If $\eta_{\widetilde{a}_{k,n}}(\widetilde{\boldsymbol{a}}_k) = 0$ then play $a_n = 1, ..., M$ sequentially.
     (c) Matching was found: If you did not experience a collision in the last $M$ turns, set $\gamma_{k+1} = \gamma_k + \varepsilon_k$ and $S_k = 1$, else set $\gamma_{k+1} = \gamma_k$, $S_k = 0$.
  4. Exploitation Phase: For $\left\lceil c_3 \left(\frac{4}{3}\right)^k \right\rceil$ turns, play $\widetilde{a}_{k^*,n}$ for the maximal $k^*$ such that
         $k^* \in \underset{\lceil \frac{k}{2} \rceil \leq \ell \leq k}{\arg\max}\, \gamma_\ell S_\ell.$
End
```

In our exploration phase each player picks an arm uniformly at random. This type of exploration phase is common in various multi-player bandit algorithms (Bistritz & Leshem, 2018; Rosenski et al., 2016). However, the nature of what the players are trying to estimate is different. With a sum of rewards objective, players just need to improve over time the accuracy of the estimation of the expected rewards. With max-min fairness each player needs to make a hard (binary) decision whether a certain arm has expected reward above or below γ. After the confidence intervals become small enough, if the estimations do fall within their confidence intervals, players can be confident about this hard decision. Under this success event, where the confidence intervals are small enough, a matching $\boldsymbol{a}$ is a γ-matching if all players observe that $\mu^k_{n,a_n} \geq \gamma - C^k_{n,a_n}$. The next lemma bounds the probability that this success event does not occur, so the estimation for epoch k failed.

Lemma 1 (Exploration Error Probability)**.** *Let $\left\{\mu^k_{n,i}\right\}$ be the estimated reward expectations using all the exploration phases up to epoch k, with confidence intervals $\left\{C^k_{n,i}\right\}$. Define the minimal gap by*

$$\Delta \triangleq \min_n \min_{i \neq j} \left| \mu_{n,i} - \mu_{n,j} \right|. \tag{5}$$

Define the k-th exploration error event as

$$E_{e,k} = \left\{ \exists n, i \,\middle|\, \left| \mu^k_{n,i} - \mu_{n,i} \right| \geq C^k_{n,i} \text{ or } C^k_{n,i} \geq \frac{\Delta}{4} \right\}. \tag{6}$$

Then for all $k > k_0$ for a large enough constant k_0 we have

$$\mathbb{P}\left(E_{e,k}\right) \leq 3NMe^{-\frac{c_1}{6}k}. \tag{7}$$

Proof. After the k-th exploration phase, the estimation of the expected rewards is based on $T_e(k)$ samples, and

$$T_e(k) \geq c_1 \sum_{i=1}^{k} \log(i+1) \geq c_1 \frac{k}{2} \log \frac{k}{2}. \tag{8}$$

Let $A_{n,i}(t)$ be the indicator that is equal to one if only player n chose arm i at time t. Define $V_{n,i}$, as the number of visits of player n to arm i with no collision up to time t, and $V_m = \min\limits_{n,i} V_{n,i}$. The exploration phase consists of uniform and independent arm choices, so $\mathbb{P}\left(A_{n,i}(t) = 1\right) = \frac{1}{M}\left(1 - \frac{1}{M}\right)^{N-1}$. We show that each player pulls each arm many times without collisions. Formally:

$$\begin{aligned}
\mathbb{P}\left(V_m < \frac{T_e(k)}{5M}\right) &= \mathbb{P}\left(\bigcup_{i=1}^{M} \bigcup_{n=1}^{N} \left\{ V_{n,i} < \frac{T_e(k)}{5M} \right\}\right) \\
&\underset{(a)}{\leq} NM\mathbb{P}\left(V_{1,1} < \frac{T_e(k)}{5M}\right) \\
&\underset{(b)}{\leq} NMe^{-2\frac{1}{M^2}\left(\left(1-\frac{1}{M}\right)^{N-1}-\frac{1}{5}\right)^2 T_e(k)} \\
&\underset{(c)}{\leq} NMe^{-\frac{1}{18M^2}T_e(k)}
\end{aligned} \tag{9}$$

where (a) is a union bound, (b) is Hoeffding's inequality for Bernoulli random variables and (c) follows since $M \geq N$ and $\left(1-\frac{1}{M}\right)^{M-1}-\frac{1}{5} \geq e^{-1}-\frac{1}{5} > \frac{1}{6}$. By Hoeffding's inequality for random variables (Hoeffding, 1994) on $[0,1]$

$$\mathbb{P}\left(\bigcup_{n=1}^{N}\bigcup_{i=1}^{M}\left\{\left|\mu_{n,i}^{k}-\mu_{n,i}\right| \geq C_{n,i}^{k}\right\} \middle| \left\{V_{n,i}\right\}\right) \\ \leq \sum_{n=1}^{N}\sum_{i=1}^{M} 2e^{-2V_{n,i}\left(C_{n,i}^{k}\right)^{2}} \underset{(a)}{\leq} 2NMe^{-2\frac{MV_m}{\log V_m}} \quad (10)$$

