Decentralized Dynamic Rate and Channel Selection Over a Shared Spectrum

Alireza Javanmardi, Muhammad Anjum Qureshi, *Member, IEEE*, and Cem Tekin, *Senior Member, IEEE*

***Abstract*— We consider the problem of distributed dynamic rate and channel selection in a multi-user network, in which each user selects a wireless channel and a modulation and coding scheme (corresponds to a transmission rate) in order to maximize the network throughput. We assume that the users are cooperative, however, there is no coordination and communication among them, and the number of users in the system is unknown. We formulate this problem as a multi-player multi-armed bandit problem and propose a decentralized learning algorithm that performs almost optimal exploration of the transmission rates to learn fast. We prove that the regret of our learning algorithm with respect to the optimal allocation increases logarithmically over rounds with a leading term that is logarithmic in the number of transmission rates. Finally, we compare the performance of our learning algorithm with the state-of-the-art via simulations and show that it substantially improves the throughput and minimizes the number of collisions.**

***Index Terms*— Cognitive radio, decentralized algorithms, multi-armed bandits, regret bounds.**

I. Introduction

THE radio spectrum is becoming deficient due to the surge of advanced wireless devices and the typical licensing of frequency bands to primary users (PUs) by the Federal Communications Commission (FCC). As interfering with PUs has undesirable consequences, secondary users (SUs) are either enforced to implement sophisticated channel sensing algorithms or confine their transmission to unused frequency bands. In this paper, we consider the latter case as SUs are able to use a geolocation database to get a list of channels free from PUs [1]. The quality of these channels is time-varying and heavily depends on the chosen modulation and coding scheme (MCS) (or equivalently on the chosen rate). On the one hand, SUs have no prior knowledge about the quality of the (channel, rate) pairs. On the other hand, choosing the best (channel, rate) pair can significantly enhance the performance. Therefore, SUs need to adapt to the channel conditions and learn the optimal transmission parameters through repeated interaction with the environment. To achieve the aforementioned goal, a SU either explores (channel, rate) pairs to refurbish its belief about them or exploits the information collected so far. This typical exploration-exploitation trade-off in sequential decision making problems has been studied in the literature on multi-armed bandits (MAB) [1]–[3].

MAB problem is a type of sequential optimization problem, where in each round, a decision-maker pulls an arm enforced by a specific policy and receives a random reward whose distribution is not known beforehand [4], [5]. The objective of the decision-maker is to maximize the cumulative reward. In the aforementioned scenario, (channel, rate) pairs are considered as *arms* and SUs as *players* or *learners*. In each round, each player selects a (channel, rate) pair for packet transmission, and then, receives a random reward which indicates that either the transmission is successful or unsuccessful. In this scenario, the expected reward of each (channel, rate) pair is the expected number of successfully transmitted bits in a round (throughput) and is proportional to the chosen rate times the packet successful transmission probability.

We consider a multi-user network, where the channels are utilized by several SUs. We rigorously formulate this resource allocation problem as a multi-player multi-armed bandit (MPMAB) problem, a variant of the standard MAB problem, in which the players' goal is to maximize the sum of their rewards [6]–[9]. When the throughput of each (channel, rate) pair for each SU is known by a central entity (practically not feasible), then the optimal assignment, i.e., the (channel, rate) pair assigned to each SU, can be computed offline. We call the difference between the cumulative reward of the optimal assignment (summed over all SUs) and the cumulative reward of the learning algorithm (summed over all SUs) as the regret. The regret measures the loss in performance due to decentralization and not knowing the throughputs beforehand.

We list the challenges faced in the above multi-user resource allocation problem as follows: Firstly since the locations of the transmitter-receiver pairs are different for different SUs, each SU experiences a different gain on a given channel. This implies that the quality of each channel, and consequently the expected reward of each (channel, rate) pair is different for different users. Similarly, variations in the channel gains and the noise levels introduce intra-user variability to the expected rewards of different channels. Secondly, as the number of (channel, rate) pairs increases, learning becomes more challenging since there are more options to explore. Lastly, as the

Manuscript received March 1, 2021; accepted March 4, 2021. Date of publication March 15, 2021; date of current version June 16, 2021. This work was supported by the Scientific and Technological Research Council of Turkey (TUBITAK) under Grant 116E229. The associate editor coordinating the review of this article and approving it for publication was S. M. Perlaza. *(Corresponding author: Muhammad Anjum Qureshi.)*

The authors are with the Department of Electrical and Electronics Engineering, Bilkent University, 06800 Ankara, Turkey (e-mail: javanmardi@ee.bilkent.edu.tr; qureshi@ee.bilkent.edu.tr; cemtekin@ee.bilkent.edu.tr).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCOMM.2021.3066002.

Digital Object Identifier 10.1109/TCOMM.2021.3066002

users act without any coordination, multiple SUs may choose the same channel simultaneously. In this case, we say that a collision occurs and all the colliding SUs get zero rewards. A high number of collisions can slow down the learning process and cause the wastage of resources.

In addition to the challenges mentioned above, even in the single-user setting, identifying the optimal rates for a given channel by naively exploring all rates sufficiently enough results in a sample complexity that scales linearly in the number of rates, while sequential elimination of the suboptimal rates results in a sample complexity that is near logarithmic in the number of rates [10]. Our proposed learning method generalizes this idea to regret minimization in the decentralized multi-user case by using the sequential halving of the transmission rates to learn faster for all users, resulting in a fewer number of collisions and significantly higher throughput than the state-of-the-art.

Our main contributions are summarized as follows:

- We design a distributed learning algorithm for channel and rate assignment in a heterogeneous multi-user network. The proposed algorithm employs a *sequential halving orthogonal exploration* phase to keep the number of collisions between users and the number of rate explorations at minimum.
- We prove that our algorithm achieves $O(\log(T))$ regret with respect to the oracle expected reward maximizing network throughput.
- We provide experimental results that show the superiority of our algorithm over the state-of-the-art decentralized learning algorithms.

The rest of the paper is organized as follows. Related work is given in Section II. The MPMAB problem is formulated in Section III. The learning algorithm is proposed in Section IV, and its analysis is given in Section V. In Section VI, we integrate the proposed learning algorithm into an existing state-of-the-art MPMAB algorithm. Numerical results for the proposed scheme are provided in Section VII, followed by concluding remarks in Section VIII.

II. Related Work

In the standard stochastic MAB problem, the learner sequentially plays one arm at a time over multiple rounds and observes at the end of each round the random reward of the arm chosen in that round. The goal of the learner is to maximize its cumulative reward without knowing the arm reward distributions beforehand by only learning through the past reward observations. This goal is achieved by balancing exploration and exploitation of the arm set [4], [5]. MPMAB problem is an extension to the stochastic MAB problem, where multiple users select arms simultaneously in each round in order to maximize the sum of the rewards. MPMAB problem is frequently used to model multi-user cognitive communications, where the SUs learn to select network resources that maximize the sum throughput [6]–[9], [11], [12].

In the centralized MPMAB problem, a central learner selects the arms of the players. For instance, in [13], a centralized policy is built based on the combinatorial MAB framework for multi-user channel allocation in a cognitive radio network (CRN) and is shown to achieve logarithmic in time regret. A decentralized MPMAB model, where the players need to agree on a time division before the channel selection process starts, is introduced in [7]. The case where the expected reward of an arm is different for different players is studied in [14] and [15]. In these works, an algorithm that achieves polylogarithmic in time regret is developed for the case when the players communicate their arm selection with each other. A distributed MPMAB model in which players learn their optimal arms by observing the global reward of their joint arm selection is considered in [16]. In this work, it is assumed that the players exchange messages in order to coordinate their exploration and exploitation phases. The players are able to achieve logarithmic in time regret by using a distributed algorithm that alternates between exploration and exploitation. While [7], [14], [15] do not allow multiple players to use the same arm (channel), [17] and [18] consider the case when multiple players can obtain non-zero rewards from the same arm. Authors in [17] use the idea of synchronized explorations in order to achieve logarithmic in time regret.

MPMAB problem can be further categorized into two groups based on the structure of the rewards:

1) Homogeneous MPMAB problem: In this setting, the expected reward of an arm is the same for all the players. A decentralized MPMAB problem in which the number of players in the network is unknown is considered in [11] and [12]. Both works propose decentralized algorithms that employ an explore-then-exploit approach. When the time horizon is not known in advance, this approach is employed by using a doubling trick. In particular, [11] proposes Musical Chairs (MC) algorithm that estimates the number of players in the system based on the number of collisions and learns the channel rewards by random explorations. One disadvantage of MC is that it results in an excessive number of collisions during the exploration phase. In [12], exploration is done by employing an orthogonalization technique that orthogonalizes the players with respect to the channels in order to have collision-free transmissions even during the exploration phase. Algorithms in [11] and [12] achieve expected regret that grows logarithmically over time.

   Departing from the works above, [19] assumes that the number of players is known beforehand, and proposes MCTopM algorithm, which instead of separating exploration and exploitation, uses *upper confidence bound* (UCB) indices to select arms for each player. While all of the works mentioned above assume that collisions are observed by all players that experience them, [20] considers the case when the players are not provided collision feedback and propose an algorithm that achieves logarithmic in time regret for this case. It is shown in [21] that regret with optimal dependence on the gap between the best and the second best allocation can be achieved by utilizing synchronization across players. In [22], a decentralized algorithm is proposed which uses an accelerated consensus procedure and an adaptation

TABLE I
COMPARISON OF OUR WORK WITH PRIOR WORKS

Property/ Algorithm	Musical Chairs [11]	Trekking Approach [12]	GoT [9]	**GoT-SHOE** (our work)
Expected regret (given T)	$O(\log T)$	$O(\log T)$	$O(\log T)$	$O(\log T)$
No. of users	Unknown	Unknown	Unknown	Unknown
Heterogeneous	×	×	✓	✓
No. of collisions	High	Low	High	Low
Collision feedback	✓	✓	✓	✓
Rate selection	×	×	×	✓
Reward structure	Any distribution on [0,1]	Any distribution on [0,1]	Cont. distribution on [0,1]	Discrete (Bernoulli)

of the UCB algorithm to compute estimates of the average rewards and cater to the delay and error of the estimates. In [23], decentralized algorithms that utilize a leader-follower approach for communication among players are proposed. These algorithms are investigated under MPMAB models with collision (players who select the same arm get zero rewards) and no collision.

2) Heterogeneous MPMAB problem: In this setting, the expected reward of an arm is different for different players. A fully-distributed algorithm, known as *Game of Thrones* (GoT), is proposed in [6] and enhanced in [9]. This algorithm solves a special case of the well-known distributed assignment problem [24] by using collision and reward feedback. An improved algorithm with a better convergence time than GoT is proposed in [25], [26]. However, this algorithm works under a more restrictive assumption: it requires that the quality of each (channel, rate) pair is an integer multiple of a common resolution $\Delta_{\min}$ which is known to the SUs. The algorithms proposed in [6] and [25] are able to achieve a near-$O(\log T)$ expected regret using the doubling trick and $O(\log T)$ expected regret in the one-shot scenario (when T is known). Authors in [27] provide a distributed algorithm that aims to achieve optimal network throughput without any direct communication among players. However, the algorithm has a binary signalling phase that allows players to exchange information by transmitting in specific patterns, and players are also allowed to sense such transmissions from other players. In [28], an efficient MPMAB algorithm is proposed, which exploits the idea of forced collisions and matching elimination to attain logarithmic regret. In addition to the works mentioned above, a non-stochastic version of MPMAB, where the losses incurred by the arms arbitrarily change over time, is studied in [29] and [30].

Apart from MPMAB problem, many works have proposed almost optimal pure exploration algorithms for the single-player best arm identification problem given a fixed budget or fixed confidence (see, e.g., [10], [31]–[33]). Specifically, authors in [10] propose a set of algorithms achieving an upper bound for the number of arm pulls whose gap from the lower bound is only doubly-logarithmic in the problem parameters. These algorithms are mainly built upon the idea of sequential elimination. Within an episode, arms are sampled uniformly, and at the end of each episode, arms are eliminated according to a data-dependent elimination rule. This process continues until a single arm remains, and thus, at the end of the game, the player must choose the best arm, whether with specified confidence or within a specified time horizon.

