# Distributed Multi-Player Bandits - a Game of Thrones Approach

Ilai Bistritz and Amir Leshem

**Abstract**

We consider a multi-armed bandit game where $N$ players compete for $K$ arms for $T$ turns. Each player has different expected rewards for the arms, and the instantaneous rewards are independent and identically distributed. Performance is measured using the expected sum of regrets, compared to the optimal assignment of arms to players. We assume that each player only knows her actions and the reward she received each turn. Players cannot observe the actions of other players, and no communication between players is possible. We present a distributed algorithm and prove that it achieves an expected sum of regrets of near-$O\left(\log^2 T\right)$. This is the first algorithm to achieve a poly-logarithmic regret in this fully distributed scenario. All other works have assumed that either all players have the same vector of expected rewards or that communication between players is possible.

## I. Introduction

In online learning problems, an agent needs to learn on the run how to behave optimally. The crux of these problems is the trade-off between exploration and exploitation. This trade-off is well captured by the multi-armed bandit problem, which has attracted enormous attention from the research community. Recently, there has been a growing interest in the case of the multi-player multi-armed bandit. In the multi-player scenario, the nature of the interaction between the players can take many forms. Players may want to solve the problem of finding the best mutual arm as a team [1]–[6], or may compete over the arms as resources they all individually require [7]–[19].

The idea of regret in the competitive multi-player multi-armed bandit problem is the expected sum of regrets and is defined as the performance loss compared to the optimal assignment of arms to players. The rationale for this notion of regret is formulated from the designer's perspective, who wants the distributed system of individuals to converge to a globally good solution.

Many works have considered a scenario where all the players have the same expectations for the rewards of all arms. Some of these works assume that communication between players is possible [10]–[12], [14], [19], whereas others consider a fully distributed scenario [7], [13], [15].

One of the main reasons for studying resource allocation bandits is their applications in cognitive radio or wireless networks in general. In these scenarios, the channels are interpreted as arms and the channel gains as the rewards. However, since

Ilai Bistritz is with the department of Electrical Engineering, Stanford University, Stanford, CA, USA, e-mail: bistritz@stanford.edu.
Amir Leshem is with the Faculty of Engineering, Bar-Ilan University, Ramat-Gan, Israel, e-mail: leshema@eng.biu.ac.il.

This research was supported by the Israeli Ministry of Science and Technology under grant 3-13038 and by a joint ISF-NRF research grant number 2277/16.

users are scattered in space, the physical reality dictates that different arms have different expected channel gains for different players.

This essential generalization for a matrix of expectations introduces a famous combinatorial optimization problem known as the assignment problem [20]. Achieving a sublinear expected sum of regrets in a distributed manner requires a distributed solution to the assignment problem, which by itself has been explored extensively, e.g. [21], [22].

This generalization was first considered in [9], and later enhanced in [8], where an algorithm that achieves an expected sum of regrets of near-$O\left(\log T\right)$ was presented. However, this algorithm requires communication between players. It is based on the distributed auction algorithm in [21], which is not fully distributed. It requires that players can observe the bids of other players. This was possible in [8], [9] since it was assumed that the players could observe the actions of other players, which allows them to communicate by using the arm choices as a signaling method. In [19], the authors suggest an algorithm that only assumes that users can sense all the channels without knowing which channels was chosen by whom. This algorithm requires less communication than [8], but has no regret guarantees. In wireless networks, assuming that each user can hear all other transmissions (fully connected network) is very demanding in practice. In a fully distributed scenario, players only have access to their previous actions and rewards. However, to date there is no completely distributed algorithm that converges to the exact optimal solution of the assignment problem. The fully distributed multi-armed bandit problem remains unresolved.

Our work generalizes [7] for different expectations for different players and [8], [9], [19] for a fully distributed scenario with no communication between players.

Recently, very powerful payoff-based dynamics were introduced [23]–[25]. These dynamics only require each player to know her own action and the reward she received for that action. Specifically, the dynamics in [24] guarantee that the optimal sum of utilities strategy profile will be played a sufficiently large portion of the time, even if it is not a Nash equilibrium. The crucial issue of applying these results to our case is that they all assume interdependent games. In an interdependent game, each group of players can always influence at least one player from outside this group. In the multiplayer multi-armed bandit collision model, this does not hold. A player in a collision receives zero reward. Nothing that other players (who chose other arms) can do will change that.

In this paper, we suggest novel modified dynamics that behave similarly to [24], but in our non-interdependent game. Specifically, they guarantee that the optimal solution to the assignment problem is played a considerable amount of time. We present a fully distributed multi-player multi-armed bandit algorithm for the resource allocation and collision scenario, based on these modified dynamics. By fully distributed we mean that players only have access to their own actions and rewards. This is the first algorithm that achieves a poly-logarithmic expected sum of regrets, near- $O\left(\log^2 T\right)$, with a matrix of expected rewards and no communication at all between players.

## II. Problem Formulation

We consider a stochastic game with the set of players $\mathcal{N}=\{1,...,N\}$ and a finite time horizon $T$. The horizon $T$ is not known in advance by any of the players. The discrete turn index is denoted by $t$. The strategy space of each player is a set of $K$ arms with indices that are denoted by $i,j=1,...,K$. We assume that $K\geq N$. At each turn $t$, all players simultaneously pick one arm each. The arm that player $n$ chooses at time $t$ is $a_n(t)$ and the strategy profile at time $t$ is $\boldsymbol{a}(t)$. Players do not know which arms the other players chose, and need not even know how many other players are there.

Define the set of players that chose arm $i$ in strategy profile $\boldsymbol{a}$

$$\mathcal{N}_i(\boldsymbol{a}) = \{n \,|\, a_n = i\}. \tag{1}$$

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

$$\eta_i(\boldsymbol{a}) = \begin{cases} 0 & \left|\mathcal{N}_i(\boldsymbol{a})\right| > 1 \\ 1 & o.w. \end{cases}. \tag{2}$$

The instantaneous utility of player $n$ in strategy profile $\boldsymbol{a}(t)$ in time $t$ is

$$\upsilon_n(\boldsymbol{a}(t)) = r_{n,a_n(t)}(t)\,\eta_{a_n(t)}(\boldsymbol{a}(t)) \tag{3}$$

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 $\{r_{n,i}(t)\}_t$ of arm $i$ for player $n$ is i.i.d. ("in time") with expectation $\mu_{n,i}$.

Next we define the expected total regret, which we want our distributed algorithm to minimize.