where (a) uses $C_{n,i}^{k}=\sqrt{\frac{M}{\log V_{n,i}}}$ and $V_m \geq 3$. Now note that for all $k > k_0$ for a sufficiently large k_0:

$$\sqrt{\frac{M}{\log\left(\frac{T_e(k)}{5M}\right)}} \leq \sqrt{\frac{M}{\log\left(\frac{c_1\frac{k}{2}\log\frac{k}{2}}{5M}\right)}} < \frac{\Delta}{4} \quad (11)$$

and therefore $V_m \geq \frac{T_e(k)}{5M}$ implies $\max_{n,i} C_{n,i}^{k} < \frac{\Delta}{4}$. Hence, given (11) the event $\bigcup_{n=1}^{N}\bigcup_{i=1}^{M}\left\{\left|\mu_{n,i}^{k}-\mu_{n,i}\right| \geq C_{n,i}^{k}\right\}$ coincides with $E_{e,k}$, so for all $k > k_0$:

$$\mathbb{P}\left(E_{e,k}\middle| V_m \geq \frac{T_e(k)}{5M}\right) \\ = \mathbb{P}\left(\bigcup_{n=1}^{N}\bigcup_{i=1}^{M}\left\{\left|\mu_{n,i}^{k}-\mu_{n,i}\right| \geq C_{n,i}^{k}\right\} \middle| V_m \geq \frac{T_e(k)}{5M}\right) \\ \underset{(a)}{\leq} 2NMe^{-2\frac{\frac{T_e(k)}{5}}{\log\frac{T_e(k)}{5M}}} \quad (12)$$

where (a) uses the law of total probability with respect to $\{V_{n,i}\}$ with Bayes's rule on $\{V_m \geq \frac{T_e(k)}{5M}\}$, using the bound in (10). We conclude that for all $k > k_0$:

$$\mathbb{P}(E_{e,k}) = \mathbb{P}\left(E_{e,k}\middle| V_m < \frac{T_e(k)}{5M}\right)\mathbb{P}\left(V_m < \frac{T_e(k)}{5M}\right) \\ + \mathbb{P}\left(E_{e,k}\middle| V_m \geq \frac{T_e(k)}{5M}\right)\mathbb{P}\left(V_m \geq \frac{T_e(k)}{5M}\right) \\ \leq \mathbb{P}\left(V_m < \frac{T_e(k)}{5M}\right) + \mathbb{P}\left(E_{e,k}\middle| V_m \geq \frac{T_e(k)}{5M}\right) \\ \underset{(a)}{\leq} NMe^{-\frac{T_e(k)}{18M^2}} + 2NMe^{-\frac{2T_e(k)}{5\log\frac{T_e(k)}{5M}}} \quad (13)$$

where (a) uses (9) and (12). Finally, (7) follows by using (8) in (13) for a sufficiently large k. □

5. Matching Phase

In this section we analyze the matching phase, where the goal is to distributedly find γ-matchings based on the estimated expected rewards from the exploration phase. We conclude by upper bounding the probability that an optimal γ^*-matching is not played during the exploitation phase. During the matching phase, the rewards of the arms are ignored, as only the binary decision of whether an arm is better or worse than γ matters. These binary decisions induce the following bipartite graph between the N players and M arms:

Definition 3. Let G_k be the bipartite graph where edge (n,i) exists if and only if $\mu_{n,i}^{k} \geq \gamma_k - C_{n,i}^{k}$.

During the k-th matching phase, players follow our dynamics to switch arms in order to find a γ_k-matching in G_k. These γ_k-matchings (up to confidence intervals) are absorbing states in the sense that players stop switching arms if they are all playing a γ_k-matching. The dynamics of the players induce the following Markov chain:

Definition 4. Define $\mathcal{E}_n^k = \left\{i \,|\, \mu_{n,i}^{k} \geq \gamma_k - C_{n,i}^{k}\right\}$. The transition into $\boldsymbol{a}(t+1)$ is dictated by the transition of each player n:

1. If $\eta_{a_n(t)}(\boldsymbol{a}(t)) = 1$ then $a_n(t+1) = a_n(t)$ with probability 1.
2. If $\eta_{a_n(t)}(\boldsymbol{a}(t)) = 0$ then $a_n(t+1) = i$ with probability $\frac{1}{|\mathcal{E}_n^k|}$ for all $i \in \mathcal{E}_n^k$.

Note that the matchings in G_k are γ_k-matchings only when the confidence intervals are small enough. Next we prove that if a matching exists in G_k, then the matching phase will find it with a probability that goes to one. However, we do not need this probability to converge to one, but simply to exceed a large enough constant.

Lemma 2. *Let $\mathcal{G}_{N,M}$ be the set of all bipartite graphs with N left vertices and M right vertices that have a matching of size N. Define the random variable $\tau(G,\boldsymbol{a}(0))$ as the first time the process of Definition 4, $\{\boldsymbol{a}(t)\}$, constitutes a matching of size N, starting from $\boldsymbol{a}(0)$. Define*

$$\bar{\tau} = \max_{G\in\mathcal{G}_{N,M},\boldsymbol{a}(0)} \mathbb{E}\left\{\tau\left(G,\boldsymbol{a}(0)\right)\right\}. \quad (14)$$

If G_k permits a matching then the k-th matching phase converges to a matching with probability $p \geq 1-\frac{\bar{\tau}}{\lceil c_2\log(k+1)\rceil}$.