Departing from the other works, our algorithm allocates channels to players and adapts the rate for the selected channels in a heterogeneous network, where the expected rewards of the (channel, rate) pairs are different for different players. The proposed algorithm learns channels and optimal rates together while keeping the number of collisions as low as possible by using *sequential halving orthogonal exploration* (SHOE), which extends the orthogonalization method proposed in [12]. Moreover, as our reward signal is binary, we only require 1-bit feedback (ACK/NACK), which reduces the overhead in communication applications compared to the case with continuous rewards, which requires multi-bit feedback. In addition, the feedback model we consider is extensively used in MAB-based communication papers [1], [3]. The differences between our work and the related works are summarized in Table I.

III. PROBLEM FORMULATION

A. Dynamic Rate and Channel Selection Problem

Consider N users (SUs) indexed by the set $\mathcal{N} := [N]$ and T rounds (time slots) of fixed and equal duration indexed by $t \in [T]$.[1] As in prior work [11], [12], we assume that the users are synchronized with respect to these rounds. In each round t, each user selects one of the K available channels indexed by the set $\mathcal{K} := [K]$ and a MCS from a finite set of MCSs, in which each MCS is associated with a unique transmission rate from the set $\mathcal{R} := \{\gamma_1, \ldots, \gamma_R\}$. We assume that $\mathcal{R}$ is ordered, i.e., $\gamma_1 < \ldots < \gamma_R$. The strategy set of each user consists of $K \times R$ (channel, rate) pairs. Similar to [9], [11], [12], we focus on the case where $K \geq N$ in the remainder of this section.[2] An example of channel allocation problem with $N = 5$ and $K = 5$ is provided in Fig. 1.

Let $c_n(t)$ represent the channel and $\gamma_n(t)$ represent the rate selected by user n in round t. We call the tuple $a_n(t) = (c_n(t), \gamma_n(t))$ the (channel, rate) pair (arm) selected by user n in round t. Let $\mathbf{a}(t) := [a_n(t)]_{n\in\mathcal{N}}$ represent the strategy profile in round t and let $\mathcal{A}$ represent the set of all possible strategy profiles. Users do not know N and the arms chosen by the other users. There is no communication among users and each user utilizes its own knowledge and history to select its (channel, rate) pair.

[1]For a positive integer N, $[N] := \{1, \ldots, N\}$.
[2]The case where $N > K$ is discussed in Section V-E

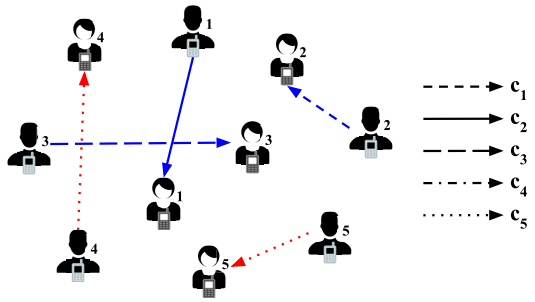


Fig. 1. The system model of a network with 5 users ($N = 5$) and 5 channels ($K = 5$). Different channels are represented by different dash types while their colors determine whether they are collision-free (blue) or not (red). Here, users 1, 2, and 3 transmit over channels 2, 1, and 3 respectively. Users 4 and 5 face collision on channel 5 while channel 4 is unused.

If two or more users select the same channel in the same round all of them get zero rewards and we say that a collision occurs on that channel. We assume that all users can identify whether the current round resulted in a collision or not on their channel. We define the no-collision indicator of channel i in the strategy profile $\mathbf{a}$ as:

$$\eta_i(\mathbf{a}) = \begin{cases} 0 & |\,\mathcal{N}_i(\mathbf{a})\,| > 1 \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

where $\mathcal{N}_i(\mathbf{a}) := \{n : c_n = i\}$ is the set of users who select channel i in strategy profile $\mathbf{a}$. For user n and her action a_n, let Bernoulli random variable $X_{n,a_n}(t)$ represent the transmission success ($X_{n,a_n}(t) = 1$) or failure ($X_{n,a_n}(t) = 0$) when user n transmits as the sole user on the channel specified in a_n. For $a_n = (c_n, \gamma_n)$, $r_{n,a_n}(t) = \frac{\gamma_n}{\gamma_R} X_{n,a_n}(t)$ represents the random reward that user n gets when it transmits with rate γ_n on channel c_n as the sole user on that channel. This indicates that the number of bits which has been successfully received by receiver n in round t is γ_n if $X_{n,a_n}(t) = 1$ and 0 otherwise. We assume that $\{X_{n,a_n}(t)\}_{t\in[T]}$ forms an i.i.d. sequence with a positive mean $\theta_{n,a_n} := \mathbb{E}[X_{n,a_n}(t)]$. As a result, the sequence $\{r_{n,a_n}(t)\}_{t\in[T]}$ is i.i.d. with expected reward $\mu_{n,a_n} := \mathbb{E}[r_{n,a_n}(t)] = \frac{\gamma_n}{\gamma_R}\theta_{n,a_n}$. Based on these, the reward obtained by user n in round t is given as:

$$v_n(\mathbf{a}(t)) := r_{n,a_n(t)}(t)\eta_{c_n(t)}(\mathbf{a}(t)). \tag{2}$$

We assume that each transmitter receives an ACK/NAK feedback over an error-free channel that determines whether a packet transmission has been successful or not. When there is a collision on the chosen channel, the transmitter also receives a collision feedback. Thus, user n observes that $\eta_{c_n(t)}(\mathbf{a}(t)) = 0$ when there is a collision and that $\eta_{c_n(t)}(\mathbf{a}(t)) = 1$ and whether $X_{n,a_n}(t)$ is 0 or 1 when there is no collision. Let

$$\gamma^*_{n,c} := \operatorname*{argmax}_{\gamma\in\mathcal{R}} \mu_{n,(c,\gamma)} \tag{3}$$

represent the unique best rate for user n on channel c and $m^*_{n,c}$ represent its index, i.e., $\gamma^*_{n,c} = \gamma_{m^*_{n,c}}$. When the user that we refer to is clear from the context, with an abuse of notation we let $\gamma^*_c = \gamma^*_{n,c}$ represent the optimal rate on channel c for user n. Similar to [10], we define

$$H_{n,c} := \max_{\gamma\neq\gamma^*_{n,c}} \frac{I_\gamma}{(\mu_{n,(c,\gamma^*_{n,c})} - \mu_{n,(c,\gamma)})^2} \tag{4}$$

where I_γ is the rank of rate γ in the list where rates are ordered by their expected throughputs (e.g., $I_{\gamma^*_{n,c}} = 1$). Also let $H_{\max} = \max_{n,c} H_{n,c}$. Note that $H_{n,c}$ is large when it is difficult to distinguish an optimal rate from a suboptimal rate. Thus, when $H_{\max}$ is large, it is difficult to learn the optimal strategy.

B. Definition of the Regret

Let $\gamma^*_n(t) := \gamma^*_{n,c_n(t)}$ and $\tilde{a}_n(t) := (c_n(t), \gamma^*_n(t))$. Subsequently, define $\tilde{\mathbf{a}}(t) := [\tilde{a}_n(t)]_{n\in\mathcal{N}}$ as the strategy profile where all the users select the best rate for their chosen channels. The expected reward of user n in strategy profile $\mathbf{a}$ is denoted by $g_n(\mathbf{a}) := \mathbb{E}[v_n(\mathbf{a})]$, where $v_n(\mathbf{a})$ is defined in (2). The (pseudo) regret over period T is defined as:

$$\mathrm{Reg}(T) := \sum_{t=1}^{T}\sum_{n=1}^{N} g_n(\mathbf{a}^*) - \sum_{t=1}^{T}\sum_{n=1}^{N} g_n(\mathbf{a}(t)) \tag{5}$$

where

$$\mathbf{a}^* := \operatorname*{argmax}_{\mathbf{a}\in\mathcal{A}} \sum_{n=1}^{N} g_n(\mathbf{a}). \tag{6}$$

We assume that $\mathbf{a}^*$ is unique. It is obvious that the optimal solution is an orthogonal allocation of the users in $\mathbf{a}^*$. The expected regret is given as $\overline{\mathrm{Reg}}(T) := \mathbb{E}[\mathrm{Reg}(T)]$.

Let $\tilde{\mathcal{A}} \subset \mathcal{A}$ be the subset in which the best rates are selected for the chosen channel of every user. Note that $|\mathcal{A}| = (RK)^N$ while $|\tilde{\mathcal{A}}| = K^N$. It is proved in the following lemma that the optimal strategy profile is always from the set $\tilde{\mathcal{A}}$.[3]

Lemma 1: The expected sum of the rewards of any strategy profile $\mathbf{a} = [(c_n, \gamma_n)]_{n\in[N]} \in \mathcal{A}$, is always less than or equal to the expected sum of the rewards of the strategy profile $\tilde{\mathbf{a}} = [(c_n, \gamma^*_{c_n})]_{n\in[N]} \in \tilde{\mathcal{A}}$, where the best rates are selected for the same channel allocation strategy.

Proof: Since, $\gamma^*_{c_n}$ is the true optimal rate for user n with channel c_n, we have

$$\mu_{n,(c_n,\gamma)} \leq \mu_{n,(c_n,\gamma^*_{c_n})}, \quad \forall\gamma \in \mathcal{R}. \tag{7}$$

The expected sum of the rewards of the strategy profile $\mathbf{a} = [(c_n, \gamma_n)]_{n\in[N]}$ is:

$$g(\mathbf{a}) = \sum_{n=1}^{N} g_n(\mathbf{a}) = \sum_{n=1}^{N} \mu_{n,(c_n,\gamma_n)}\eta_{c_n}(\mathbf{a}).$$

Similarly, the expected sum of the rewards of the strategy profile $\tilde{\mathbf{a}} = [(c_n, \gamma^*_n)]_{n\in[N]}$ is:

$$g(\tilde{\mathbf{a}}) = \sum_{n=1}^{N} g_n(\tilde{\mathbf{a}}) = \sum_{n=1}^{N} \mu_{n,(c_n,\gamma^*_{c_n})}\eta_{c_n}(\mathbf{a}).$$

Using (7), we obtain that $g(\mathbf{a}) \leq g(\tilde{\mathbf{a}})$. ■

[3]Indeed, it is from the set of orthogonal allocations in $\tilde{\mathcal{A}}$.

Algorithm 1 GoT-SHOE

Input: set of channels $\mathcal{K}$, set of rates $\mathcal{R}$, time horizon T
Initialization: Set $\phi > 0$ and $\epsilon > 0$
Set exploration phase length T_e and GoT phase length T_g
$(\bar{\mu}_n, \bar{\gamma}_n^*) =$ SHOE$(\mathcal{K}, \mathcal{R}, T_e)$
$c_n^* =$ GoT$(\mathcal{K}, \mathcal{R}, T_g, \epsilon, \phi, \bar{\mu}_n, \bar{\gamma}_n^*)$
EXP$(T - T_e - T_g, c_n^*, \gamma_{n,c_n^*,b})$

IV. The Learning Algorithm

We propose an algorithm for decentralized dynamic rate and channel selection that (i) learns the optimal rate for each (user, channel) pair based on sequential elimination of suboptimal rates and (ii) employs a distributed agreement scheme as in [9] to settle users on orthogonal channels while achieving the highest sum of expected rewards. In the optimal strategy profile of this setting, each user picks a different channel in order not to cause a linear regret. Note that if the users estimate the best rate for each channel, then the problem would turn into the optimal channel allocation problem.

The proposed algorithm is composed of *exploration*, *Game of Thrones (GoT)* and *exploitation* phases as in [9]. Unlike [9], which is only interested in channel scheduling, we consider rate adaptation and channel scheduling jointly by eliminating suboptimal rates sequentially. Thus, we name our algorithm Game of Thrones with Sequential Halving Orthogonal Exploration (GoT-SHOE) (pseudocode is given in Algorithm 1). During the exploration phase, the expected rewards of the (channel, rate) pairs are estimated in a way that minimizes the number of collisions. The rate selection process for each (user, channel) pair is performed in a way that at the end of this phase, there will remain a single rate for each (user, channel) pair. Users utilize these estimated best rates for all the channels which reduce the strategy space to K (channel, channel's estimated best rate) pairs for each user. These K pairs are given to the GoT phase in order to identify the optimal pair. In the end, the best allocation is exploited in the exploitation phase. The details of the phases are provided in the following sections.

A. Sequential Halving Orthogonal Exploration

This phase consists of T_e number of rounds. In comparison to [9] where there are only K arms, our setup involves $K \times R$ arms for each user. As the number of (channel, rate) pairs increases, random exploration would not be a reasonable choice for learning the optimal arms because of two reasons: First of all, it takes too long to sample all the pairs sufficiently, and secondly, it leads to a high number of collisions in the network which indeed slows down the learning process and degrades the performance. In order to overcome these limitations, we develop a new exploration method called *Sequential Halving Orthogonal Exploration*. Our method adopts and mixes techniques from [12] and [10]. Flowchart and pseudocode of this phase are given in Fig. 2 and Algorithm 2 respectively.