**Definition 1.** Denote the expected utility of player $n$ in strategy profile $\boldsymbol{a}$ by $g_n(\boldsymbol{a}) = E\{\upsilon_n(\boldsymbol{a})\}$. The total regret is defined as the random variable

$$R = \sum_{t=1}^{T}\sum_{n=1}^{N}\upsilon_n(\boldsymbol{a}^*) - \sum_{t=1}^{T}\sum_{n=1}^{N} r_{n,a_n(t)}(t)\,\eta_{a_n(t)}(\boldsymbol{a}(t)) \tag{4}$$

where

$$\boldsymbol{a}^* = \arg\max_{\boldsymbol{a}} \sum_{n=1}^{N} g_n(\boldsymbol{a}). \tag{5}$$

The expected total regret $\bar{R} \triangleq E\{R\}$ is the average of (4) over the randomness of the rewards $\{r_{n,i}(t)\}_t$, that dictate the random actions $\{a_n(t)\}$.

The problem in (5) is no other than the famous assignment problem [20] on the $N \times K$ matrix of expectations $\{\mu_{n,i}\}$. In this sense, our problem is a generalization of the distributed assignment problem to an online learning framework.

Assuming continuously distributed rewards is well justified in wireless networks. Given no collision, the quality of an arm (channel) always has a continuous measure like the SNR or the channel gain. However, this assumption is only used in two arguments and can be easily replaced without changing the analysis in this paper. The first argument is that since the probability for zero reward in a non-collision is zero, players can safely rule out collisions in their estimation of the expected reward. In the case where the probability for a zero reward is not zero, we can assume instead that each player can observe her collision indicator in addition to her reward. Knowing whether other players chose the same arm is a very modest requirement compared to assuming that players can observe the actions of other players. The second argument is that the continuity of the rewards' distributions makes the solution of (5), with the estimated expectations, unique with probability 1. We can assume instead that $\{\mu_{n,i}\}$ are generated at random using a continuous distribution, so the optimal solution is unique with probability 1 (i.e., "for

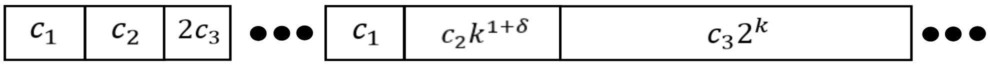

Fig. 1. Epochs structure. Depicted are the first and the $k$-th epochs.

almost all games”), with arbitrary distributions for the rewards that have expectations $\{\mu_{n,i}\}$.

According to the seminal work in [26], the optimal regret of the single-player case is logarithmic; i.e., $O(\log T)$. Players do not help each other; hence, we expect the expected total regret lower bound to be logarithmic at best. The next proposition shows that this is indeed the case.

**Proposition 2.** *The expected total regret is at least* $\Omega(\log T)$.

*Proof:* For $N = 1$, the result directly follows from [26]. Now we formally prove that more players cannot help. Assume that for $N > 1$ there is a policy that results in a better total expected regret than $\mathit{\Omega}(\log T)$. Hence, there must exist a player for which the personal regret is also better than $\mathit{\Omega}(\log T)$. This player, denoted player $n$, can simulate $N-1$ other players and generate at random their expectations and rewards, all of which are independent of the actual rewards she receives. This player also simulates the policies for other players, and even knows when a collision occurred for herself and can assign zero reward in that case. Hence, simulating $N-1$ fictitious players is a valid single player multi-armed bandit policy that violates the $\mathit{\Omega}(\log T)$ bound, which is a contradiction. We conclude that this bound is also valid for $N > 1$. ■

## III. The Game of Thrones Algorithm

When all players have the same arm expectations, the exploration phase is used to identify the $N$ best arms. Once the best arms are identified, players need to coordinate to be sure that each of them will sit on a different “chair” (see the Musical Chairs algorithm in [7]). When players have different arm expectations, a non-cooperative **game** is induced where the estimated expected rewards serve as utilities. In this game, players cannot sit on an ordinary chair without causing a linear regret, and must strive for a single **throne**. This throne is the arm they must play in the allocation that maximizes the sum of the expected rewards of all players. Any other solution will result in linear (in $T$) expected total regret. Note that our assignment problem has a unique optimal allocation with probability 1 (as shown in Lemma 11).

The total time needed for exploration increases with $T$ since the cost of being wrong becomes higher. When $T$ is known to the players, a long enough exploration can be accomplished at the beginning of the game. In order to maintain the right balance between exploration and exploitation when $T$ is not known in advance to the players, we divide the $T$ turns into epochs, one starting immediately after the other. Each epoch is further divided into three phases - exploration, Game of Thrones (GoT) and exploitation. During the exploration phase, players estimate the expected reward of each arm. The goal of the GoT phase is to let the players distributedly identify the optimal solution for the assignment problem on the estimated expected rewards from the exploration phase. It is done by playing a game with the estimated expectations as utilities, using random dynamics that probabilistically prefer strategy profiles with a higher sum of utilities. In the exploitation phase, each player plays the constant action she deduced from the GoT phase. The division into epochs is depicted in Fig. 1. The GoT Algorithm and GoT Dynamics are described in Algorithm 1 and Algorithm 2, respectively.

**Algorithm 1** Game of Thrones Algorithm

```
Initialization - Set $o_{n,i} = 0$ and $s_{n,i}(0) = 0$ for all $i$. Set $k = 1$, $\delta > 0$, $0 < \rho < 1$ and $\varepsilon > 0$.
For $t = 1, ..., T$
  1) Exploration Phase - for the next $c_1$ turns
     a) Sample an arm $i$ uniformly at random from all $K$ arms.
     b) 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.
     c) If $\eta_i(\boldsymbol{a}(t)) = 1$ then update $o_{n,i} = o_{n,i} + 1$ and $s_{n,i}(t) = s_{n,i}(t-1) + r_{n,i}(t)$.
     d) Estimate the expectation of the arm $i$ by $\mu^k_{n,i} = \frac{s_{n,i}(t)}{o_{n,i}}$, for each $i = 1, ..., K$.
  2) GoT Phase - for the next $c_2 k^{1+\delta}$ turns, play according to Algorithm 2 with $\varepsilon$ and $\rho$.
     a) Starting from the $d_g = \lceil \rho c_2 k^{1+\delta} \rceil$-th turn inside the GoT Phase, keep track on the number of times each action
        was played that resulted in being content

        $F^n_t(i) = \sum_{l=d_g}^{c_2 k^{1+\delta}} I(a_n(l) = i, M_n(l) = C)$        (6)

        where $I$ is the indicator function.
  3) Exploitation Phase - for the next $c_3 2^k$ turns, play

        $a^k_n = \arg\max_i F^n_t(i)$        (7)

  4) Update $k \leftarrow k + 1$.
End
```