Proof. We start by noting that the process $\boldsymbol{a}(t)$ that evolves according to the dynamics in Definition 4 is a Markov chain. This follows since all transitions are a function of $\boldsymbol{a}(t)$ alone, with no dependence on $\boldsymbol{a}(t-1),...,\boldsymbol{a}(0)$ given $\boldsymbol{a}(t)$. Let $\mathcal{M}$ be a matching in G_k. Define $\Phi_{\mathcal{M}}(\boldsymbol{a})$ to be the number of players that are playing in $\boldsymbol{a}$ the arm they are matched to in $\mathcal{M}$. Observe the process $\Phi_{\mathcal{M}}(\boldsymbol{a}(t))$. If there are no colliding players, then $\boldsymbol{a}(t)$ is a matching (potentially different from $\mathcal{M}$) and no player will ever change their chosen arm. Otherwise, for every collision, at least one of the colliding players is not playing their arm in $\mathcal{M}$. There is a positive probability that this player will pick their arm in $\mathcal{M}$ at random and all other players will stay with the same arm. Hence, if $\boldsymbol{a}\left(t\right)$ is not a matching, then there is a positive probability that $\Phi_{\mathcal{M}}\left(\boldsymbol{a}\left(t+1\right)\right)=\Phi_{\mathcal{M}}\left(\boldsymbol{a}\left(t\right)\right)+1$. We conclude that every non-matching $\boldsymbol{a}$ has a positive probability path to a matching, making $\boldsymbol{a}\left(t\right)$ an absorbing Markov chain with the matchings as the absorbing states. By Markov's inequality

$$\begin{aligned}&\mathbb{P}\big(\tau\left(G_k,\boldsymbol{a}(0)\right)\geq\lceil c_2\log(k+1)\rceil\big) \qquad (15)\\ &\leq\frac{\mathbb{E}\left\{\tau\left(G_k,\boldsymbol{a}(0)\right)\right\}}{\lceil c_2\log(k+1)\rceil}\leq\frac{\bar{\tau}}{\lceil c_2\log(k+1)\rceil}. \qquad \square\end{aligned}$$

Intriguingly, the Bernoulli trials stemming from trying to find a matching in $\{G_\ell\}$ over consecutive epochs are *dependent*, as after enough successes, there will no longer be a matching in G_ℓ, yielding success probability 0. The next Lemma shows that Hoeffding's inequality for binomial random variables still applies as long as there are few enough successes, such that there is still a matching in G_k.

Lemma 3. *Consider a sequence of i.i.d. Bernoulli random variables $X_1,\ldots,X_L$ with success probability p (or at least p for each trial). For $x<Lp$, consider $S_x=\sum_{i=1}^{L}X_i\mathbb{1}\{\sum_{j<i}X_j<x\}$. Then*

$$\mathbb{P}(S_x<x)\leq e^{-2L\left(p-\frac{x}{L}\right)^2}. \qquad (16)$$

Proof. If $S_x=m<x$ then $\sum_{i=1}^{L}X_i<x$, as otherwise the indicators in S_x of the first x indices i where $X_i=1$ will be active, and so $S_x\geq x$, contradicting $S_x=m<x$. Therefore

$$\mathbb{P}(S_x<x)\leq\mathbb{P}\left(\sum_{i=1}^{L}X_i<x\right)\leq e^{-2L\left(p-\frac{x}{L}\right)^2}. \qquad \square$$

We conclude this section by proving the main Lemma used to prove Theorem 1. The idea of the proof is to show that if the past $\frac{k}{2}$ exploration phases succeeded, and enough matching trials succeeded, then a γ^*-matching was found within the last $\frac{k}{2}$ matching phases. This ensures that a γ^*-matching is played during the k-th exploitation phase.

Lemma 4 (Exploitation Error Probability)**.** *Define the k-th exploitation error event E_k as the event where the actions $\tilde{\boldsymbol{a}}_{k^*}$ played in the k-th exploitation phase are not a γ^*-matching. Let k_0 be large enough such that for all $k>k_0$*

$$\varepsilon_{\lceil\frac{k}{2}\rceil}<\frac{\Delta}{4}\text{ and }1-\frac{\bar{\tau}}{\lceil c_2\log(\frac{k}{2}+1)\rceil}-\frac{1+\log k}{k/6}\geq\frac{3}{\sqrt{10}}. \qquad (17)$$

Then for all $k>k_0$ we have

$$\mathbb{P}\left(E_k\right)\leq 7NMe^{-\frac{c_1}{12}k}+e^{-\frac{3k}{10}}. \qquad (18)$$

Proof. Define $E_{c,\ell}$ as the event where a matching existed in G_ℓ and was not found in the ℓ-th matching phase. From Lemma 2 we know that if there is a matching in G_ℓ, then the ℓ-th trial has success probability at least $1-\frac{\bar{\tau}}{\lceil c_2\log(\ell+1)\rceil}$.