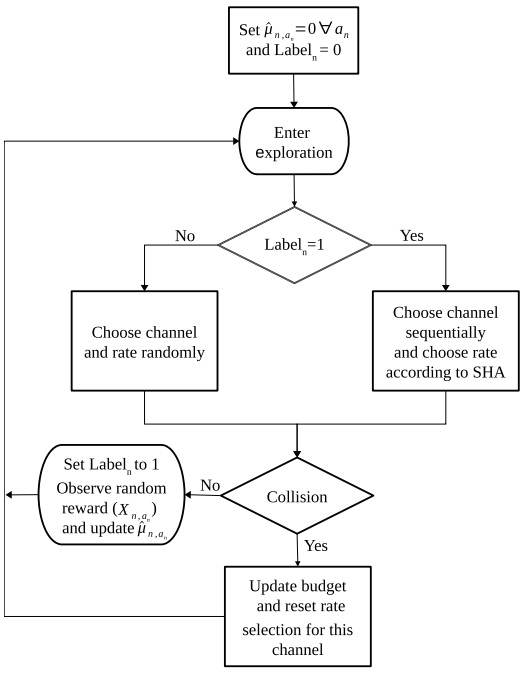


Fig. 2. Flowchart of sequential halving orthogonal exploration.

1) Channel Allocation: Channel allocation is inspired from [12], where the idea of orthogonalization is used. At first, each user starts selecting a channel randomly in each round. We refer to this sub-phase as *random selection* (RS). Once a user finds a collision-free channel, she enters the *sequential selection* (SS) sub-phase and for the remaining rounds, she simply selects channels sequentially in each round, i.e., $c_n(t+1) = c_n(t) \mod K + 1$. Let $T_{RS,n}$ and $T_{SS,n}$ be the rounds of exploration in which user n is in RS and SS sub-phase respectively. We have $T_e = T_{RS,n} + T_{SS,n},\ \forall n \in \mathcal{N}$. Also let $T_{SS,n,c}$ be the rounds of exploration in which user n selects channel c in SS sub-phase. We have the following relations:

- $\forall n \in \mathcal{N}$:

$$\sum_{c \in \mathcal{K}} T_{SS,n,c} = T_{SS,n}. \tag{8}$$

- $\forall n \in \mathcal{N}$ and $\forall c \in \mathcal{K}$:

$$\left\lfloor \frac{T_{SS,n}}{K} \right\rfloor \leq T_{SS,n,c} \leq \left\lceil \frac{T_{SS,n}}{K} \right\rceil. \tag{9}$$

Note that once a user enters the SS sub-phase, she will never come back to the RS sub-phase again and when all the users get into the SS sub-phase, there are no more collisions in this phase afterwards. Compared to the random exploration used in [9], this method reduces the number of collisions significantly which is crucial since most of the cognitive users are battery-powered. Thus, reducing the number of collisions prevents wastage of resources and increases opportunities for exploring different (channel, rate) pairs. From now till the end of this sub-section, whenever we refer to channel we mean the selected channel.

2) Rate Selection: Right after selecting the channel, the rate has to be chosen. Depending on the sub-phase, users select the rate differently. When a user is in RS sub-phase, she will select

Algorithm 2 Sequential Halving Orthogonal Exploration (SHOE)

Input: $\mathcal{K},\ \mathcal{R},\ T_e$
Initialization: Set $t = 1$, $\text{Label}_n = 0$, $\hat{\mu}_{n,(i,j)} = 0,\ V_{n,(i,j)} = 0,\ S_{n,(i,j)} = 0,\ \forall i \in \mathcal{K}$ and $\forall j \in \mathcal{R}$, $c_n(1) \sim U(1,\ldots,K)$ and $\gamma_n(1) \sim U(1,\ldots,R)$
while $t \leq T_e$ **do**
  Transmit a packet on channel $c_n(t)$ with rate $\gamma_n(t)$, and observe feedback $\eta_{c_n(t)}(\mathbf{a}(t))$
  if (no collision) **then**
    if ($\text{Label}_n = 0$) **then**
      Set $\text{Label}_n = 1$
      $\forall c \in \mathcal{K} : \mathcal{R}_{n,c} \leftarrow \mathcal{R},\ T_{e,n,c} = T_e - t + 1$
    end if
    Observe $X_{n,a_n}(t)$
    $V_{n,a_n(t)} = V_{n,a_n(t)} + 1$
    $S_{n,a_n(t)} = S_{n,a_n(t)} + X_{n,a_n}(t)$
    $\hat{\mu}_{n,a_n(t)} = \dfrac{\gamma_n(t)}{\gamma_R}\dfrac{S_{n,a_n(t)}}{V_{n,a_n(t)}}$
    $c_n(t+1) = c_n(t) \mod K + 1$
    $(\gamma_n(t+1), \mathcal{R}_{n,c_n(t+1)}) = \text{SHA}(c_n(t+1), T_{e,n,c_n(t+1)}, \mathcal{R}_{n,c_n(t+1)}, [V_{n,c_n(t+1),\gamma}]_{\gamma \in \mathcal{R}_{n,c_n(t+1)}})$
  else if (collision) **then**
    if ($\text{Label}_n = 0$) **then**
      $c_n(t+1) \sim U(1,\ldots,K)$
      $\gamma_n(t+1) \sim U(1,\ldots,R)$
    else
      $T_{e,n,c_n(t)} = T_e - t$
      $V_{n,c_n(t),\gamma} = 0$ and $S_{n,c_n(t),\gamma} = 0\ \forall \gamma \in \mathcal{R}$
      $\mathcal{R}_{n,c_n(t)} \leftarrow \mathcal{R}$
      $c_n(t+1) = c_n(t) \mod K + 1$
      $(\gamma_n(t+1), \mathcal{R}_{n,c_n(t+1)}) = \text{SHA}(c_n(t+1), T_{e,n,c_n(t+1)}, \mathcal{R}_{n,c_n(t+1)}, [V_{n,c_n(t+1),\gamma}]_{\gamma \in \mathcal{R}_{n,c_n(t+1)}})$
    end if
  end if
  $t \leftarrow t + 1$
end while
$\gamma_{n,c,b} \leftarrow$ a randomly selected rate from $\mathcal{R}_{n,c},\ \forall c \in \mathcal{K}$
$\bar{\gamma}_n^* = [\gamma_{n,c,b}]_{c \in \mathcal{K}}$
$\bar{\mu}_n = [\hat{\mu}_{n,(c,\gamma_{n,c,b})}]_{c \in \mathcal{K}}$
return $\bar{\mu}_n, \bar{\gamma}_n^*$

a rate uniformly at random. Once she enters the SS sub-phase, rate selection is performed separately for each channel based on the Sequential Halving algorithm in [10].

Let $\tau_{n,c}(t)$ be the time index of the last round up to round t in which user n collided with any other user when her channel is c. The *budget* for user n and her channel c in round t is defined as:

$$\text{Budget}_{n,c}(\tau_{n,c}(t)) = \left\lfloor \frac{T_e - \tau_{n,c}(t)}{K} \right\rfloor. \tag{10}$$

According to Sequential Halving algorithm in [10], the given budget for each (user, channel) pair will be split evenly across $\lceil \log_2 R \rceil$ elimination stages and rates will be played uniformly within a stage. At the end of a stage, the worst half of the rates which have the lowest estimated expected rewards will be removed from the rate set. For (user, channel) pair (n, c), we denote the set of remaining rates in stage s by $\mathcal{R}_{n,c,s}$, e.g., $\mathcal{R}_{n,c,0} = \mathcal{R},\ \forall (n,c) \in \mathcal{N} \times \mathcal{K}$.

Algorithm 3 Sequential Halving Algorithm (SHA)

Input: $c_n(t+1),\ T_{e,n,c_n(t+1)},\ \mathcal{R}_{n,c_n(t+1),s}$, $[V_{n,c_n(t+1),\gamma}]_{\gamma \in \mathcal{R}_{n,c_n(t+1),s}}$
if in the current stage, all rates in $\mathcal{R}_{n,c_n(t+1),s}$ are selected $\left\lfloor \frac{T_{e,n,c_n(t+1)}}{K|\mathcal{R}_{n,c_n(t+1),s}|\lceil \log_2 R \rceil} \right\rfloor$ number of rounds **then**
  Update $\mathcal{R}_{n,c_n(t+1),s}$ to be the set of $\left\lceil \frac{|\mathcal{R}_{n,c_n(t+1),s}|}{2} \right\rceil$ rates in $\mathcal{R}_{n,c_n(t+1),s}$ with the highest estimated throughputs
  $\gamma_n(t+1) \leftarrow$ First rate in $\mathcal{R}_{n,c_n(t+1),s}$
else
  $\gamma_n(t+1) \leftarrow$ Next rate in $\mathcal{R}_{n,c_n(t+1),s}$ that comes after the last rate played for $c_n(t+1)$
end if
return $\gamma_n(t+1), \mathcal{R}_{n,c_n(t+1),s}$

Meanwhile, user n might face collisions with other users that are in RS sub-phase. If such event happens on a given channel c, that user will update her budget for that channel using the updated value of $\tau_{n,c}(t)$ and reset the rate selection process (Sequential Halving algorithm) of that channel.[4] Let $\gamma_{n,c,b}$[5] be the estimated best rate for (user, channel) pair (n, c) at the end of the exploration phase. The vectors $\bar{\gamma}_n^* = [\gamma_{n,c,b}]_{c \in \mathcal{K}}$ and $\bar{\mu}_n = [\hat{\mu}_{n,(c,\gamma_{n,c,b})}]_{c \in \mathcal{K}}$ are provided to the GoT phase as inputs.

B. Game of Thrones (GoT)

The pseudocode of this phase is given in Algorithm 4. This phase consists of a T_g number of rounds. Similar to the GoT phase in [9], the strategy space of this phase consists of K channels with their corresponding estimated best rates. The empirical estimates are used as deterministic utilities for the GoT dynamics, i.e.,

$$u_n(\mathbf{a}) := \hat{\mu}_{n,(c_n,\gamma_{n,c_n,b})}\eta_{c_n}(\mathbf{a}). \tag{11}$$

Let $u_{n,max} := \max_{c \in \mathcal{K}} \hat{\mu}_{n,(c,\gamma_{n,c,b})}$. Each user has a state including a baseline action and a content/discontent (C/D) status. Each user starts with the content state and her baseline action is a random channel. She will select a channel randomly while she is discontent. Once she becomes content, she will select her baseline action with high probability. The transitions between the states are given in Algorithm 4. These dynamics guarantee that the optimal arms will be played a significant amount of time given that the utilities form an ergodic Markov chain.

[4] In practice, channel orthogonalization is fast, i.e., the users find orthogonal channels in a small number of rounds, and thus, the number of collisions is small. Moreover, collisions only appear in the early rounds. As a consequence, resetting SHA does not significantly degrade the performance.

[5] When the (user, channel) pair is clear from context, we suppress n and c.

Algorithm 4 Game of Thrones (GoT)

Input: $\mathcal{K},\ \mathcal{R},\ T_g,\ \epsilon,\ \phi,\ \bar{\mu}_n,\ \bar{\gamma}_n^*$
Initialization: Set $t=1$, $M_n=C$, and $\bar{c}_n \sim U(1,..,K)$
while $t \le T_g$ **do**
  if $(M_n = C)$ **then**
    $$p_n(c_n) = \begin{cases} \frac{\epsilon^\phi}{K-1} & c_n \neq \bar{c}_n \\ 1-\epsilon^\phi & c_n = \bar{c}_n \end{cases}$$
  else
    $p_n(c_n) = \frac{1}{K}$
  end if
  Choose a channel according to $c_n \sim p_n$
  Transmit a packet on the channel c_n given rate $\gamma_{n,c_n,b}$
  Observe $\eta_{c_n}(\mathbf{a}(t))$
  if $c_n \neq \bar{c}_n$ or $u_n(\mathbf{a}(t))=0$ or $M_n = D$ **then**
    Change the state:
    $$[\bar{c}_n, C/D] \to \begin{cases} [c_n, C] & \frac{u_n \epsilon^{u_{n,max}-u_n}}{u_{n,max}} \\ [c_n, D] & 1-\frac{u_n \epsilon^{u_{n,max}-u_n}}{u_{n,max}} \end{cases}$$
  end if
  $t \leftarrow t+1$
end while
$F_n(i,\gamma_{n,i,b}) := \sum_{t\in\mathcal{G}} \mathbb{1}(a_n(t)=(i,\gamma_{n,i,b}), M_n(t)=C)$, $\forall i \in \mathcal{K}$, where $\mathcal{G}$ is the set of rounds in the GoT phase
$c_n^* = \operatorname{argmax}_{i\in\mathcal{K}} F_n(i,\gamma_{n,i,b})$
return c_n^*

Algorithm 5 Exploitation (EXP)

Input: $T-T_e-T_g,\ c_n^*,\ \gamma_{n,c_n^*,b}$
Set $t=1$
while $t \le T-T_e-T_g$ **do**
  Transmit on the channel c_n^* with the rate $\gamma_{n,c_n^*,b}$
  $t \leftarrow t+1$
end while

C. Exploitation

The pseudocode of this phase is given in Algorithm 5. In this phase, each user selects the fixed (channel, the channel's estimated best rate) pair, which has the highest number of times being played resulting in being content in the GoT phase.