**Algorithm 2** Game of Thrones Dynamics

```
Initialization - Let $c \geq N$. Each player $n$ has a personal state $M_n$, either content $C$ or discontent $D$, which determines her
mixed strategy. Each player also keeps a baseline action $\overline{a}_n$ and her utility is $u_n$. Denote $u_{n,\max} = \max_{\boldsymbol{a}} u_n(\boldsymbol{a})$.
In each turn during the GoT Phase
  • A content player chooses an action according to

        $p^{a_n}_n = \begin{cases} \frac{\varepsilon^c}{\left|\mathcal{A}_n\right|-1} & a_n \neq \overline{a}_n \\ 1 - \varepsilon^c & a_n = \overline{a}_n \end{cases}.$        (8)

  • A discontent player chooses an action uniformly at random; i.e.,

        $p^{a_n}_n = \frac{1}{|\mathcal{A}_n|}, \forall a_n \in \mathcal{A}_n.$        (9)

The transitions between $C$ and $D$ are determined as follows:
  • If $\overline{a}_n = a_n$ and $u_n > 0$, then a content player remains content with probability 1

        $[\overline{a}_n, C] \to [\overline{a}_n, C]$        (10)

  • If $\overline{a}_n \neq a_n$ or $u_n = 0$ or $M_n = D$, then ($C/D$ denoting either $C$ or $D$)

        $[\overline{a}_n, C/D] \to \begin{cases} [a_n, C] & \frac{u_n}{u_{n,\max}} \varepsilon^{u_{n,\max} - u_n} \\ [a_n, D] & 1 - \frac{u_n}{u_{n,\max}} \varepsilon^{u_{n,\max} - u_n} \end{cases}.$        (11)

End
```

In this paper, we prove the following main result.

**Theorem 3** (Main Theorem)**.** *Assume that the rewards $\{r_{n,i}(t)\}_t$ are independent in $n$ and i.i.d. in time $t$, with continuous distributions on $[0,1]$ with positive expectations $\{\mu_{n,i}\}$. Let the game have a finite horizon $T$, unknown to the players. Denote the optimal objective by $J_1 = \max_{\boldsymbol{a}} \sum_{n=1}^{N} g_n(\boldsymbol{a})$ and the second best one by $J_2$. Let each player play according to Algorithm 1, with a small enough $\varepsilon$, exploration phase length of $c_1 > \frac{16N^2K}{(J_1-J_2)^2}$ and $\delta > 0$. Then, for large enough $T$, the expected total* *regret is upper bounded by*

$$\bar{R} \leq 3c_2 N \log_2^{2+\delta}\left(\frac{T}{c_3}+2\right) = O\left(\log^{2+\delta} T\right). \tag{12}$$

*Proof:* Let $\delta > 0$. Denote the number of epochs that start within $T$ turns by $E$. Since

$$T \geq \sum_{k=1}^{E-1}\left(c_1 + c_2 k^{1+\delta} + c_3 2^k\right) \geq c_3\left(2^E - 2\right) \tag{13}$$

$E$ is upper bounded by $E \leq \log_2\left(\frac{T}{c_3}+2\right)$. Denote by $P_{e,k}$ and $P_{c,k}$ the error probabilities of the exploration and GoT phases of epoch $k$ respectively. Observe that if none of these errors occurred, the optimal solution to (5) is played in the $k$-th exploitation phase, which adds no additional regret to the total regret. We will prove in Lemma 5 and Lemma 13 that $P_{e,k} \leq 4K^2 e^{-k}$ and $P_{c,k} \leq A_0 e^{-\frac{c_2(1-\rho)}{1728 T_m\left(\frac{1}{8}\right)} k^{1+\delta}}$, where $A_0$ is a constant and $T_m\left(\frac{1}{8}\right)$ is the mixing time of the Markov chain of the GoT Dynamics. Note that $T_m\left(\frac{1}{8}\right)$ depends on $N, K$ and $\varepsilon$ , so there exists a $k_0$ such that for all $k > k_0$ we have

$$e^{-\frac{c_2(1-\rho)}{1728 T_m\left(\frac{1}{8}\right)} k^{\delta}} < \frac{1}{2}. \tag{14}$$

We now bound the expected total regret of epoch $k > k_0$, denoted by $\bar{R}_k$, as follows

$$\bar{R}_k \leq \left(c_1 + c_2 k^{1+\delta}\right) N + \left(4K^2 e^{-k} + A_0 e^{-\frac{c_2(1-\rho)}{1728 T_m\left(\frac{1}{8}\right)} k^{1+\delta}}\right) c_3 2^k N \leq c_1 N + 2A_0 c_3 N \beta^k + c_2 k^{1+\delta} N \tag{15}$$

for some constant $\beta < 1$. We conclude that, for some additive constant $C$,

$$\bar{R} = \sum_{k=1}^{E} \bar{R}_k \underset{(a)}{\leq} C + 2c_2 N \sum_{k=k_0+1}^{E} k^{1+\delta} \leq C + 2c_2 N E^{2+\delta} \underset{(b)}{\leq} C + 2c_2 N \log_2^{2+\delta}\left(\frac{T}{c_3}+2\right) \tag{16}$$

where (a) follows since completing the last epoch to a full epoch increases $\bar{R}_k$, and (b) is (13).

■

If either the exploration or the GoT phases fail, the regret becomes linear with $T$. Like many other online learning algorithms, we avoid a linear expected regret by ensuring that the error probabilities vanish with $T$. By using instead a single epoch with a constant duration for the first two phases, we obtain that with high probability (in $T$) our algorithm achieves a constant regret (as in [7]). However, our main result is formulated using the more conservative formulation of the expected regret.

## IV. Exploration Phase - Estimation of the Expected Rewards

In this section, we describe the exploration phase, and analyze its addition to the expected total regret. At the beginning of the game, players still do not have any evaluation of the $K$ different arms. They estimate these values on the run, based on the rewards they get. We propose a pure exploration phase where each player picks an arm uniformly at random, similar to the one suggested in [7]. Note that in contrast to [7], we do not assume that $T$ is known to the players. Hence, the exploration phase is repeated in each epoch. In each epoch, only a constant number $c_1$ of turns is dedicated to exploration. However, the estimation uses all the previous exploration phases, so that the number of samples for estimation grows linearly with time.