Next we bound from below the number of trials we have between resets in order to find a γ^* matching. We define $k_w\geq\left\lceil\frac{k}{2}\right\rceil$ as the first epoch since $\left\lceil\frac{k}{2}\right\rceil$ where a reset occurred (so $\gamma_{k_w}=0$). In the worst case the algorithm resets in epoch $\left\lceil\frac{k}{2}\right\rceil-1$. Even still, the algorithm will reset again no later than $k_w\leq\left\lceil\frac{k}{2}\right\rceil-1+\left\lceil\frac{\lceil\frac{k}{2}\rceil-1}{3}\right\rceil\leq\frac{2}{3}k$. The subsequent reset will then happen at k_{w+1}, where $k_{w+1}\leq\frac{2}{3}k+\left\lceil\frac{2}{9}k\right\rceil\leq\left\lceil\frac{8}{9}k\right\rceil<k$ for $k>9$. We conclude that during the past $\frac{k}{2}$ epochs, there exists at least one full period (from reset to reset) with length at least $\frac{k}{6}$. Recall the definition of the ℓ-th exploration error event $E_{e,\ell}$ in (6). Define the event $A_k=\bigcap_{\ell=\lceil\frac{k}{2}\rceil}^{k}\bar{E}_{e,\ell}$ for which $\bar{A}_k=\bigcup_{\ell=\lceil\frac{k}{2}\rceil}^{k}E_{e,\ell}$. We define β_{k_w} as the number of successful trials needed after reset w to reach $\gamma_k\geq\gamma^*-\frac{\Delta}{4}$. Note that $\beta_{k_w}\leq 1+\log k$ since no more than $1+\log k$ steps of size $\varepsilon_{k_w}=\frac{1}{1+\log k}$ are needed. Then for all $k>k_0$

$$\begin{aligned}\mathbb{P}(E_k\,|\,A_k)&\underset{(a)}{\leq}\mathbb{P}\left(\gamma_k<\gamma^*-\frac{\Delta}{4}\,\middle|\,A_k\right)\\ &\underset{(b)}{\leq}\mathbb{P}\left(\sum_{\ell=k_w}^{k_{w+1}}\mathbb{1}\left\{\bar{E}_{c,\ell}\right\}<\beta_{k_w}\,\middle|\,A_k\right)\\ &\underset{(c)}{\leq}e^{-2\left(1-\frac{\bar{\tau}}{\lceil c_2\log(\frac{k}{2}+1)\rceil}-\frac{1+\log k}{k_{w+1}-k_w}\right)^2(k_{w+1}-k_w)}\\ &\underset{(d)}{\leq}e^{-\frac{3k}{10}} \qquad (19)\end{aligned}$$

where (a) follows since given $\bigcap_{\ell=\lceil\frac{k}{2}\rceil}^{k}\bar{E}_{e,\ell}$, if $\gamma_k\geq\gamma^*-\frac{\Delta}{4}$ then a γ^*-matching was found before the k-th exploitation phase and E_k did not occur. This follows since at the last success at $\ell\leq k$ we must have then had that for all n

$$\begin{aligned}\mu_{n,a_n}&\geq\mu_{n,a_n}^{\ell}-C_{n,a_n}^{\ell}\geq\gamma_\ell-2C_{n,a_n}^{\ell}\\ &\geq\gamma^*-\frac{\Delta}{4}-\varepsilon_{k_w}-2C_{n,a_n}^{\ell}>\gamma^*-\Delta \qquad (20)\end{aligned}$$

which can only happen if $\mu_{n,a_n}\geq\gamma^*$. Inequality (b) in (19) follows by noting that the probability that $\max_{\lceil\frac{k}{2}\rceil\leq\ell\leq k}\gamma_\ell<\gamma^*-\frac{\Delta}{4}$ with a constant step size (between resets) ε_{k_w} implies fewer than $\left\lceil\frac{\gamma^*-\frac{\Delta}{4}}{\varepsilon_{k_w}}\right\rceil$ successful trials between k_w and k_{w+1}. Given A_k, in any trial $\ell\in[k_w,k_{w+1}]$ such that there have been fewer than $\left\lceil\frac{\gamma^*-\frac{\Delta}{4}}{\varepsilon_{k_w}}\right\rceil$

successes in $[k_w, \ell)$, at least one matching will exist in G_ℓ (an optimal matching a_n^*), since

$$\mu_{n,a_n^*}^{\ell} \geq \mu_{n,a_n^*} - C_{n,a_n^*}^{\ell} \geq \gamma^* - C_{n,a_n^*}^{\ell} \underset{(1)}{>} \gamma_\ell - C_{n,a_n^*}^{\ell} \quad (21)$$

where (1) follows since for all $k > k_0$, ε_{k_w} is sufficiently small such that $\gamma_\ell \leq \gamma^* - \frac{\Delta}{4} + \varepsilon_{k_w} < \gamma^*$. Inequality (c) in (19) follows from Lemma 3 with $p \triangleq 1 - \frac{\bar{\tau}}{\lceil c_2 \log(\frac{k}{2}+1) \rceil}$. Inequality (d) follows from $k_{w+1} - k_w \geq \frac{k}{6}$ and (17). Finally, (18) is obtained by:

$$\begin{aligned}