V. Regret Analysis

A. Preliminaries

Let $J_1 := \sum_{n=1}^N g_n(\mathbf{a}^*)$ be the value of the optimal assignment, $\mathbf{a}' \in \operatorname{argmax}_{\mathbf{a}\in\tilde{\mathcal{A}}:\mathbf{a}\neq\mathbf{a}^*} \sum_{n=1}^N g_n(\mathbf{a})$ be a second best assignment, and $J_2 := \sum_{n=1}^N g_n(\mathbf{a}')$ be its value. Let $\mathcal{A}_n' := \{(c,\gamma_{n,c,b}) : c \in \mathcal{K}\}$ represent the set of available actions of user n in the GoT phase and $\mathcal{A}' := \mathcal{A}_1' \times \ldots \times \mathcal{A}_N'$. A Markov chain is induced by the GoT dynamics over the state space $Z = \prod_n (\mathcal{A}_n' \times \mathcal{M})$, where $\mathcal{M} = \{C, D\}$. The transition matrix of this Markov chain depends both on ϵ and $\bar{\boldsymbol{\mu}} = \{\bar{\mu}_n\}_{n\in\mathcal{N}}$, and is denoted by $P^{\epsilon,\bar{\boldsymbol{\mu}}}$. Since $P^{\epsilon,\bar{\boldsymbol{\mu}}}$ is a random matrix, we need to analyze the convergence of GoT dynamics for each realization of $\bar{\boldsymbol{\mu}}$. Let $(\Omega,\mathcal{F},P)$ be the probability space over which $\bar{\boldsymbol{\mu}}$ is defined, and for $\omega\in\Omega$, let $P_\omega^\epsilon = P^{\epsilon,\bar{\boldsymbol{\mu}}(\omega)}$ represent a particular realization of this matrix. In the following discussion, subscript ω is used to indicate a particular realization of the random quantity involved.

Let $\mathbf{a}^{b*} := \operatorname{argmax}_{\mathbf{a}\in\mathcal{A}'} \sum_{n=1}^N u_n(\mathbf{a})$ represent the estimated optimal action profile at the end of the exploration rounds. We can write the optimal state as $z^* = [\mathbf{a}^{b*}, C^N]$. Denote the stationary distribution of Z by π. GoT dynamics ensure concentration of the stationary distribution to the estimated optimal action profile. According to [9, Theorem 2], if for a given $\omega\in\Omega$ and $\phi \ge \sum_n u_{n,max,\omega} - J_1$, the Markov chain (Z, P_ω^ϵ) is ergodic for all $\epsilon \in (0,1)$, then for any $0<\rho<\frac{1}{2}$ there exists a small enough $\epsilon_\omega > 0$ such that $\pi_{z_\omega^*,\omega} > \frac{1}{2(1-\rho)}$ or equivalently $(1-\rho)\pi_{z_\omega^*,\omega} > \frac{1}{2}$ (see [9, Eqns. A.20∼A.21] for more details). Here and below, we suppressed the dependence of stationary distribution on ϵ_ω in the notation for the sake simplicity. Finally, according to [9, Lemma 5], for this small enough $\epsilon_\omega > 0$ we have:

$$\begin{aligned} P_{g,\omega} &:= \Pr\left(\sum_{t\in\mathcal{G}} \mathbb{1}(z(t)=z_\omega^*) \le (1-\rho)\pi_{z_\omega^*,\omega} T_g |\bar{\boldsymbol{\mu}}(\omega)\right) \\ &\le B_0 ||\varphi||_{\pi_\omega} e^{-\frac{\rho^2 \pi_{z_\omega^*,\omega} T_g}{72 T_{m,\omega}(\frac{1}{8})}} \end{aligned} \tag{12}$$

where $T_{m,\omega}(\frac{1}{8})$ is the mixing time of $(Z, P_\omega^{\epsilon_\omega})$ with an accuracy of $\frac{1}{8}$, B_0 is a constant independent of $\pi_{z_\omega^*,\omega}$ and ρ, and $||\varphi||_{\pi_\omega} = \sqrt{\sum_{i=1}^{|Z|} \frac{\varphi_i^2}{\pi_{i,\omega}}}$ where φ_i is the probability distribution of the state i at the beginning of the GoT phase, i.e.,

$$\varphi_i = \begin{cases} \frac{1}{K^N} & \text{if } i=[\mathbf{a},C^N] \text{ for some } \mathbf{a}\in\mathcal{A}' \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

Note that GoT dynamics may fail to converge when the ergodicity of the induced Markov chain does not hold, which can happen with non-zero probability as we consider the case with binary feedback instead of the case with continuous feedback studied in [9]. In Appendix A, we confirm that when the exploration phase is long enough to have all GoT utilities positive, the induced Markov chain is indeed ergodic (with high probability) and $T_m(\frac{1}{8})$ and $||\varphi||_\pi$ are almost surely bounded.

B. Main Result

Our main result is given in the following theorem.

Theorem 1: Fix $\rho\in(0,1/2)$ and let $A := B_0 \times ||\varphi||_\pi$. For all $\Delta < \min\{\min_{n,a_n} \mu_{n,a_n}, \frac{2(J_1-J_2)}{5N}\}$, $\phi \ge N(1+\Delta)-J_1$,[6] small enough $\epsilon > 0$ and $\eta \in (0,1)$, if all the users play according to GoT-SHOE algorithm with K channels and R rates for T rounds with the exploration length

$$\begin{aligned} T_e \ge & \left\lceil \frac{\log(\eta/3K)}{\log(1-1/4K)} \right\rceil \\ & + \max\left(\left\lceil 8KH_{\max} \log_2 R \log(\frac{18NK\log_2 R}{\eta}) \right\rceil + K, \right. \\ & \left. \left\lceil \frac{K\log_2 R}{2\Delta^{2\frac{R-1}{R}}} \log\left(\frac{12NKe^{2\Delta^2\log_2 R}}{\eta}\right) \right\rceil \right) \end{aligned} \tag{14}$$

[6]Since N and J_1 are not known to the users, the upper bound $\phi \ge K(1+\Delta)$ will also work.

and GoT length

$$T_g \geq \left\lceil \frac{72 T_m(1/8)}{\rho^2 \pi_{z^*}} \log(\frac{3A}{\eta}) \right\rceil, \tag{15}$$

then with probability at least $1-\eta$*, the regret is upper bounded by*

$$Reg(T) \leq N(T_e + T_g). \tag{16}$$

Theorem 1 shows that the regret of GoT-SHOE is bounded with a high probability given that T_e and T_g are set long enough. While the exact values for some of the variables in (14) and (15) are unknown to the users, they can be upper bounded. For instance, K can be used as an upper bound for N. If users' rewards are multiples of a common resolution $\Delta_{\min}$ like the QoS as in [25], then $\Delta_{\min}$ can be used as a lower bound for $J_1 - J_2$ to find an appropriate Δ, and also as a lower bound for the denominator of $H_{\max}$. While theoretical results require certain bounds on the parameters of GoT-SHOE, we show in Section VII that GoT-SHOE learns well when these parameters are reasonably chosen.

As none of these constants grow with T, if T_e and T_g are set as $O(\log(T))$ by all users, then both (14) and (15) will be satisfied for T large enough even when we set $\eta = 1/T$. In this case the regret will be $O(\log(T))$ with probability at least $1 - 1/T$ and $O(T)$ with probability at most $1/T$, which implies an expected regret bound of $O(\log(T))$. One can easily check that the leading term ($\log T$ term) on that bound has logarithmic dependence on R.

Corollary 1: For T *large enough, when* T_e *and* T_g *are set as* $O(\log T)$ *by all users, then we have* $\overline{Reg}(T) = O(\log(T))$.

C. Facts and Lemmas for the Regret Analysis

Let $T_{SS,\min}$ be the minimum value of $T_{SS,n}$ among the users, i.e.,

$$T_{SS,\min} := \min_{n \in \mathcal{N}} T_{SS,n}. \tag{17}$$

Similarly, let $T_{RS,\max}$ be the maximum value of $T_{RS,n}$ among the users, i.e.,

$$T_{RS,\max} := \max_{n \in \mathcal{N}} T_{RS,n}. \tag{18}$$

Lemma 2: Let $\eta_1 \in (0,1)$ *and* $T_{RS}(\eta_1) := \lceil \frac{\log(\eta_1/K)}{\log(1-1/4K)} \rceil$. *After* $T_{RS}(\eta_1)$ *rounds of exploration, all the players will be orthogonalized with probability at least* $1 - \eta_1$.

Lemma 2 tells us that with probability at least $1-\eta_1$, we have $T_{RS,\max} \leq T_{RS}(\eta_1)$.

Definition 1: Player n *is said to have a* Δ*-correct estimate of arm* a_n *at the end of the exploration phase if the difference between the estimated expected reward of that arm and its true value is less than* Δ*, i.e.,* $|\hat{\mu}_{n,a_n} - \mu_{n,a_n}| < \Delta$.

Lemma 3: For all strategy profiles $\boldsymbol{a} \in \tilde{\mathcal{A}}$*, let* $\hat{\mu}_{n,a_n}$ *be such that* $|\hat{\mu}_{n,a_n} - \mu_{n,a_n}| < \Delta$, $\forall n \in \mathcal{N}$. *If* $\Delta < \frac{2(J_1 - J_2)}{5\ N}$, *then the optimal assignment does not change due to the uncertainty, i.e.,* $\operatorname{argmax}_{\boldsymbol{a} \in \tilde{\mathcal{A}}} \sum_{n=1}^{N} \hat{\mu}_{n,a_n} = \boldsymbol{a}^*$. *Moreover, we have* $\sum_{n=1}^{N} \hat{\mu}_{n,a_n^*} - \operatorname{argmax}_{\boldsymbol{a} \in \tilde{\mathcal{A}} \setminus \{\boldsymbol{a}^*\}} \sum_{n=1}^{N} \hat{\mu}_{n,a_n} > \frac{(J_1 - J_2)}{5}$.

It is inferred from Lemma 3 that if i) the best rate is estimated correctly for each (user, channel) pair, ii) each user has Δ-correct estimate of its (channel, estimated best rate) pairs, and iii) $\Delta < \frac{2(J_1 - J_2)}{5N}$, then the optimal state of the GoT dynamics correspond to the optimal assignment.

Lemma 4: Let $\eta_2 \in (0,1)$. *Let* $\mathcal{E}$ *be the event that* $T_{SS,\min} \geq T_{SS}(\eta_2) := \max(T_1(\eta_2), T_2(\eta_2))$, *where*

$$T_1(\eta_2) := \left\lceil 8KH_{\max} \log_2 R \log\left(\frac{6NK \log_2 R}{\eta_2}\right) \right\rceil + K \tag{19}$$

and

$$T_2(\eta_2) := \left\lceil \frac{K \log_2 R}{2\Delta^2 \frac{R-1}{R}} \log\left(\frac{4NKe^{2\Delta^2 \log_2 R}}{\eta_2}\right) \right\rceil. \tag{20}$$

Under event $\mathcal{E}$*, at the end of the exploration phase, the best rates are correctly identified for each (user, channel) pair and each user has the* Δ*-correct estimate of (channel, estimated best rate) pairs with probability at least* $1 - \eta_2$.