The estimation of the expected rewards is never perfect. Hence, the optimal solution to the assignment problem given the estimated expectations might be different from the optimal solution with the correct expectations. However, if the uncertainty of the true value of each expectation is small enough, we expect both of these optimal assignments to coincide. This is exactly the precision we require from the estimation, as formulated in the following lemma.

**Lemma 4.** *Assume that $\{\mu_{n,i}\}$ are known up to an uncertainty of $\Delta$, i.e., $|\hat{\mu}_{n,i} - \mu_{n,i}| \leq \Delta$ for each $n$ and $i$ for some $\{\hat{\mu}_{n,i}\}$. Denote the optimal assignment by $\boldsymbol{a_1} = \arg\max_{\boldsymbol{a}} \sum_{n=1}^{N} g_n(\boldsymbol{a})$ and its objective by $J_1 = \sum_{n=1}^{N} g_n(\boldsymbol{a_1})$. Denote the second best objective and the corresponding assignment by $J_2$ and $\boldsymbol{a_2}$, respectively. If $\Delta < \frac{J_1 - J_2}{2N}$ then*

$$\arg\max_{\boldsymbol{a}} \sum_{n=1}^{N} g_n(\boldsymbol{a}) = \arg\max_{\boldsymbol{a}} \sum_{n=1}^{N} \hat{\mu}_{n,a_n} \eta_{a_n}(\boldsymbol{a}) \tag{17}$$

*so that the optimal assignment does not change due to the uncertainty.*

*Proof:* First note that an optimal solution must not have any collisions, otherwise it can be improved since $K \geq N$. Hence $J_1 = \sum_{n=1}^{N} \mu_{n,a_{1,n}}$. For all $n$ and $i$ we have $\hat{\mu}_{n,i} = \mu_{n,i} + z_{n,i}$ such that $|z_{n,i}| \leq \Delta$. In the perturbed assignment problem, $\boldsymbol{a}_1$ performs at least as well as

$$\sum_{n=1}^{N} \hat{\mu}_{n,a_{1,n}} = \sum_{n=1}^{N} \left(\mu_{n,a_{1,n}} + z_{n,i}\right) \geq \sum_{n=1}^{N} \mu_{n,a_{1,n}} - \Delta N \tag{18}$$

and any assignment $\boldsymbol{a} \neq \boldsymbol{a}_1$ performs at most as well as

$$\sum_{n=1}^{N} \hat{\mu}_{n,a_n} \eta_{a_n}(\boldsymbol{a}) = \sum_{n=1}^{N} \left(\mu_{n,a_n} + z_{n,i}\right) \eta_{a_n}(\boldsymbol{a}) \leq \sum_{n=1}^{N} \mu_{n,a_{2,n}} \eta_{a_{2,n}}(\boldsymbol{a}_2) + \Delta N. \tag{19}$$

Hence it follows that if $\Delta < \frac{J_1 - J_2}{2N}$ then for every $\boldsymbol{a} \neq \boldsymbol{a}_1$

$$\sum_{n=1}^{N} \hat{\mu}_{n,a_{1,n}} > \sum_{n=1}^{N} \hat{\mu}_{n,a_n} \eta_{a_n}(\boldsymbol{a}). \tag{20}$$

■

If the exploration phase is long enough, players know their arm expectations accurately enough with a very small failure probability. The following lemma concludes this section by providing an upper bound for the probability that the estimation for epoch $k$ failed.

**Lemma 5** (Exploration Error Probability)**.** *Let $\left\{\mu_{n,i}^{k}\right\}$ be the estimated reward expectations using all the exploration phases up to epoch $k$. Denote $\boldsymbol{a}^* = \arg\max_{\boldsymbol{a}} \sum_{n=1}^{N} g_n(\boldsymbol{a})$ and $\boldsymbol{a}^{k*} = \arg\max_{\boldsymbol{a}} \sum_{n=1}^{N} \mu_{n,a_n}^{k} \eta_{a_n}(\boldsymbol{a})$. Also denote $J_1 = \sum_{n=1}^{N} g_n(\boldsymbol{a}^*)$ and the second best*[1] *objective by $J_2$. If the length of the exploration phase satisfies $c_1 > \frac{16N^2K}{(J_1 - J_2)^2}$, then after the $k$-th epoch we have*

$$P_{e,k} \triangleq \Pr\left(\boldsymbol{a}^* \neq \boldsymbol{a}^{k*}\right) \leq 4K^2 e^{-k}. \tag{21}$$

[1] Note that this is the second best objective and not the second best allocation, so $J_2 < J_1$. If all allocations have the same objective then this Lemma trivially holds with $c_1 \geq 1$.

*Proof:* According to [7, Lemma 1], for each $\Delta > 0$ and error probability $0 < P_{e,k} < 1$ , after

$$T_0 = \frac{4K}{\Delta^2} \ln\left(\frac{4K^2}{P_{e,k}}\right) \tag{22}$$

turns of pure exploration, for all $n$ and all $i$ we have with a probability of at least $1 - P_{e,k}$ that

$$\left|\mu_{n,i}^k - \mu_{n,i}\right| \leq \Delta. \tag{23}$$

In [7], the formulation of the lemma is slightly different and states the probability of an $2\Delta$-correct ranking. However, the proof follows by showing (23). Note that we used the fact that

$$\Delta < \frac{J_1 - J_2}{2N} \leq \frac{N - 0}{2N} = \frac{1}{2} < 1$$

so

$$T_0 = \max\left\{\frac{K}{2} \ln\left(\frac{2K^2}{P_{e,k}}\right), \frac{4K}{\Delta^2} \ln\left(\frac{4K^2}{P_{e,k}}\right)\right\} = \frac{4K}{\Delta^2} \ln\left(\frac{4K^2}{P_{e,k}}\right). \tag{24}$$

We conclude that after $T_0$ exploration turns, the error probability is at most $P_{e,k}$. Hence, if the exploration phase has a duration of at least $\frac{4K}{\Delta^2}$ turns we obtain

$$\frac{4K}{\Delta^2} k \leq c_1 k = \frac{4K}{\Delta^2} \ln\left(\frac{4K^2}{P_{e,k}}\right) \Longrightarrow P_{e,k} \leq 4K^2 e^{-k} \tag{25}$$

which together with the requirement $\Delta < \frac{J_1 - J_2}{2N}$ of Lemma 4 completes the proof. ■

## V. Game of Thrones Dynamics Phase

In this section we analyze the game of thrones (GoT) dynamics between players. These dynamics guarantee that the optimal state will be played a significant amount of time, and only require the players to know their own action and the received payoff on each turn. Note that these dynamics assume deterministic utilities. We use the estimated expected reward of each arm as the utility for this step, and zero if a collision occurred. This means that players ignore the numerical reward they receive by choosing the arm, as long as it is positive.