\mathbb{P}(E_k) &= \mathbb{P}\left(E_k \,|\bar{A}_k\right) \mathbb{P}\left(\bar{A}_k\right) + \mathbb{P}\left(E_k \,|A_k\right) \mathbb{P}\left(A_k\right) \\
&\underset{(a)}{\leq} \left(3NM \sum_{\ell=\lceil \frac{k}{2} \rceil}^{k} e^{-\frac{c_1}{6}\ell}\right) + e^{-\frac{3k}{10}} \\
&\underset{(b)}{\leq} 3NMe^{-\frac{c_1}{12}k} \left(\frac{1-e^{-\frac{c_1}{12}k}}{1-e^{-\frac{c_1}{6}}}\right) + e^{-\frac{3k}{10}} \\
&\leq 7NMe^{-\frac{c_1}{12}k} + e^{-\frac{3k}{10}} \quad (22)
\end{aligned}$$

where (a) is a union bound of $\bar{A}_k = \bigcup_{\ell=\lceil \frac{k}{2} \rceil}^{k} E_{e,\ell}$ using Lemma 1 together with (19), and (b) is a geometric sum. □

6. Numerical Simulations

We simulated two multi-armed bandit games with the following expected rewards matrices:

$$U_1 = \begin{bmatrix} \frac{1}{2} & \frac{9}{10} & \frac{1}{10} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & \frac{1}{10} \\ \frac{1}{10} & \frac{1}{4} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{10} & \frac{9}{10} & \frac{1}{4} & \frac{1}{2} \end{bmatrix}$$

$$U_2 = \begin{bmatrix}
\frac{9}{10} & \frac{2}{5} & \frac{4}{5} & \frac{1}{10} & \frac{3}{10} & \frac{1}{20} & \frac{1}{5} & \frac{1}{10} & \frac{3}{10} & \frac{1}{5} \\
\frac{4}{10} & \frac{3}{10} & \frac{3}{10} & \frac{1}{10} & \frac{1}{5} & \frac{3}{10} & \frac{2}{5} & \frac{2}{5} & \frac{3}{10} & \frac{2}{5} \\
\frac{1}{10} & \frac{1}{20} & \frac{1}{10} & \frac{2}{5} & \frac{1}{10} & \frac{1}{5} & \frac{9}{10} & \frac{3}{10} & \frac{2}{5} & \frac{1}{10} \\
\frac{1}{20} & \frac{1}{10} & \frac{9}{10} & \frac{1}{5} & \frac{9}{10} & \frac{3}{4} & \frac{1}{10} & \frac{9}{10} & \frac{1}{4} & \frac{1}{20} \\
\frac{4}{5} & \frac{3}{10} & \frac{1}{10} & \frac{7}{10} & \frac{1}{10} & \frac{2}{5} & \frac{1}{20} & \frac{1}{5} & \frac{3}{4} & \frac{1}{20} \\
\frac{2}{5} & \frac{1}{20} & \frac{3}{10} & \frac{7}{10} & \frac{1}{20} & \frac{1}{10} & \frac{1}{4} & \frac{3}{4} & \frac{3}{5} & \frac{1}{20} \\
\frac{9}{10} & \frac{3}{10} & \frac{3}{10} & \frac{4}{5} & \frac{1}{10} & \frac{1}{4} & \frac{7}{10} & \frac{1}{20} & \frac{1}{5} & \frac{3}{10} \\
\frac{3}{10} & \frac{1}{10} & \frac{2}{5} & \frac{1}{4} & \frac{1}{20} & \frac{9}{10} & \frac{1}{4} & \frac{1}{10} & \frac{1}{20} & \frac{2}{5} \\
\frac{4}{5} & \frac{3}{4} & \frac{1}{10} & \frac{1}{5} & \frac{2}{5} & \frac{1}{20} & \frac{3}{10} & \frac{1}{5} & \frac{1}{10} & \frac{1}{4} \\
\frac{2}{5} & \frac{2}{5} & \frac{9}{10} & \frac{7}{10} & \frac{1}{4} & \frac{1}{5} & \frac{1}{20} & \frac{1}{10} & \frac{2}{5} & \frac{1}{4}
\end{bmatrix}.$$

Given expected rewards $\{\mu_{n,i}\}$, the rewards are generated as $r_{n,i}(t) = \mu_{n,i} + z_{n,i}(t)$ where $\{z_{n,i}(t)\}$ are independent and uniformly distributed on $[-0.05, 0.05]$ for each n, i. The chosen parameters were $c_1 = 1000$ and $c_2 = 2000$ and $c_3 = 4000$ for all experiments, and are chosen to ensure that the additive constant C_0 is small (since k_0 is small), as the exploration and matching phases are long enough from the beginning.

At the beginning of the k-th matching phase, each player played her action from the last exploitation phase if it is in $\mathcal{E}_n^k$, or a random action from $\mathcal{E}_n^k$ otherwise. Although it has no effect on the theoretical bounds, it improved the performance in practice significantly. Another practical improvement was achieved by introducing a factor of 0.01 to the confidence intervals, which requires larger c_1 but does not affect the analysis otherwise. The step size sequence, that is updated only on resets, was chosen as $\varepsilon_k = \frac{0.2}{1+\log k}$.