Lemma 5: Fix the set of utilities for the GoT dynamics, and let $\eta_3 \in (0,1)$. *If the GoT phase is run for at least* $T_g(\eta_3) := \lceil \frac{72\ T_m(1/8)}{\rho^2 \pi_{z^*}} \log(\frac{A}{\eta_3}) \rceil$ *number of rounds, then the optimal state will be played at least half of the rounds in GoT with probability at least* $1 - \eta_3$.

Proof: According to (12), when the GoT phase is run for T_g number of rounds, then the error probability of the GoT phase (P_g) is bounded as $P_g \leq Ae^{-\frac{\rho^2 \pi_{z^*} T_g}{72 T_m(\frac{1}{8})}} \leq Ae^{-\frac{\rho^2 \pi_{z^*} T_g(\eta_3)}{72 T_m(\frac{1}{8})}} \leq \eta_3$. ∎

D. Proof of Theorem 1

According to Lemma 2, after $T_{RS}(\frac{\eta}{3})$ number of rounds in exploration phase, all the players will be orthogonalized with probability at least $1-\frac{\eta}{3}$. Furthermore, according to Lemma 4, after $T_{SS}(\frac{\eta}{3})$ number of rounds in SS sub-phase of exploration phase, we will have Δ-correct estimate of (channel, estimated best rate) pairs with probability at least $1 - \frac{\eta}{3}$. Since $\Delta < \frac{2(J_1 - J_2)}{5}$, according to Lemma 3 the optimal assignment given the estimated rewards is unique, and coincides with the optimal assignment given the expected rewards with probability at least $1 - \frac{\eta}{3}$. Moreover, since $\Delta < \min_{n,a_n} \mu_{n,a_n}$, we are ensured that the mixing time of the GoT dynamics is bounded almost surely under this event (see Fact 1). Having $\phi \geq N(1+\Delta) - J_1$ guarantees the existence of an ϵ for which the stationary probability of the optimal state is greater than $\frac{1}{2}$. Therefore, according to Lemma 5, if ϵ is set to be small enough and the GoT phase is run for $T_g(\frac{\eta}{3})$ number of rounds, the optimal state will be played most of the rounds in GoT phase with probability at least $1-\frac{\eta}{3}$. Hence we conclude that if the length of the exploration phase is set to be $T_e \geq T_{RS}(\frac{\eta}{3}) + T_{SS}(\frac{\eta}{3})$ number of rounds and the length of the GoT phase is set to be $T_g \geq T_g(\frac{\eta}{3})$ number of rounds, then with probability at least $1 - \eta$, the regret is at most $N(T_e + T_g)$.

E. Extension to $N > K$

When the number of channels is less than the number of users, the optimal strategy is to assign those K channels to K different users which maximize the expected reward and urge the other $N - K$ users to leave the game. In this case,

in order to find the optimal allocation in the GoT phase, in addition to the available K channels, each user must be able to refrain from selecting a channel in some rounds. For this purpose, we introduce the idea of *virtual channels*. Once a user selects the virtual channel, she actually selects none of the real channels. This is similar to the idea of adding zero columns to an unbalanced assignment problem. However, in contrast to the assignment problem, it is impossible to assign zero utility to the virtual channels since according to GoT dynamics, no user can remain content selecting an arm with zero utility. On the other hand, in order to preserve the optimal allocation, the utility of the virtual channel of different users must be the same and it has to be less than the minimum utility of the system. It should be obvious that there is no collision defined on the virtual channel, i.e., two different users can select their virtual channels and receive nonzero utility. At the end of the GoT phase, if a user is assigned to her virtual channel, she will leave the game. First, consider the case when N is known by all users. In order to have orthogonal exploration as in SHOE, each user must start the exploration by selecting one of the K real channels at random, and once she finds a collision-free channel, she enters SS sub-phase and selects among $K+1$ channels (K real and one virtual channel). However, when she reaches the virtual channel, she has to select it for $N-K$ consecutive rounds so that the period of selecting each real channel would be N.

However, N is assumed to be unknown in our problem. Similar to [11], by selecting the channels randomly for T_{est} number of rounds and keeping track of the number of collisions till round T_{est} (denoted by $C_{T_{\text{est}}}$), each user will be able to estimate N as:

$$N^{\text{est}} = \min\left(\text{round}\left(\frac{\log(\frac{T_{\text{est}}-C_{T_{\text{est}}}}{T_{\text{est}}})}{\log(1-\frac{1}{K})}+1\right), N^{\text{upper}}\right)$$

where N^{upper} is the upper bound on the number of users in the system. The following lemma tells us that if N^{upper} is given, then with a sufficient amount of random exploration, each user can estimate the number of users with high probability.

Lemma 6: Let $\eta_4 \in (0,1)$. Let N^{upper} be an upper bound on the number of users in the system, i.e., $N \leq N^{upper}$ with probability one. Let $\Upsilon = (1-1/K)^{N^{upper}-1}\frac{0.49}{K}$ and $T_{est}(\eta_4) := \lceil \frac{\log(2/\eta_4)}{2\Upsilon^2} \rceil$. After $T_{est}(\eta_4)$ rounds of random exploration, we have $N^{est} = N$ with probability at least $1-\eta_4$.

The proof of this lemma is similar to the proof of [11, Lemma 3] where we used N^{upper} instead of K as an upper bound for N. One may claim that once we are given N^{upper}, each user simply uses N^{upper} instead of N and apply SHOE as explained. This also can be done, however, when N^{upper} is much greater than N, users waste many rounds playing none of the real channels. That is why it is good to do the estimation. Note that the idea of estimating N can be used even when $N \leq K$, thus the users are not required to know whether $N \leq K$ or not. Note that the above approach pushes some users out of the game, thereby making their cumulative reward in the exploitation phase zero. Besides maximizing the system throughput when $N > K$, an alternative approach is to let all the users benefit from the resources equally. For instance, a fair scheduling can be achieved by the protocol described below.

- Channel selection can be done similar to the aforementioned procedure until each user finds a collision-free channel and enters the SS sub-phase. This time, however, there is no GoT and exploitation phases, and the users have to remain in the exploration phase for the entire game.
- For the rate selection, each user can implement an ordinary MAB algorithm (e.g., UCB or TS) for each channel to select a rate such that her expected reward on that channel is maximized.

In terms of channel selection, this scheduling (when all the players enter SS sub-phase) is nothing but the well-known *round-robin scheduling*. Another instance of fair scheduling is given in [34], where the authors proposed a distributed bandit approach that maximizes the minimum expected reward received by each player. Designing more practical fair scheduling algorithms (e.g., proportional fair scheduling) is an interesting topic that requires further investigation.

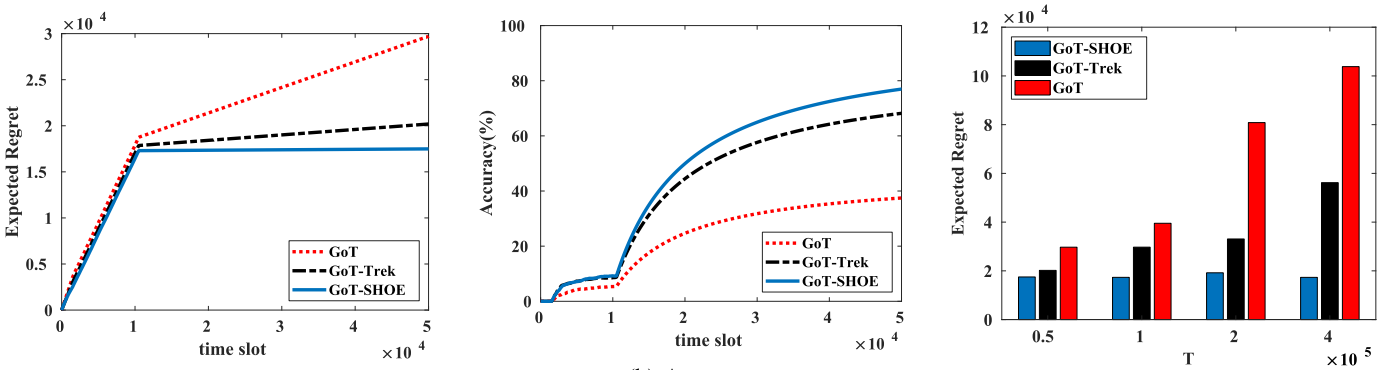
Algorithm 6 OALA-SHOE

Input: $\mathcal{K},\ \mathcal{R},\ \Delta_{min},\ \Delta_{max}$
Initialization: Set $\epsilon < \frac{\Delta_{min}}{4K}$, b
Set T_e and $T_A = \lceil 4K^2\ N\big(\frac{\max_{n,c,\gamma}\mu_{n,c,\gamma}}{\Delta_{min}} + \frac{1}{N}\big)\big(2^b+1\big)\rceil$
$(\bar{\mu}_n, \bar{\gamma}_n^*) =$ SHOE$(\mathcal{K}, \mathcal{R}, T_e)$
$c_n^* =$ Auction$(\mathcal{K}, \mathcal{R}, T_A, \epsilon, b, \bar{\mu}_n, \bar{\gamma}_n^*, \Delta_{min})$
EXP$(T - T_e - T_A, c_n^*, \gamma_{n,c_n^*,b})$

VI. OALA-SHOE Algorithm

While GoT is a distributed assignment algorithm that can make use of SHOE, SHOE can indeed be used with any distributed channel assignment algorithm. In this section, we discuss the integration of SHOE to Online Auction based Learning Algorithm (OALA) algorithm proposed in [25] and [26]. Different from GoT, OALA works under the additional assumptions that the reward of each arm is a multiple of a common known resolution $\Delta_{\min}$ and users are equipped with channel sensing capability before transmission. Under these assumptions, it implements a different distributed agreement scheme than that of GoT, which is shown to have better scaling with the number of users and channels [25].

In this section, we enhance OALA by using SHOE as its exploration mechanism. The resulting algorithm is called OALA-SHOE (the pseudocode is given in Algorithms 6 and 7). OALA-SHOE consists of exploration, auction, and exploitation phases. The rewards are estimated in the exploration phase, channel assignment is learned in the auction phase, and the estimated optimal assignment is played in the exploitation phase. Since OALA uses random selections in its exploration phase, it requires a long exploration phase in order to converge to the optimal assignment. On the other hand, SHOE's smarter exploration mechanism enables fast learning even when the length of the exploration phase is small. Regret bounds similar to the ones in Section V can also



(a) Regret (b) Accuracy (c) Regret for different T

Fig. 3. Comparison of GoT-SHOE with GoT and GoT-Trek. (a) and (b): Regret and accuracy comparison under the baseline configuration. (c): Regret comparison for different time horizons.

Algorithm 7 Auction

Input: $\mathcal{K},\ \mathcal{R},\ T_A,\ \epsilon,\ b,\ \bar{\mu}_n,\ \bar{\gamma}_n^*,\ \Delta_{min}$

$\varepsilon_{n,(c,\gamma_{n,c,b})} \sim U([-\frac{\Delta_{min}}{8N}, \frac{\Delta_{min}}{8N}]),\ \forall c \in \mathcal{K}$

$\hat{\mu}_{n,(c,\gamma_{n,c,b})} \leftarrow \hat{\mu}_{n,(c,\gamma_{n,c,b})} + \varepsilon_{n,(c,\gamma_{n,c,b})},\ \forall c \in \mathcal{K}$

//Artificial Dithering

Initialization: Set $t = 1$, State_n = unassigned and $B_{n,i,j} = 0\ \forall i \in \mathcal{K}$ and $\forall j \in \mathcal{R}$

while $t \leq T_A$ **do**

  if (State_n = unassigned) **then**

    $\vartheta_n^1 = \max_c(\hat{\mu}_{n,(c,\gamma_{n,c,b})} - B_{n,(c,\gamma_{n,c,b})})$

    $\tilde{c}_n = \text{argmax}_c(\hat{\mu}_{n,(c,\gamma_{n,c,b})} - B_{n,(c,\gamma_{n,c,b})})$

    $\vartheta_n^2 = \max_{c \neq \tilde{c}_n}(\hat{\mu}_{n,(c,\gamma_{n,c,b})} - B_{n,(c,\gamma_{n,c,b})})$

    $B_{n,(\tilde{c}_n,\gamma_{n,\tilde{c}_n,b})} = B_{n,(\tilde{c}_n,\gamma_{n,\tilde{c}_n,b})} + \vartheta_n^1 - \vartheta_n^2 + \epsilon$

  end if

  During the next 2^b time slots, sense the channel $\tilde{c}_n$ after a back-off time of

$$\zeta_n = f_b(B_{n,(\tilde{c}_n,\gamma_{n,\tilde{c}_n,b})})$$

  time slots, where f_b is a quantization of some decreasing function f (e.g., $f(x) = 2^b - x$) using b bits, such that $0 \leq \zeta_n \leq 2^b$.

  if (channel is not busy) **then**

    State_n = assigned

  end if

  $t \leftarrow t + 2^b + 1$

end while

return c_n^* : the assigned channel

be derived for OALA-SHOE. The advantage of SHOE over random explorations is shown in Section VII via experiments.