**Definition 6.** The game of thrones $G$ of epoch $k$ has the $N$ players of the original multi-armed bandit game. Each player can choose from among the $K$ arms, so $\mathcal{A}_n = \{1, ..., K\}$ for each $n$. The utility of player $n$ in the strategy profile $\boldsymbol{a} = (a_1, ..., a_N)$ is

$$u_n(\boldsymbol{a}) = \mu_{n,a_n}^k \eta_{a_n}(\boldsymbol{a}) \tag{26}$$

where $\mu_{n,a_n}^k$ is the estimation of the expected reward of arm $a_n$, from all the exploration phases that have ended, up to epoch $k$. Also denote $u_{n,\max} = \max\limits_{\boldsymbol{a}} u_n(\boldsymbol{a})$.

Our dynamics belong to the family introduced in [23]–[25]. These dynamics guarantee that the optimal sum of utilities strategy profiles will be played a sufficiently large portion of the turns. However, they all rely on the following structural property of the game, called interdependence.

**Definition 7.** A game $G$ with finite action spaces $\mathcal{A}_1, ..., \mathcal{A}_N$ is interdependent if for every strategy profile $\boldsymbol{a} \in \mathcal{A}_1 \times ... \times \mathcal{A}_N$ and every set of players $J \subset N$, there exists a player $n \notin J$ and a choice of actions $\boldsymbol{a}'_J \in \prod_{m \in J} \mathcal{A}_m$ such that $u_n(\boldsymbol{a}'_J, \boldsymbol{a}_{-J}) \neq u_n(\boldsymbol{a}_J, \boldsymbol{a}_{-J})$.

Our GoT is not interdependent. To see this, pick any strategy profile $\boldsymbol{a}$ such that some players are in a collision while others are not. Choose $J$ as the set of all players that are not in a collision. All players outside this set are in a collision, and there does not exist any colliding player that the actions of the non-colliding players can make her utility non-zero.

The GoT Dynamics in Algorithm 2 modify [24] such that interdependency is no longer needed. Note that in comparison with [24], our dynamics assign zero probability that a player with $u_n = 0$ (in a collision) will be content. Additionally, we do not need to keep the benchmark utility as part of the state. A player knows with probability 1 whether there was a collision, and if there was not, she gets the same utility for the same arm. Our dynamics require that each player uses $c \geq N$. The number of players $N$ might be unknown. In this case, players can use $c \geq K$, since the number of arms is known and $K \geq N$ by definition of the problem.

The GoT dynamics induce a Markov chain over the state space $Z = \prod_{n=1}^{N} (\mathcal{A}_n \times \mathcal{M})$, where $\mathcal{M} = \{C, D\}$. The transition matrix of this Markov chain is denoted by $P^{\varepsilon}$. The following lemma characterizes the recurrence classes of the unperturbed chain $P^0$ (with $\varepsilon = 0$). In [24], interdependency was used to prove the same result. This is the sole reason interdependency was required in the first place. We provide an alternative proof that does not require interdependency but instead uses the fact that in our modified dynamics, no player can be content with $u_n = 0$. Note that this proof exploits the structure of the GoT, and cannot be applied to a more general game.

**Lemma 8.** *Denote by $D_0$ the set of all the discontent states (all players are discontent) and by $C_0$ the set of all singleton content states (all players are content). The recurrence classes of the unperturbed process $P^0$ are $D_0$ and all $z \in C_0$.*

*Proof:* In $P^0$, there is no path between the discontent states and the content ones. Moreover, all the discontent states are connected and all the content states are absorbing (i.e., singletons). Now assume there is a different recurrence class. In any state in this class, denoted $z_{C/D}$, not all the players are content, otherwise this is a $z \in C_0$ singleton. Denote one of the discontent players by $n$. Since she chooses her action at random, there is a positive probability that she will pick the same arm as any of the content players. By doing so, she changes the state of this player to discontent with probability 1. With $\varepsilon = 0$, every discontent player remains so with probability one. On the next turn, a discontent player may again choose the arm of a content player with a positive probability. By repeating this process, we conclude that there is a positive probability that all players become discontent. Hence, $z_{C/D}$ is connected to $D$ in $P^0$. We conclude that this different recurrence class is in fact connected to $D$, which is a contradiction. ■

The process $Z$ of the GoT dynamics is a regular perturbed Markov chain, defined as follows.

**Definition 9.** $P^{\varepsilon}$ is called a regular perturbed Markov Process if $P^{\varepsilon}$ is ergodic for all sufficiently small $\varepsilon > 0$ and for every $z, z' \in Z$ we have

$$\lim_{\varepsilon \to 0^+} P^{\varepsilon}_{zz'} = P^0_{zz'} \tag{27}$$

and if $P^{\varepsilon}_{zz'} > 0$ for some $\varepsilon > 0$ then

$$0 < \lim_{\varepsilon \to 0^{+}} \frac{P^{\varepsilon}_{zz'}}{\varepsilon^{r(z \to z')}} < \infty \tag{28}$$

for some real non-negative $r\left(z \to z'\right)$ that is called the resistance of the transition $z \to z'$.

Next we define stochastic stability, which is a powerful convergence analysis tool.

**Definition 10.** Let $P^{\varepsilon}$ be regular perturbed Markov process and $\mu^{\varepsilon}$ its unique stationary distribution that exists for $\varepsilon > 0$. A state $z \in Z$ is stochastically stable if and only if

$$\lim_{\varepsilon \to 0^{+}} \mu^{\varepsilon}\left(z\right) > 0. \tag{29}$$

In [24], it is shown for their dynamics that only the states with the maximal sum of utilities are stochastically stable. For a small enough $\varepsilon$ the dynamics will visit the stochastically stable states very often. However, there might be several stochastically stable states and the dynamics might fluctuate between them. Fortunately, in our case, as shown in the following lemma, there is a unique optimal state with probability one. For a small enough $\varepsilon$ the unique optimal state is played more than half of the times, which allows for the players to distributedly agree on the optimal solution. This uniqueness is due to the continuous distribution of the rewards that makes the distribution of the empirical estimation for the expectations continuous as well.