In Fig. 1, we present the total expected regret versus time, averaged over 100 realizations, with U_1 as the expected reward matrix ($N = 4$). The shaded area denotes one standard deviation around the mean. This scenario has 24 matchings - 16 with minimal expected reward $\frac{1}{10}$, 7 with $\frac{1}{4}$, and one optimal matching with $\frac{1}{2}$. It can be seen that in all 100 experiments the players learned the max-min optimal matching by the end of the third epoch. This suggests that k_0 is much smaller than our theoretical bound. As expected, the regret scales (approximately) logarithmically. For comparison, the optimal sum of expected rewards for U_1 is 2.15, but the matching that achieves it has a minimal expected reward of $\frac{1}{4}$. Hence, a multi-player bandit algorithm that optimizes the expected sum of rewards will have regret $\Omega\left(\frac{T}{4}\right)$.

In Fig. 2, we present the total expected regret versus time, averaged over 100 realizations, with U_2 as the expected reward matrix ($N = 10$). The shaded area denotes one standard deviation around the mean. In this scenario only 136 matchings out of the $10! = 3628800$ are optimal with minimal utility of 0.4. There are 3798 matchings with 0.3, 16180 with 0.25, 62066 with 0.2, 785048 with 0.1 and 2761572 with 0.05. With more players and arms, k_0 is larger, but players still learn the max-min optimal matching by the sixth epoch. Again, the regret scales (approximately) logarithmically as guaranteed by Theorem 1. For comparison, the optimal sum of expected rewards for U_2 is 7.35, but the matching that achieves it has a minimal expected reward of 0.3. Hence, a multi-player bandit algorithm that optimizes the expected sum of rewards will have regret $\Omega\left(\frac{T}{10}\right)$.

7. Conclusions and Future Work

We studied a multi-player multi-armed bandit game where players cooperate to learn how to allocate arms, thought of as resources, so as to maximize the minimal expected reward received by any player. To allow for a meaningful notion of fairness, we employed the heterogeneous model where arms can have different expected rewards for each player. Our algorithm operates in the restrictive setting of bandit feedback, where each player only observes the reward for the arm she played and cannot observe the actions or rewards of other players. We proposed a novel fully dis-

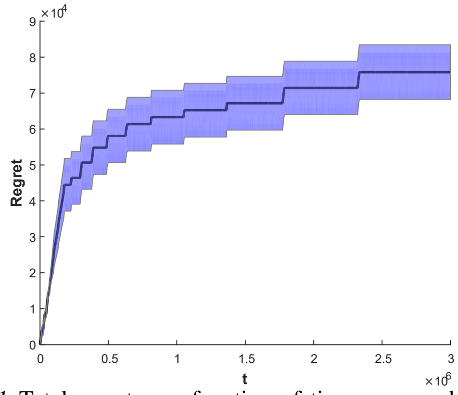

Figure 1. Total regret as a function of time, averaged over 100 experiments and with U_1 ($N = 4$).

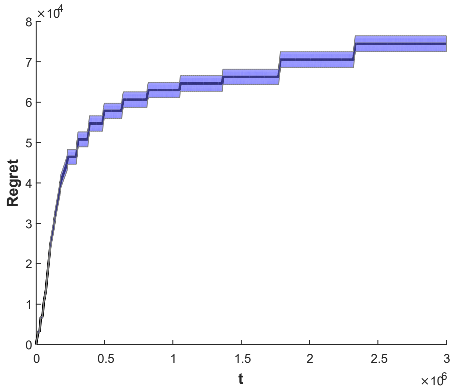

Figure 2. Total regret as a function of time, averaged over 100 experiments and with U_2 ($N = 10$).

tributed algorithm that achieves a near order optimal total expected regret of $O(\log\log T \log T)$, where $\log\log T$ can be improved to any increasing function of T.

It is still an open question whether a total expected regret of $O(\log T)$ is achievable in our scenario, when the problem parameters are unknown. Following our discussion on the additive constant C_0, an algorithm that achieves $O(\log T)$ even with unknown parameters will expose the multiplicative factors of the $\log T$ in the regret bound, and their dependence on Δ and M. These factors are likely to be strongly affected by the time it takes the algorithm to find a matching. If some limited communication is allowed, more sophisticated algorithms to distributedly compute the matching are possible, based on gossip, message passing, or auctions (Bayati et al., 2008; Naparstek & Leshem, 2016). These algorithms do not need a consensus phase, eliminating the M factor in (4) and reducing the convergence time $\bar{\tau}$, but it is unclear if these approaches can achieve $O(\log T)$ regret. Focusing on our setting, an interesting question is whether one can design a better distributed matching algorithm that can operate with no communication between players.

In some applications one is interested in guaranteeing a target QoS for each user that is "good enough" (see (Katz-Samuels & Jamieson, 2020; Lai & Robbins, 1984) for the single-player case). This is a weaker requirement than max-min fairness between users. Hence, an interesting open question is whether better regret bounds for our multi-player bandit scenario can be obtained in this case.