For the auction phase to converge to the optimal allocation, the following conditions should hold (for details, see [25]):

1) At the end of the exploration phase, the best rates must be correctly estimated for each (user, channel) pair and each user needs to have the Δ-correct estimate of (channel, estimated best rate) pairs, where $\Delta < \frac{3\Delta_{\min}}{8N}$.
2) b must be chosen large enough such that for any B_1 and B_2 such that $B_1 \neq B_2$, we have $f_b(B_1) \neq f_b(B_2)$.
3) Auction phase must run for $T_A = \lceil 4K^2\ N(\frac{\max_{n,c,\gamma}\mu_{n,c,\gamma}}{\Delta_{min}} + \frac{1}{N})(2^b + 1)\rceil$ number of rounds.

VII. Numerical Results

In this section, we compare the performances of SHOE based algorithms with other state-of-the-art algorithms. Our performance metrics are the expected regret (the less the better) and accuracy, where the accuracy is defined as the percentage of times the optimal assignment is selected (jointly) by the users, i.e.,

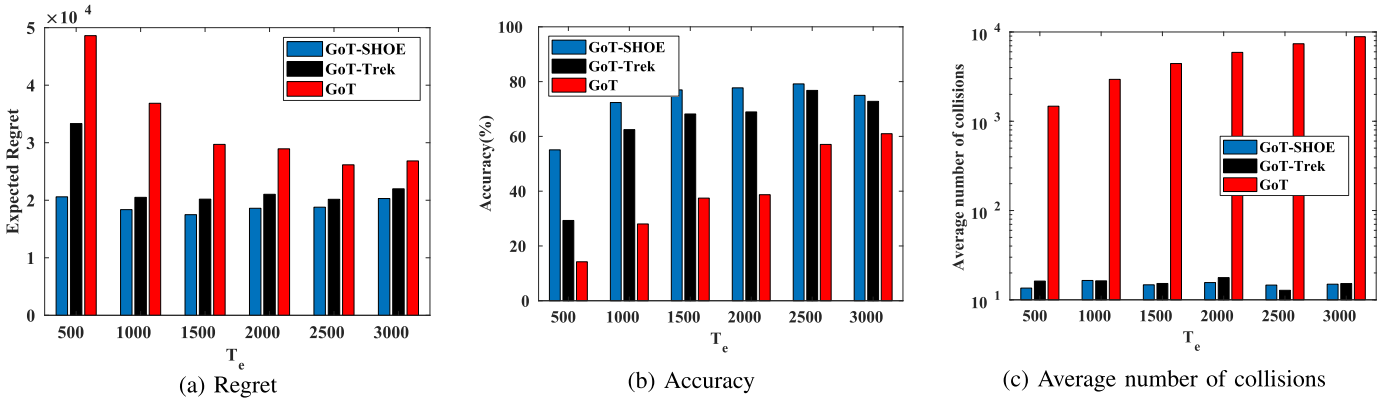
$$\text{Accuracy}(t) = \frac{\sum_{k=1}^{t} \mathbb{1}(\mathbf{a}(k) = \mathbf{a}^*)}{t} \times 100.$$

A. Experiment 1: GoT-SHOE

We compare GoT-SHOE with the following benchmarks:

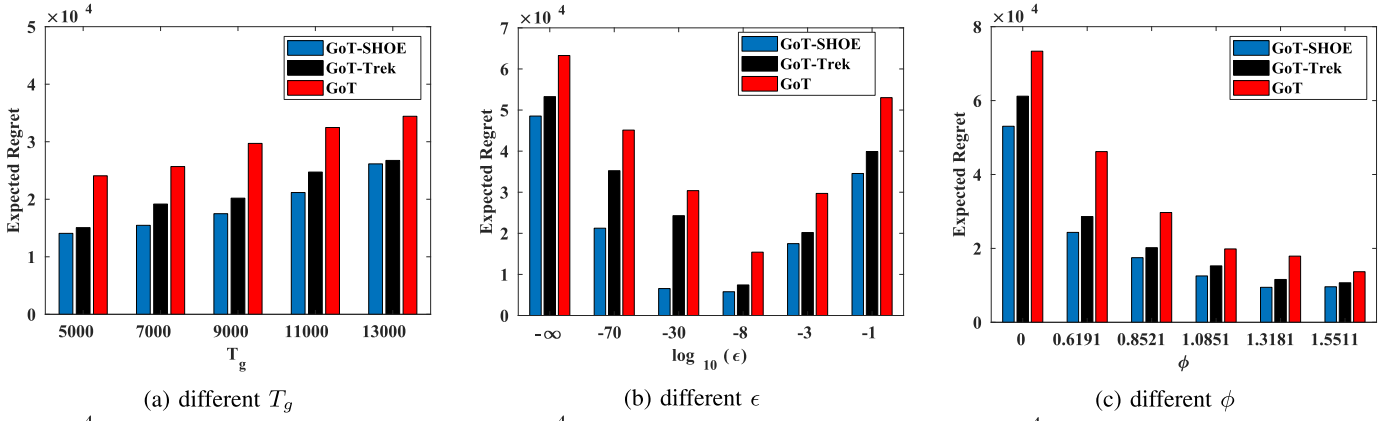
- *Game of Thrones* (GoT): A naive extension of the algorithm in [9], which randomly explores all (channel, rate) pairs. Note that since the doubling trick is not used in our algorithm, we used the one-shot version (known T) of GoT in order to have a fair comparison (see [9, Corollary 1]).
- *Game of Thrones with Trekking* (GoT-Trek): A variant of the algorithm in [9], which uses the idea of player orthogonalization [12] for channel selection. For each (user, channel) pair, rate selection is done in a round-robin fashion. This algorithm enhances the exploration phase of GoT by employing the trekking strategy in [12], and thus, serves as a good competitor benchmark.

1) Data Generation: In our simulation setup there are 5 users ($N = 5$) competing for 5 radio channels ($K = 5$). Each user can choose among 8 different transmission rates ($R = 8$), where $\mathcal{R} = \{6, 9, 12, 18, 24, 32, 48, 54\}$ *mega-bits-per-second* (Mbps). The packet successful transmission probabilities ($\theta_{n,(c_n,\gamma_n)}$'s) for different users are generated randomly. In the *baseline configuration* we set $T = 5 \times 10^4$, $T_e = 1500$, $T_g = 9000$, $\epsilon = 0.001$ and $\phi = \frac{\log(\frac{125}{NT_g})}{\log(\epsilon)} \simeq 0.8521$. Each experiment is repeated 100 times and the presented results are obtained via averaging over these runs.



(a) Regret (b) Accuracy (c) Average number of collisions

Fig. 4. Comparison of GoT-SHOE with GoT and GoT-Trek for different exploration lengths.



(a) different T_g (b) different ϵ (c) different ϕ

(d) different N (e) different R (f) estimate N

Fig. 5. Comparison of the expected regrets of GoT-SHOE, GoT and GoT-Trek under different parameters.

Fig. 3 demonstrates that GoT-SHOE significantly outperforms GoT and GoT-Trek both in terms of regret and accuracy. Random channel selections in GoT result in an excessive number of collisions, and inaccurate reward estimates which in turn increase the regret. On the other hand, although GoT-Trek avoids collisions by settling users to orthogonal channels, each user needs to explore all the rates which again results in high regret. In contrast, GoT-SHOE orthogonalizes fast and avoids over-exploration of inferior rates, resulting in a lower number of collisions, accurate reward estimates, and high cumulative reward. In addition, Fig. 3b shows that the final accuracy of GoT-SHOE is more than 8% higher than that of GoT-Trek and 40% higher than that of GoT. By the end, users were able to select the optimal assignment about 78% of the time by using GoT-SHOE.

2) Sensitivity Analysis: In order to show the effect of parameters on the performance of the considered algorithms, we perform a comprehensive set of experiments whose details are given below. In each of the following setups, we only vary the mentioned parameter, and other parameters are set to be the same as the ones in the baseline configuration.

- **Different time horizon (T):** Fig. 3c compares the expected regrets of the algorithms for different time horizons ($T \in \{0.5, 1, 2, 4\} \times 10^5$). The observed behavior can be projected from Fig. 3a. In Fig. 3a it is observed

that the expected regret of GoT-SHOE (almost) does not change after the GoT phase while the expected regrets of GoT-Trek and GoT increase with a small and a high slope respectively. Therefore, for the fixed exploration and GoT lengths ($T_e + T_g = 1500 + 9000 = 10500$), we expect to see (almost) the same expected regret ($\overline{\text{Reg}}(T)$) for GoT-SHOE and higher expected regrets for its competitors as T increases from 5×10^4.

- **Different exploration length (T_e):** Effective exploration provides fast convergence towards the optimal solution, and hence, we vary the exploration length to see the sensitivity of the algorithm to T_e. We give results for the expected regret, accuracy and the average number of collisions under different exploration lengths ($T_e \in \{500, 1000, 1500, 2000, 2500, 3000\}$) in Fig. 4. It is observed that GoT-SHOE consistently achieves the lowest regret and the highest accuracy for all values of T_e. Moreover, the expected regret of GoT-SHOE is less affected by the variations in T_e compared to the other algorithms. GoT requires T_e to be set large in order to achieve lower expected regret since its exploration mechanism is not as efficient as that of GoT-SHOE. It is observed in Fig. 4c that number of collisions in the exploration phases of GoT-Trek and GoT-SHOE is significantly less than that of GoT, which is important especially when the user devices are supplied by batteries. In terms of the number of collisions, GoT-SHOE and GoT-Trek behave similarly.
- **Different GoT length (T_g):** Results in Fig. 5a show that the expected regrets of all algorithms increase when T_g is increased from 5000 to 13000. While a longer GoT phase will increase the chance of settling to the optimal allocation in the exploitation phase, expected regret increases since suboptimal allocations can be selected during the GoT phase. Nevertheless, GoT-SHOE outperforms both GoT and GoT-Trek for all values of T_g.
- **Different ϵ:** Results in Fig. 5b show how the expected regret changes as the value of ϵ used in GoT dynamics varies. It is observed that choosing ϵ small enough is necessary to achieve low regret by converging to the optimal allocation in the exploitation phase. However, if $\epsilon^{u_{n,max}-u_n}$ becomes too small (near zero), no discontent user would be able to become content which avoids users from converging to the optimal allocation properly.
- **Different ϕ:** Note that ϵ^{ϕ} is the probability of escaping from the content state. For instance, by setting $\phi = \frac{\log(\frac{125}{NT_g})}{\log(\epsilon)}$, we obtain escape probability as $\frac{125}{NT_g}$. Fig. 5c demonstrates the fact that when escape probability is increased (ϕ decreases), the regret increases.
- **Different number of users (N):** As the number of users increases, more collisions are expected in the exploration phase. Furthermore, it takes longer for GoT dynamics to converge to the optimal allocation. Changes in the expected regret as a function of the number of users is provided in Fig. 5d.
- **Different number of rates (R):** The set of rates for different number of transmission rates $R \in \{2, 4, 6\}$ are given as: (i) $R = 6$: $\mathcal{R} = \{6, 12, 18, 32, 48, 54\}$, (ii) $R = 4$: $\mathcal{R} = \{6, 18, 32, 54\}$, (iii) $R = 2$: $\mathcal{R} = \{6, 54\}$. Fig. 5e shows that as the number of rates increases, the gap in terms of regret between GoT-SHOE and GoT increases. Thanks to SHA, the expected regret of GoT-SHOE does not degrade significantly when the number of rates increases from 2 to 8.
- **When the number of users has to be estimated (i.e., $N \leq K$ is not known a priori):** Here N is 5, and each user performs $T_{\text{est}} = 1200$ random exploration to estimate N, then she spends $T_e = 1500$ rounds in SHOE (the rest of the parameters are similar to the baseline configuration). Note that for the GoT algorithm, users do not need to estimate N and they perform random exploration over K channels for $T_e = 2700$ rounds. However, adding a virtual channel in the GoT phase is mandatory for GoT-SHOE and its competitors. The experiment is performed for two different values of K: (i) $K = 2 < N$, (ii) $K = 5 = N$. In both cases, it is observed that the users can successfully estimate the number of users in the system and GoT-SHOE outperforms its competitors (see Fig. 5f).