**Lemma 11.** *The optimal solution to* $\max\limits_{\boldsymbol{a}} \sum_{n=1}^{N} u_n\left(\boldsymbol{a}\right)$ *is unique with probability 1.*

*Proof:* First note that an optimal solution must not have any collisions, otherwise it can be improved since $K \geq N$. Let $\left\{\mu^{k}_{n,i}\right\}$ be the estimated reward expectations in epoch $k$. For two different solutions $\tilde{\boldsymbol{a}} \neq \boldsymbol{a}^{*}$ to be optimal, we must have $\sum_{n=1}^{N} \mu^{k}_{n,\tilde{a}_n} = \sum_{n=1}^{N} \mu^{k}_{n,a^{*}_n}$. However, $\tilde{\boldsymbol{a}}$ and $\boldsymbol{a}^{*}$ must differ in at least one assignment. Since the distributions of the rewards $r_{n,a_n}$ are continuous, so are the distributions of $\sum_{n=1}^{N} \mu^{k}_{n,a_n}$ (as a sum of the average of the rewards). Hence $\Pr\left(\sum_{n=1}^{N} \mu^{k}_{n,\tilde{a}_n} = \sum_{n=1}^{N} \mu^{k}_{n,a^{*}_n}\right) = 0$, and the result follows. ■

Next we show that only the unique optimal state is stochastically stable. This means that after enough time, the action that a player played most of the time is highly likely to be part of the unique optimal solution. This is crucial for the success of the exploitation phase.

**Theorem 12.** *Define* $\boldsymbol{a}^{k*} = \arg\max\limits_{\boldsymbol{a}} \sum_{n=1}^{N} u_n\left(\boldsymbol{a}\right)$. *Under the GoT dynamics, the unique stochastically stable state is* $z^{*} = \left[\boldsymbol{a}^{k*}, C^{N}\right]$ *with probability 1.*

*Proof:* Let $z, z' \in Z$. Define for each $z$

$$\mathcal{N}_z = \left\{n \,|\, \overline{a}_n \neq a_n \,\text{or}\,\; u_n = 0 \,\text{or}\, M_n = D\right\}. \tag{30}$$

This is the set of players for which the transition of $M_n$ is governed by (11). Compared to [24], our dynamics have a different transition probability $P_{zz'}$ only when $z$ has a non-empty $\mathcal{N}_z$. For each $\mathcal{N}_z \subseteq \mathcal{N}$ we have

$$\lim_{\varepsilon \to 0^{+}} \frac{\prod_{n \in \mathcal{N}_z} \frac{u_n}{u_{n,\max}} \varepsilon^{u_{n,\max} - u_n}}{\varepsilon^{\sum_{n \in \mathcal{N}_z} \left(u_{n,\max} - u_n\right)}} = \prod_{n \in \mathcal{N}_z} \frac{u_n}{u_{n,\max}}. \tag{31}$$

Hence, in the limit $\varepsilon \to 0^{+}$, the ratio between the transition probabilities in our dynamics and those of [24] is either

$\prod_{n\in\mathcal{N}_z}\frac{u_n}{u_{n,\max}}\leq 1$ or one. We conclude that each transition either has the same resistance as in [24] or it is impossible since $u_n=0$ for some $n$. From any $z\in C_0$ there is a path with resistance $c$ to $D$, where a content player explores and becomes discontent. From any $z\in D$ to any $z\in C_0$ there is a path where all discontent players become content, which has resistance $\sum_n\left(u_{n,\max}-u_n\left(z\right)\right)$ with $\left\{u_n\left(z\right)\right\}$ as the utilities in $z$. Therefore, the path from any $z$ to the maximizers of $\sum_{n\in\mathcal{N}}u_n$ (which are in $C_0$) has the same resistance as in [24] (it is the same path). Since all other paths have the same resistance as in [24] or do not exist, the maximizers of $\sum_{n\in\mathcal{N}}u_n$ remain the only stochastically stable states, as in [24]. From Lemma 11 we know that the maximizer of $\sum_{n\in\mathcal{N}}u_n$ is unique with probability 1. ■

Now we can prove the main lemma of this section that gives an upper bound for the probability that the GoT phase does not lead to the optimal solution.

**Lemma 13** (GoT Error Probability)**.** *Let $\delta>0$. Define $\boldsymbol{a}^{k*}=\arg\max_{\boldsymbol{a}}\sum_{n=1}^{N}u_n\left(\boldsymbol{a}\right)$ and $\widetilde{\boldsymbol{a}}=\left(\widetilde{a}_1,...,\widetilde{a}_N\right)$ where $\widetilde{a}_n=\arg\max_i F_t^n\left(i\right)$ for all $n$. For a small enough $\varepsilon$, the error probability of the $k$-th GoT phase, which is the probability that another strategy profile than $\boldsymbol{a}^{k*}$ will be played in the exploitation phase, is bounded as follows*

$$P_{c,k}\triangleq\Pr\left(\widetilde{\boldsymbol{a}}\neq\boldsymbol{a}^{k*}\right)\leq A_0e^{-\frac{c_2(1-\rho)}{1728T_m\left(\frac{1}{8}\right)}k^{1+\delta}}\tag{32}$$

*where $A_0$ is a constant with respect to $t$ (or $k$), and may depend on $N,K,\varepsilon$ and the initial state.*

*Proof:* Denote the optimal state by $z^*=\left[\boldsymbol{a}^{k*},C^N\right]$. From Theorem 12 and the definition of a stochastically stable state, we know that for a small enough $\varepsilon$ we have $\pi\left(z^*\right)\geq\frac{2}{3}$. Denote the length of the part of the GoT phase where counting (of (6)) took place by $L\triangleq\left\lfloor c_2\left(1-\rho\right)k^{1+\delta}\right\rfloor$. In the end of the GoT phase, each player picks the action that she played most of the times she was content. If the strategy profile $\boldsymbol{a}^{k*}$ was played more than $\frac{L}{2}$ of the time, each player played the corresponding action at least half of the time. Hence, the probability that a strategy profile other than $\boldsymbol{a}^{k*}$ would be picked is lower than the probability that $\boldsymbol{a}^{k*}$ has been played less than $\frac{L}{2}$ of the time. We bound this probability using [27, Theorem 3]. Our function is $f\left(z\right)=I\left(z=z^*\right)$, that counts the number of visits to the optimal state. Note that the events $I\left(z\left(t\right)=z^*\right)$ are not independent but rather form a Markov chain. Hence, Markovian concentration inequalities are required.