8. Proof of Theorem 1

Let K be the number of epochs that start within the T turns. Since

$$T \geq \sum_{k=1}^{K-1}\left(c_1 \log k + c_2 \log k + M + c_3\left(\frac{4}{3}\right)^k\right) \geq 3c_3\left(\left(\frac{4}{3}\right)^K - \frac{4}{3}\right) \tag{23}$$

K is upper bounded by $K \leq \log_{\frac{4}{3}}\left(\frac{T}{3c_3} + \frac{4}{3}\right)$. Let k_0 be a constant epoch index that is large enough for the bounds of Lemma 1, Lemma 4, and inequality (c) in (24) to hold. Intuitively, this is the epoch after which the matching phase duration is long enough, the step size ε_k is small enough, and the confidence intervals are sufficiently tight. Define E_k as the event where a γ^*-matching is not played in the k-th exploitation phase. We now bound the total expected regret of epoch $k > k_0$, denoted by R_k:

$$\begin{aligned}R_k &\leq M + (c_1+c_2)\log(k+1) + \mathbb{P}(E_k)\, c_3\left(\frac{4}{3}\right)^k + 3\\ &\underset{(a)}{\leq} M + (c_1+c_2)\log(k+1) + 3\\ &\qquad + \left(7NMe^{-\frac{c_1}{12}k} + e^{-\frac{3k}{10}}\right)c_3\left(\frac{4}{3}\right)^k\\ &\underset{(b)}{\leq} M + 3 + (c_1+c_2)\log(k+1) + 8NMc_3\beta^k\\ &\underset{(c)}{\leq} M + 2(c_1+c_2)\log k \end{aligned}\tag{24}$$

where (a) uses Lemma 4, (b) follows for some constant $\beta < 1$ since $e^{-\frac{3}{10}} < \frac{3}{4}$ and $c_1 \geq 4$ and (c) follows for $k > k_0$. We conclude that, for some additive constant C_0,

$$\begin{aligned}R(T) = \sum_{k=1}^{K} R_k &\underset{(a)}{\leq} MK + 2\sum_{k=k_0+1}^{K}(c_1+c_2)\log k\\ &\qquad + \sum_{k=1}^{k_0}\left((c_1+c_2)\log(k+1) + c_3\left(\frac{4}{3}\right)^k + 3\right)\\ &\leq C_0 + MK + 2(c_1+c_2)K\log K\end{aligned}\tag{25}$$

where (a) follows by completing the last epoch to a full epoch which only increases $R(T)$, and by using (24). Then, we obtain (4) by upper bounding $K \leq \log_{\frac{4}{3}}\left(\frac{T}{3c_3} + \frac{4}{3}\right) \leq \log_{\frac{4}{3}}\left(\frac{T}{c_3}\right)$, where the second inequality is only used to simplify the expression, and holds for all $T \geq 2c_3$.

Acknowledgements

The authors gratefully acknowledge funding from the Koret Foundation grant for Smart Cities and Digital Living, NSF GRFP, Alcatel-Lucent Stanford Graduate Fellowship, and ISF grant 1644/18.

References

Alatur, P., Levy, K. Y., and Krause, A. Multi-player bandits: The adversarial case. *arXiv preprint arXiv:1902.08036*, 2019.

Anandkumar, A., Michael, N., Tang, A. K., and Swami, A. Distributed algorithms for learning and cognitive medium access with logarithmic regret. *IEEE Journal on Selected Areas in Communications*, 29(4):731–745, 2011.

Asadpour, A. and Saberi, A. An approximation algorithm for max-min fair allocation of indivisible goods. *SIAM Journal on Computing*, 39(7):2970–2989, 2010.

Avner, O. and Mannor, S. Concurrent bandits and cognitive radio networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 66–81, 2014.

Avner, O. and Mannor, S. Multi-user lax communications: a multi-armed bandit approach. In *INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE*, pp. 1–9, 2016.

Bar-On, Y. and Mansour, Y. Individual regret in cooperative nonstochastic multi-armed bandits. In *Advances in Neural Information Processing Systems*, pp. 3110–3120, 2019.

Bayati, M., Shah, D., and Sharma, M. Max-product for maximum weight matching: Convergence, correctness, and lp duality. *IEEE Transactions on Information Theory*, 54(3):1241–1251, 2008.

Besson, L. and Kaufmann, E. Multi-player bandits revisited. In *Algorithmic Learning Theory*, pp. 56–92, 2018.

Bistritz, I. and Leshem, A. Distributed multi-player bandits-a game of thrones approach. In *Advances in Neural Information Processing Systems*, pp. 7222–7232, 2018.

Boursier, E. and Perchet, V. SIC-MMAB: synchronisation involves communication in multiplayer multi-armed bandits. In *Advances in Neural Information Processing Systems*, pp. 12048–12057, 2019.

Boursier, E., Perchet, V., Kaufmann, E., and Mehrabian, A. A practical algorithm for multiplayer bandits when arm means vary among players. *arXiv preprint arXiv:1902.01239*, 2019.