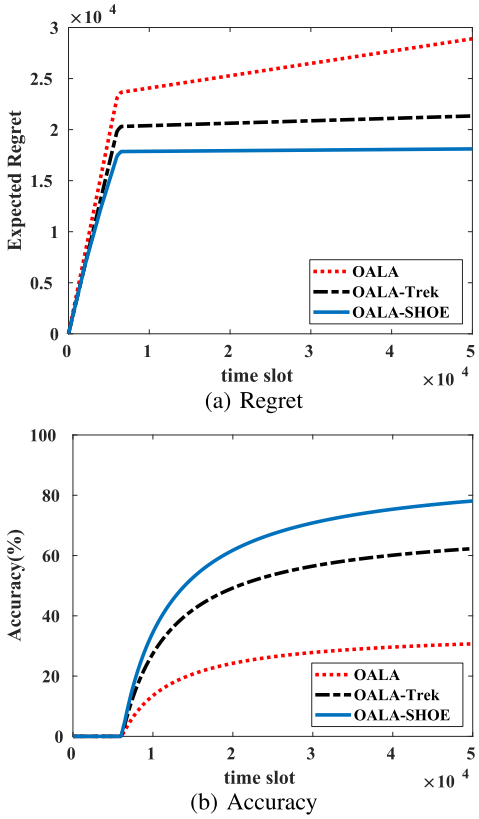


Fig. 6. Comparison of OALA-SHOE with OALA and OALA-Trek.

B. Experiment 2: OALA-SHOE

We compare OALA-SHOE with the following benchmarks:

- OALA: A naive extension of the algorithm in [26], which randomly explores all (channel, rate) pairs. Again here, we used the one-shot version of their algorithm in order to have a fair comparison.

- OALA with Trekking (OALA-Trek): A variant of the algorithm in [25], which uses the idea of player orthogonalization [12] for channel selection. For each (user, channel) pair, rate selection is done in a round-robin fashion. This algorithm enhances the exploration phase of OALA by employing the trekking strategy in [12], and thus, serves as a good competitor benchmark.

We consider 8 users that compete for 8 radio channels. Each user can choose among 8 different rates, where $\mathcal{R} = \{6, 9, 12, 18, 24, 32, 48, 54\}$ Mbps. We set $T = 5 \times 10^4$, $\epsilon = 0.01$ and Δ_{min} is assumed to be $\frac{1}{54}$. Given the true expected rewards of the users' (channel, bets rate) pairs, it turns out that 500 time slots is enough for the auction phase in order to converge to the optimal allocation. We set $T_e = 6000$. Each experiment is performed 100 times and results are obtained via averaging over these runs. Fig. 6a shows that the regret of OALA-SHOE at the final time slot is around 10% lower than that of OALA-Trek and 40% lower than that of OALA. Similarly, Fig. 6b shows that the final accuracy of OALA-SHOE is more than 10% higher than that of OALA-Trek and 40% higher than that of OALA.

VIII. Conclusion

In this paper, we proposed a decentralized learning algorithm for dynamic rate and channel adaptation over a shared spectrum. Our algorithm uses the ideas of orthogonal exploration and sequential halving to learn the best (channel, rate) pairs as fast as possible. We proved a logarithmic in time regret bound for our algorithm and showed that it can be used together with other distributed channel assignment algorithms when the users are required to select the transmission rate in addition to the channel. We provided extensive simulations to illustrate the superiority of the proposed algorithms over the state-of-the-art algorithms. These simulations also imply that the parameter-tuning for our algorithm is not difficult and one can easily find reasonable parameters by simulating random models for the unknown environment.

It is often the case that the expected throughput varies with the transmission rates in a structured way (such as being unimodal [1]). An interesting future research direction is to exploit such a structure in order to obtain faster convergence rates. Another interesting future research direction is to improve the auction algorithm. Specifically, one can investigate if convergence will be possible when the optimal assignment is not unique. Another point that deserves attention is how to design auction algorithms that converge when it is sufficient to select approximately optimal allocations instead of the optimal allocation.

Appendix A
Almost Sure Boundedness of $T_m(\frac{1}{8})$ and $||\varphi||_\pi$

For any $i \in \mathcal{N}$, let $\nu_i = \{\nu_{i,j}\}_{j\in\mathcal{K}} \in \mathbb{R}_+^K$ represent a vector of expected rewards, and let $\Xi(\delta) := \{(\nu_1, \ldots, \nu_N) \in \mathbb{R}_+^{NK} : |\nu_{n,a_n} - \mu_{n,a_n}| \leq \delta, \forall a_n \in \mathcal{A}'_n, \forall n \in \mathcal{N}\}$. Let Ψ_{T_e} be the set of all possible values that $\bar{\boldsymbol{\mu}}$ can take after T_e rounds of exploration. This set is finite since rewards are binary and we use sample mean rewards to define $\bar{\boldsymbol{\mu}}$.

Fact 1: Under the conditions of Theorem 1, when each user has a Δ-correct estimate of its (channel, estimated best rate) pairs, then $T_m(\frac{1}{8})$ and $||\varphi||_\pi$ are almost surely bounded.

Proof: Since $\Delta < \min_{n,a_n} \mu_{n,a_n}$ and each user has a Δ-correct estimate of its (channel, estimated best rate) pairs (i.e., $\bar{\boldsymbol{\mu}} \in \Xi(\Delta)$), the GoT phase does not include any zero utility actions (for any of the players). This, together with the positivity of ϵ, will guarantee the ergodicity of the Markov chain induced by the GoT utilities. For a fixed $\epsilon > 0$ and $\boldsymbol{\nu} \in \Xi(\Delta)$, let $T_{m,\boldsymbol{\nu},\epsilon}(\frac{1}{8})$ represent the mixing time with an accuracy of $\frac{1}{8}$ and $\pi_{\boldsymbol{\nu},\epsilon}$ represent the stationary distribution of the Markov chain $(M, P^{\boldsymbol{\nu},\epsilon})$. Since all Markov chains are ergodic, $\max_{\boldsymbol{\nu}\in\Xi(\Delta)\cap\Psi_{T_e}} T_{m,\boldsymbol{\nu},\epsilon}(\frac{1}{8}) < \infty$ and $\max_{\boldsymbol{\nu}\in\Xi(\Delta)\cap\Psi_{T_e}} ||\varphi||_{\pi_{\boldsymbol{\nu},\epsilon}} < \infty$. ∎

Appendix B
Proofs of Lemmas

A. Proof of Lemma 2

Although we are selecting both channel and rate, our rate selection process does not affect the channel selection process. Therefore, the channel selection process of our algorithm in the exploration phase is similar to the one in [12]. According to [12, Lemma 1], for every $\eta_1 \in (0, 1)$, $T_{RS}(\eta_1)$ rounds of exploration are needed to ensure that all the players are orthogonalized with probability at least $1 - \eta_1$.

B. Proof of Lemma 3

$\forall \mathbf{a} \in \tilde{\mathcal{A}}$, we have $\hat{\mu}_{n,a_n} = \mu_{n,a_n} + z_{n,a_n}$ where $|z_{n,a_n}| < \Delta$. Thus, $\sum_{n=1}^N \hat{\mu}_{n,a_n^*} = \sum_{n=1}^N (\mu_{n,a_n^*} + z_{n,a_n^*}) > \sum_{n=1}^N \mu_{n,a_n^*} - N\Delta = J_1 - N\Delta$. $\forall \mathbf{a} \in \tilde{\mathcal{A}} \setminus \{\mathbf{a}^*\}$: $\sum_{n=1}^N \hat{\mu}_{n,a_n} = \sum_{n=1}^N (\mu_{n,a_n} + z_{n,a_n}) < \sum_{n=1}^N \mu_{n,a_n} + N\Delta \leq J_2 + N\Delta$. The above equations imply that when $J_2 + N\Delta < J_1 - N\Delta$, we always have $\sum_{n=1}^N \hat{\mu}_{n,a_n^*} > \sum_{n=1}^N \hat{\mu}_{n,a_n}$, $\forall \mathbf{a} \in \tilde{\mathcal{A}} \setminus \{\mathbf{a}^*\}$, which holds since $\Delta < \frac{2(J_1 - J_2)}{5\ N}$. We also have $\sum_{n=1}^N \hat{\mu}_{n,a_n^*} - \operatorname{argmax}_{\mathbf{a}\in\tilde{\mathcal{A}}\setminus\{\mathbf{a}^*\}} \sum_{n=1}^N \hat{\mu}_{n,a_n} \geq J_1 - N\Delta - J_2 - N\Delta \geq \frac{(J_1 - J_2)}{5}$.

C. Proof of Lemma 4

Let $E_{1,n,c}$ be the event in which the best rate for (user, channel) pair (n, c) is not identified correctly after T_e exploration rounds and $E_1 := \cup_n \cup_c E_{1,n,c}$. $\bar{E}_1$ represents the event that the best rates are correctly identified for each (user, channel) pair at the end of the exploration phase. Note that under event $\mathcal{E}$, we have $\forall (n, c) \in \mathcal{N} \times \mathcal{K}$, $\text{Budget}_{n,c}(\tau_{n,c}(T_e)) \geq \lfloor \frac{T_{SS,\min}}{K} \rfloor \geq 8\ H_{\max} \log_2\ R \log(\frac{6NK \log_2\ R}{\eta_2})$. Thus, according to [10, Theorem 4.1], we have:

$$\Pr(E_{1,n,c}|\mathcal{E}) \leq 3 \log_2\ R \exp\Big(- \frac{\text{Budget}_{n,c}(\tau_{n,c}(T_e))}{8 H_{n,c} \log_2\ R}\Big) \leq \frac{\eta_2}{2NK}.$$

Taking a union bound over all n and c, we obtain:

$$\Pr(E_1|\mathcal{E}) \leq \frac{\eta_2}{2} \text{ and } \Pr(\bar{E}_1|\mathcal{E}) \geq 1 - \frac{\eta_2}{2}. \quad (21)$$

Next, we proceed with the remaining part of the proof. Let $E_{2,n}$ be the event that user n does not have Δ-correct estimate of the $(c, \gamma_{n,c,b})$ pairs, for some $c \in \mathcal{K}$ at the end of exploration, and $E_2 := \cup_n E_{2,n}$. Note that $\bar{E}_2$ denotes the event that all users have Δ-correct estimates of all (channel, estimated best rate) pairs. Let B_n be the event in which user n has at least $\kappa := \left\lceil \frac{1}{2\Delta^2} \log(\frac{4NK}{\eta_2}) \right\rceil$ observations of $(c, \gamma_{n,c,b})$ pairs, $\forall c \in \mathcal{K}$. We can write:

$$
\begin{aligned}
&\Pr(E_{2,n}|B_n) \\
&= \Pr(\exists c \in \mathcal{K} : |\hat{\mu}_{n,(c,\gamma_b)} - \mu_{n,(c,\gamma_b)}| \geq \Delta | B_n) \\
&\leq \sum_{c=1}^{K} \Pr(|\hat{\mu}_{n,(c,\gamma_b)} - \mu_{n,(c,\gamma_b)}| \geq \Delta | B_n) \\
&= \sum_{c=1}^{K} \sum_{j=\kappa}^{\infty} \Pr(|\hat{\mu}_{n,(c,\gamma_b)} - \mu_{n,(c,\gamma_b)}| \geq \Delta, V_{n,(c,\gamma_b)} = j | B_n) \\
&= \sum_{c=1}^{K} \sum_{j=\kappa}^{\infty} \Pr(|\hat{\mu}_{n,(c,\gamma_b)} - \mu_{n,(c,\gamma_b)}| \geq \Delta | V_{n,(c,\gamma_b)} = j, B_n) \\
&\qquad \times \Pr(V_{n,(c,\gamma_b)} = j | B_n) \\
&\leq \sum_{c=1}^{K} \sum_{j=\kappa}^{\infty} 2e^{-2j\Delta^2} \Pr(V_{n,(c,\gamma_b)} = j | B_n) \qquad (22) \\
&\leq \sum_{c=1}^{K} 2e^{-2\kappa\Delta^2} \sum_{j=\kappa}^{\infty} \Pr(V_{n,(c,\gamma_b)} = j | B_n) \leq 2Ke^{-2\kappa\Delta^2}
\end{aligned}
$$