We denote by $T_m\left(\frac{1}{8}\right)$ the mixing time of $Z$ with an accuracy of $\frac{1}{8}$. We define for the initial distribution $\varphi$ on $Z$ (after $d_g=\left\lceil\rho c_2k^{1+\delta}\right\rceil$ turns in the $k$-th GoT Phase),

$$\left\|\varphi\right\|_\pi\triangleq\sqrt{\sum_{i=1}^{|Z|}\frac{\varphi_i^2}{\pi_i}}.\tag{33}$$

By choosing $\eta=1-\frac{1}{2\pi(z^*)}$ (so $0<\eta<1$ when $\pi\left(z^*\right)>\frac{1}{2}$), we obtain the following bound for a small enough $\varepsilon$

$$\Pr\left(\sum_{\tau=1}^{L}f\left(z\left(\tau\right)\right)\leq\left(1-\eta\right)\pi\left(z^*\right)L\right)\leq\Pr\left(\sum_{\tau=1}^{L}I\left(z\left(\tau\right)=z^*\right)\leq\frac{c_2\left(1-\rho\right)k^{1+\delta}}{2}\right)\leq$$
$$c\left\|\varphi\right\|_\pi e^{-\frac{\left(1-\frac{1}{2\pi(z^*)}\right)^2\pi(z^*)c_2(1-\rho)}{72T_m\left(\frac{1}{8}\right)}k^{1+\delta}}\underset{(a)}{\leq}c\left\|\varphi\right\|_\pi e^{-\frac{c_2(1-\rho)}{1728T_m\left(\frac{1}{8}\right)}k^{1+\delta}}\tag{34}$$

where $c$ is some constant. Note that $\pi\left(1-\frac{1}{2\pi}\right)^2=\pi-1+\frac{1}{4\pi}$ is monotonically increasing for all $\frac{1}{2}<\pi<1$. Since for a small enough $\varepsilon$ we have $\pi\left(z^*\right)\geq\frac{2}{3}$, (a) follows by substituting $\pi\left(z^*\right)=\frac{2}{3}$.

There is a tradeoff regarding $\|\varphi\|_\pi$. Starting from an arbitrary initial condition, $\|\varphi\|_\pi$ can be large. By dedicating the first $d_g$ turns of the GoT phase to letting $Z$ approach its stationary distribution, and starting to count the visits to $z^*$ only afterwards, we can reduce $\|\varphi\|_\pi$ significantly, at the cost of $d_g$ turns less for estimating $z^*$. Optimizing over $d_g$ (or $\rho$) and $\|\varphi\|_\pi$ can improve the constants of the bound (34). ■

## VI. Numerical Simulations and Practical Considerations

The total regret compares the sum of utilities to the ideal one that could have been achieved in a centralized scenario. With no communication between players and with a matrix of expected rewards, the gap to this ideal naturally increases. In this scenario, converging to the exact optimal solution might take a long time, even for the (unknown) optimal algorithm. Our main result provides theoretical guarantees for the asymptotic performance of our algorithm, which suggest that performance improves with time on its way to converge to the optimal solution. The simulations in this section complete the picture by showing how the sum of utilities behaves in the non-asymptotic regime.

We simulated a multi-armed bandit game with $\{\mu_{n,i}\}$ that are chosen independently and uniformly at random in $[0.05, 0.95]$. 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 $n, i$.

In the simulations presented here we use $\delta = 0$ since it yields good results in practice. We conjecture that the bound (32) is not tight for our particular Markov chain and indicator function, since it applies for all Markov chains with the same mixing time and all functions on the states. This explains why modest choices of $c_2$ are large enough and the $k^\delta$ factor in the exponent is not needed in practice. The lengths of the phases should be chosen so that the exploitation phase occupies most of the turns already in early epochs, while allowing for a considerable GoT phase. Note that the exploration ($c_1$) is much easier than the GoT phase ($c_2$) and achieves a good accuracy relatively fast. Hence we choose $c_1 = 1000, c_2 = c_3 = 6000$. We use $\rho = \frac{1}{2}$ in the simulations we present, since the performance is very similar for $\rho$ values not too close to zero or one. We use $c = N$, that gives the highest possible escape probability of $\varepsilon^c$ from a content state.

In Fig. 2, we present the sample mean of the accumulated sum of utilities $\sum_{n=1}^{N} \frac{1}{t} \sum_{\tau=1}^{t} u_n(\boldsymbol{a}(\tau))$ as a function of time $t$, averaged over 100 experiments. The performance was normalized by the optimal solution to the assignment problem (for each experiment). On the left graph we compare our sum of utilities for $N = K = 5$ to that of the selfish algorithm, reported to achieve good performance for this problem in [17], and to a random choice of arms. The selfish algorithm consists of each player playing a standard upper confidence bound (UCB) algorithm, treating collisions as any other value for the reward. Both algorithms perform much better than the random selection. Our sum of utilities is slightly better and is increasing with time. More importantly, our algorithm has provably performance guarantees while [17] have none. On its way to converge to the optimal solution, our algorithm performs very well straight from the beginning. While visiting near-optimal solutions inflicts linear regret at the beginning, it is very satisfying in practice considering that players cannot communicate and have a matrix of expected rewards. Similar results were obtained for different choices of $c_1, c_2, c_3$.

In Fig. 3, we present the median and the best 90% of the sample mean of the sum of utilities for $K = N = 6$ and $\varepsilon = 0.01, 0.001, 0.0001$. It is evident that our algorithm behaves very similarly in all the 100 experiments, indicating that it is robust and rarely fails. Additionally, our algorithm behaves very similarly for a wide range of $\varepsilon$ values (two orders of magnitude). This supports the intuition that there is no threshold phenomenon on $\varepsilon$ (becoming “small enough”), since the dynamics prefer states with a higher sum of utilities for all $\varepsilon < 1$.

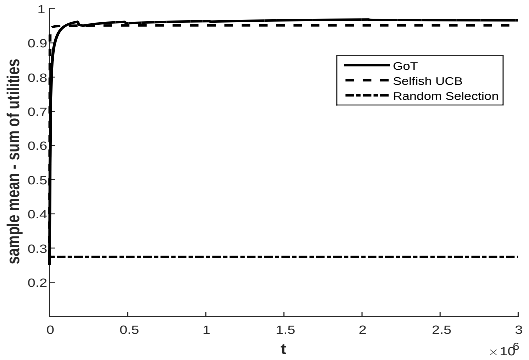

Fig. 2. Sample mean of the sum of utilities as a function of time, averaged over 100 experiments.