Bubeck, S., Cesa-Bianchi, N., et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

Bubeck, S., Li, Y., Peres, Y., and Sellke, M. Non-stochastic multi-player multi-armed bandits: Optimal rate with collision information, sublinear without. *arXiv preprint arXiv:1904.12233*, 2019.

Cohen, J., Héliou, A., and Mertikopoulos, P. Learning with bandit feedback in potential games. In *Proceedings of the 31th International Conference on Neural Information Processing Systems*, 2017.

Darak, S. J. and Hanawal, M. K. Multi-player multi-armed bandits for stable allocation in heterogeneous ad-hoc networks. *IEEE Journal on Selected Areas in Communications*, 37(10):2350–2363, 2019.

Evirgen, N. and Kose, A. The effect of communication on noncooperative multiplayer multi-armed bandit problems. In *arXiv preprint arXiv:1711.01628, 2017*, 2017.

Hanawal, M. K. and Darak, S. J. Multi-player bandits: A trekking approach. *arXiv preprint arXiv:1809.06040*, 2018.

Hoeffding, W. Probability inequalities for sums of bounded random variables. In *The Collected Works of Wassily Hoeffding*, pp. 409–426. Springer, 1994.

Jabbari, S., Joseph, M., Kearns, M., Morgenstern, J., and Roth, A. Fairness in reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1617–1626. JMLR. org, 2017.

Joseph, M., Kearns, M., Morgenstern, J. H., and Roth, A. Fairness in learning: Classic and contextual bandits. In *Advances in Neural Information Processing Systems*, pp. 325–333, 2016.

Kalathil, D., Nayyar, N., and Jain, R. Decentralized learning for multiplayer multiarmed bandits. *IEEE Transactions on Information Theory*, 60(4):2331–2345, 2014.

Katz-Samuels, J. and Jamieson, K. The true sample complexity of identifying good arms. In *International Conference on Artificial Intelligence and Statistics*, pp. 1781–1791, 2020.

Lai, L., Jiang, H., and Poor, H. V. Medium access in cognitive radio networks: A competitive multi-armed bandit framework. In *Signals, Systems and Computers, 2008 42nd Asilomar Conference on*, pp. 98–102, 2008.

Lai, T. L. and Robbins, H. Asymptotically optimal allocation of treatments in sequential experiments. *Design of Experiments: Ranking and Selection*, pp. 127–142, 1984.

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

Liu, H., Liu, K., and Zhao, Q. Learning in a changing world: Restless multiarmed bandit with unknown dynamics. *IEEE Transactions on Information Theory*, 59 (3):1902–1916, 2013.

Liu, K. and Zhao, Q. Distributed learning in multi-armed bandit with multiple players. *IEEE Transactions on Signal Processing*, 58(11):5667–5681, 2010.

Liu, L. T., Mania, H., and Jordan, M. I. Competing bandits in matching markets. *arXiv preprint arXiv:1906.05363*, 2019.

Magesh, A. and Veeravalli, V. V. Multi-player multi-armed bandits with non-zero rewards on collisions for uncoordinated spectrum access. *arXiv preprint arXiv:1910.09089*, 2019.

Mo, J. and Walrand, J. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on networking*, (5):556–567, 2000.

Naparstek, O. and Leshem, A. Expected time complexity of the auction algorithm and the push relabel algorithm for maximum bipartite matching on random graphs. *Random Structures & Algorithms*, 48(2):384–395, 2016.

Nayyar, N., Kalathil, D., and Jain, R. On regret-optimal learning in decentralized multi-player multi-armed bandits. *IEEE Transactions on Control of Network Systems*, PP(99):1–1, 2016.

Radunovic, B. and Le Boudec, J.-Y. A unified framework for max-min and min-max fairness with applications. *IEEE/ACM Transactions on networking*, 15(5): 1073–1083, 2007.

Rosenski, J., Shamir, O., and Szlak, L. Multi-player bandits–a musical chairs approach. In *International Conference on Machine Learning*, pp. 155–163, 2016.

Sankararaman, A., Ganesh, A., and Shakkottai, S. Social learning in multi agent multi armed bandits. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3):1–35, 2019.

Tibrewal, H., Patchala, S., Hanawal, M. K., and Darak, S. J. Distributed learning and optimal assignment in multiplayer heterogeneous networks. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pp. 1693–1701. IEEE, 2019.

Vakili, S., Liu, K., and Zhao, Q. Deterministic sequencing of exploration and exploitation for multi-armed bandit problems. *IEEE Journal of Selected Topics in Signal Processing*, 7(5):759–767, 2013.

Wei, K., Iyer, R. K., Wang, S., Bai, W., and Bilmes, J. A. Mixed robust/average submodular partitioning: Fast algorithms, guarantees, and applications. In *Advances in Neural Information Processing Systems*, pp. 2233–2241, 2015.

Zehavi, E., Leshem, A., Levanda, R., and Han, Z. Weighted max-min resource allocation for frequency selective channels. *IEEE transactions on signal processing*, 61 (15):3723–3732, 2013.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. Learning fair representations. In *International Conference on Machine Learning*, pp. 325–333, 2013.