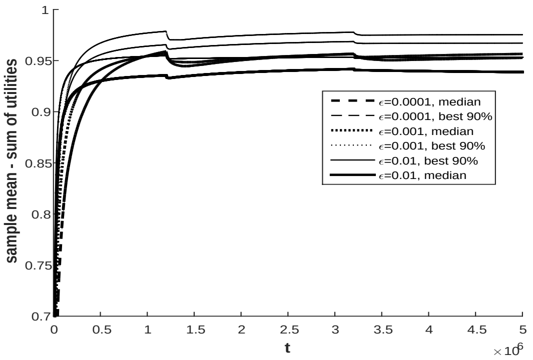

Fig. 3. Statistics of the sample mean of the sum of utilities as a function of time, over 100 experiments.

## VII. Conclusions and Future Work

In this paper, we considered a multi-player multi-armed bandit game where players compete over the arms as resources. In contrast to all existing multi-player bandit problems, we allow for different arm expected rewards between players **and** assume each player only knows her own actions and rewards. We proposed a novel fully distributed algorithm that achieves a poly-logarithmic expected total regret of near-$O\left(\log^2 T\right)$ when the horizon $T$ is unknown to the players.

Our simulations suggest that tuning the parameters for our algorithm is a relatively easy task in practice. The algorithm designer can do so by simulating a random model for the unknown environment and varying the parameters, knowing that only a very slack accuracy is needed for the tuning.

It is still an open question whether the lower bound $\Omega\left(\log T\right)$ on the expected total regret is tight for a fully distributed algorithm.

Our game is not a general one but has a structure that allowed us to modify the dynamics such that the interdependence assumption can be dropped. We conjecture that the same structure can be exploited to accelerate the convergence rate of the GoT dynamics, specifically by relaxing the $c \geq N$ condition.

# References

[1] S. Shahrampour, A. Rakhlin, and A. Jadbabaie, “Multi-armed bandits in multi-agent networks,” in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, 2017, pp. 2786–2790.

[2] E. Hillel, Z. S. Karnin, T. Koren, R. Lempel, and O. Somekh, “Distributed exploration in multi-armed bandits,” in *Advances in Neural Information Processing Systems*, 2013, pp. 854–862.

[3] P. Landgren, V. Srivastava, and N. E. Leonard, “Distributed cooperative decision-making in multiarmed bandits: Frequentist and Bayesian algorithms,” in *Decision and Control (CDC), 2016 IEEE 55th Conference on*, 2016, pp. 167–172.

[4] N. Cesa-Bianchi, C. Gentile, Y. Mansour, and A. Minora, “Delay and cooperation in nonstochastic bandits,” in *Conference on Learning Theory*, 2016, pp. 605–622.

[5] B. Szorenyi, R. Busa-Fekete, I. Hegedus, R. Ormándi, M. Jelasity, and B. Kégl, “Gossip-based distributed stochastic bandit algorithms,” in *International Conference on Machine Learning*, 2013, pp. 19–27.

[6] N. Korda, B. Szörényi, and L. Shuai, “Distributed clustering of linear bandits in peer to peer networks,” in *International Conference on Machine Learning*, vol. 48, 2016, pp. 1301–1309.

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

[8] N. Nayyar, D. Kalathil, and R. Jain, “On regret-optimal learning in decentralized multi-player multi-armed bandits,” *IEEE Transactions on Control of Network Systems*, vol. PP, no. 99, pp. 1–1, 2016.

[9] D. Kalathil, N. Nayyar, and R. Jain, “Decentralized learning for multiplayer multiarmed bandits,” *IEEE Transactions on Information Theory*, vol. 60, no. 4, pp. 2331–2345, 2014.

[10] K. Liu and Q. Zhao, “Distributed learning in multi-armed bandit with multiple players,” *IEEE Transactions on Signal Processing*, vol. 58, no. 11, pp. 5667–5681, 2010.

[11] S. Vakili, K. Liu, and Q. Zhao, “Deterministic sequencing of exploration and exploitation for multi-armed bandit problems,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 5, pp. 759–767, 2013.

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

[13] A. Anandkumar, N. Michael, A. K. Tang, and A. Swami, “Distributed algorithms for learning and cognitive medium access with logarithmic regret,” *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 731–745, 2011.

[14] H. Liu, K. Liu, and Q. Zhao, “Learning in a changing world: Restless multiarmed bandit with unknown dynamics,” *IEEE Transactions on Information Theory*, vol. 59, no. 3, pp. 1902–1916, 2013.

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

[16] N. Evirgen and A. Kose, “The effect of communication on noncooperative multiplayer multi-armed bandit problems,” in *arXiv preprint arXiv:1711.01628, 2017*, 2017.

[17] L. Besson and E. Kaufmann, “Multi-player bandits revisited,” in *Algorithmic Learning Theory*, 2018, pp. 56–92.

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

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

[20] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1998.

[21] D. P. Bertsekas, “The auction algorithm: A distributed relaxation method for the assignment problem,” *Annals of operations research*, vol. 14, no. 1, pp. 105–123, 1988.

[22] M. M. Zavlanos, L. Spesivtsev, and G. J. Pappas, “A distributed auction algorithm for the assignment problem,” in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, 2008, pp. 1212–1217.

[23] B. S. Pradelski and H. P. Young, “Learning efficient Nash equilibria in distributed systems,” *Games and Economic behavior*, vol. 75, no. 2, pp. 882–897, 2012.

[24] J. R. Marden, H. P. Young, and L. Y. Pao, "Achieving pareto optimality through distributed learning," *SIAM Journal on Control and Optimization*, vol. 52, no. 5, pp. 2753–2770, 2014.

[25] A. Menon and J. S. Baras, "Convergence guarantees for a decentralized algorithm achieving pareto optimality," in *American Control Conference (ACC), 2013*, 2013.

[26] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in applied mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[27] K.-M. Chung, H. Lam, Z. Liu, and M. Mitzenmacher, "Chernoff-Hoeffding bounds for Markov chains: Generalized and simplified," in *29th International Symposium on Theoretical Aspects of Computer Science*, 2012, p. 124.