where (22) follows from Hoeffding's inequality. Hence, using the union bound we obtain

$$\Pr(E_2 | \cap_n B_n) \leq 2\ NKe^{-2\kappa\Delta^2} \leq \eta_2/2.$$

Next, we will show that B_n happens under event $\mathcal{E}$ for all n. According to Sequential Halving algorithm in [10], the given budget for each (user, channel) pair will be split evenly across $\lceil \log_2\ R \rceil$ elimination stages and rates will be played uniformly within a stage. At the end of a stage, the worst half of the rates will be removed from the rate set. For (user, channel) pair (n, c), we denote the set of remaining rates in stage s by $\mathcal{R}_{n,c,s}$, e.g., $\mathcal{R}_{n,c,0} = \mathcal{R}$ and $\mathcal{R}_{n,c,\lceil \log_2\ R \rceil} = \{\gamma_{n,c,b}\}$, $\forall (n,c) \in \mathcal{N} \times \mathcal{K}$. Similar to analysis in [10], to avoid technicalities and ease the reading, we also assume that R is a power of 2 (i.e., $R = 2^L$). For (user, channel) pair (n, c), we let $C_m := \sum_{s=0}^{\log_2\ R-1} \left\lfloor \frac{\text{Budget}_{n,c}(\tau_{n,c}(T_e))}{\log_2\ R |\mathcal{R}_{n,c,s}|} \right\rfloor$ denote the number of samples in $\hat{\mu}_{n,(c,\gamma_b)}$. Note that

$$
\begin{aligned}
C_m &> \left(\frac{T_{SS,\min}}{K \log_2\ R} \sum_{s=0}^{\log_2\ R-1} \frac{1}{|\mathcal{R}_{n,c,s}|} \right) - \log_2\ R \\
&= \frac{T_{SS,\min}}{K \log_2\ R} \left(\frac{1}{|\mathcal{R}|} + \frac{2}{|\mathcal{R}|} + \frac{4}{|\mathcal{R}|} + \ldots + \frac{2^{L-1}}{|\mathcal{R}|} \right) - \log_2 R \\
&= \frac{T_{SS,\min}}{K \log_2\ R} \frac{R-1}{R} - \log_2\ R.
\end{aligned}
$$

We observe that when $T_{SS,\min} \geq T_2(\eta_2)$, we have $\frac{T_{SS,\min}}{K \log_2\ R} \frac{R-1}{R} - \log_2\ R \geq \kappa$. Thus, we get

$$\Pr(\bar{E}_2 | \mathcal{E}) \geq 1 - \eta_2/2. \qquad (23)$$

TABLE II
NOTATION TABLE

Notation	Explanation
$\mathcal{N}$, N, $\mathcal{K}$, K	Set of users, number of users, set of channels, number of channels.
$\mathcal{R}$, R	Set of transmission rates, number of transmission rates.
$c_n(t)$	Channel selected by user n in round t.
$\gamma_n(t)$, $\gamma^*_{n,c}$	Transmission rate selected by user n in round t, best rate for user n on channel c.
$a_n(t)$, $\mathbf{a}(t)$	Arm selected by user n in round t, strategy profile in round t.
$\tilde{a}_n(t)$, $\tilde{\mathbf{a}}(t)$	Arm selected by user n in round t where the best rate is selected for the chosen channels, strategy profile in round t where all the users select the best rate for their chosen channels.
$\mathbf{a}^*$, J_1	Optimal strategy profile (best assignment), the objective of the best assignment.
$\mathbf{a}'$, J_2	Second best assignment, the objective of the second best assignment.
$\mathcal{A}$	Set of all possible strategy profiles.
$\tilde{\mathcal{A}}$	Set of all possible strategy profiles in which the best rates are selected for the chosen channel of every user.
$\mathcal{N}_i(\mathbf{a})$	Set of users who select channel i in strategy profile $\mathbf{a}$.
$\eta_i(\mathbf{a})$	No-collision indicator of channel i in strategy profile $\mathbf{a}$.
$X_{n,a_n}(t)$	Indicator of the transmission success or failure when user n transmits as the sole user on the channel specified in a_n in round t.
$r_{n,a_n}(t)$	Random reward that user n gets when she transmits as the sole user on the channel in a_n in round t.
θ_{n,a_n}	The transmission success probability when user n transmits as the sole user on the channel specified in a_n.
μ_{n,a_n}	Expected reward that user n gets when she transmits as the sole user on the channel specified in a_n.
$v_n(\mathbf{a}(t))$	Reward obtained by user n in round t.
$g_n(\mathbf{a})$	Expected reward of user n in strategy profile $\mathbf{a}$.
Reg(T), $\overline{\text{Reg}}(T)$	Regret over period T, expected regret over period T.
$\hat{\mu}_{n,a_n}(t)$	Empirical mean reward of user n for (channel, rate) pair a_n up to round t.
$\hat{\mu}_{n,(c,\gamma_{n,c,b})}$	Estimated expected reward of user n for the arm $(c, \gamma_{n,c,b})$ at the end of the exploration phase.
$u_n(\mathbf{a})$	Utility of user n in strategy profile $\mathbf{a}$.
T, T_e, T_g	Total number of rounds, length of the exploration phase, length of the GoT phase.
$\mathcal{A}'_n$	Set of available actions of user n in the GoT phase.
C, D	Content State, discontent State.
Z	State Space, $Z = \prod_n (\mathcal{A}'_n \times \mathcal{M})$, where $\mathcal{M} = \{C, D\}$.
$T_m(\frac{1}{8})$, π	Mixing time of state space Z with an accuracy of $\frac{1}{8}$, stationary distribution of Z.
φ_i	The probability distribution of the state i.
z^*	The optimal state.
$T_{RS,n}$	Number of exploration rounds in which user n is in RS sub-phase.
$T_{SS,n}$	Number of exploration rounds in which user n is in SS sub-phase.
$T_{SS,n,c}$	Number of exploration rounds in which user n selects channel c in SS sub-phase.
$\tau_{n,c}(t)$	Index of the last round up to round t in which user n collided with any other user when her channel is c.
$\gamma_{n,c,b}$	Estimated best rate for (user, channel) pair (n, c) at the end of the exploration phase.
Budget$_{n,c}(\tau_{n,c}(t))$	Budget of (user, channel) pair (n,c) in round t.
$\mathcal{R}_{n,c,s}$	Set of remaining rates in elimination stage s.
$\mathcal{G}$	Set of rounds in the GoT phase.

Combining (21) and (23) by a union bound, we get $\Pr(\bar{E}_1 \cap \bar{E}_2 | \mathcal{E}) \geq 1 - \eta_2$.

APPENDIX C
NOTATION TABLE

See Table II.

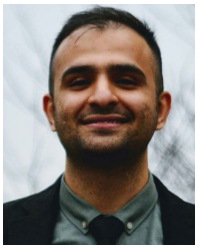
ACKNOWLEDGMENT

The authors would like to thank anonymous reviewers for their comments which significantly improved this article.

REFERENCES

[1] R. Combes and A. Proutiere, "Dynamic rate and channel selection in cognitive radio systems," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 5, pp. 910–921, May 2015.

[2] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Mach. Learn.*, vol. 47, no. 2, pp. 235–256, 2002.
[3] R. Combes, J. Ok, A. Proutiere, D. Yun, and Y. Yi, "Optimal rate sampling in 802.11 systems: Theory, design, and implementation," *IEEE Trans. Mobile Comput.*, vol. 18, no. 5, pp. 1145–1158, May 2019.
[4] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, pp. 285–294, Dec. 1933.
[5] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Adv. Appl. Math.*, vol. 6, no. 1, pp. 4–22, Mar. 1985.
[6] I. Bistritz and A. Leshem, "Distributed multi-player bandits-a game of thrones approach," in *Proc. 32nd Conf. Neural Inf. Process. Syst.*, pp. 7222–7232, 2018.
[7] K. Liu and Q. Zhao, "Distributed learning in multi-armed bandit with multiple players," *IEEE Trans. Signal Process.*, vol. 58, no. 11, pp. 5667–5681, Nov. 2010.
[8] A. Anandkumar, N. Michael, A. K. Tang, and A. Swami, "Distributed algorithms for learning and cognitive medium access with logarithmic regret," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 4, pp. 731–745, Apr. 2011.
[9] I. Bistritz and A. Leshem, "Game of thrones: Fully distributed learning for multiplayer bandits," *Math. Oper. Res.*, vol. 46, no. 1, pp. 159–178, Feb. 2021.
[10] Z. Karnin, T. Koren, and O. Somekh, "Almost optimal exploration in multi-armed bandits," in *Proc. 30th Int. Conf. Mach. Learn.*, vol. 28, May 2013, pp. 1238–1246.
[11] J. Rosenski, O. Shamir, and L. Szlak, "Multi-player bandits—A musical chairs approach," in *Proc. 33rd Int. Conf. Mach. Learn.*, Jun. 2016, pp. 155–163.
[12] M. K. Hanawal and S. J. Darak, "Multi-player bandits: A trekking approach," 2018, *arXiv:1809.06040*. [Online]. Available: http://arxiv.org/abs/1809.06040
[13] Y. Gai, B. Krishnamachari, and R. Jain, "Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation," in *Proc. IEEE Symp. New Frontiers Dyn. Spectr. (DySPAN)*, Apr. 2010, pp. 1–9.
[14] D. Kalathil, N. Nayyar, and R. Jain, "Decentralized learning for multiplayer multiarmed bandits," *IEEE Trans. Inf. Theory*, vol. 60, no. 4, pp. 2331–2345, Apr. 2014.
[15] N. Nayyar, D. Kalathil, and R. Jain, "On regret-optimal learning in decentralized multiplayer multiarmed bandits," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 1, pp. 597–606, Mar. 2018.
[16] J. Xu, C. Tekin, S. Zhang, and M. van der Schaar, "Distributed multi-agent online learning based on global feedback," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2225–2238, May 2015.
[17] C. Tekin and M. Liu, "Online learning in decentralized multi-user spectrum access with synchronized explorations," in *Proc. MILCOM IEEE Mil. Commun. Conf.*, Oct. 2012, pp. 1–6.
[18] M. Bande and V. V. Veeravalli, "Multi-user multi-armed bandits for uncoordinated spectrum access," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, Feb. 2019, pp. 653–657.
[19] L. Besson and E. Kaufmann, "Multi-player bandits revisited," in *Proc. Algorithmic Learn. Theory*, vol. 83, Apr. 2018, pp. 56–92.
[20] G. Lugosi and A. Mehrabian, "Multiplayer bandits without observing collision information," 2018, *arXiv:1808.08416*. [Online]. Available: http://arxiv.org/abs/1808.08416
[21] E. Boursier and V. Perchet, "SIC-MMAB: Synchronisation involves communication in multiplayer multi-armed bandits," in *Proc. 33rd Conf. Neural Inf. Process. Syst.*, 2019, pp. 12048–12057.
[22] D. Martínez-Rubio, V. Kanade, and P. Rebeschini, "Decentralized cooperative stochastic bandits," in *Proc. Adv. Neural Inf. Proces. Syst.*, 2019, pp. 4529–4540.
[23] P.-A. WANG, A. Proutiere, K. Ariu, Y. Jedra, and A. Russo, "Optimal algorithms for multiplayer multi-armed bandits," in *Proc. Int. Conf. Artif. Intell. Statist.* vol. 108, Aug. 2020, pp. 4120–4129.
[24] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Chelmsford, MA, USA: Courier Corporation, 1998.
[25] S. M. Zafaruddin, I. Bistritz, A. Leshem, and D. Niyato, "Multiagent autonomous learning for distributed channel allocation in wireless networks," in *Proc. IEEE 20th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2019, pp. 1–5.
[26] S. M. Zafaruddin, I. Bistritz, A. Leshem, and D. Niyato, "Distributed learning for channel allocation over a shared spectrum," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2337–2349, Oct. 2019.
[27] H. Tibrewal, S. Patchala, M. K. Hanawal, and S. J. Darak, "Distributed learning and optimal assignment in multiplayer heterogeneous networks," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, Apr. 2019, pp. 1693–1701.
[28] A. Mehrabian, E. Boursier, E. Kaufmann, and V. Perchet, "A practical algorithm for multiplayer bandits when arm means vary among players," in *Proc. Int. Conf. Artif. Intell. Statist.*, vol. 108, Aug. 2020, pp. 1211–1221.
[29] P. Alatur, K. Y. Levy, and A. Krause, "Multi-player bandits: The adversarial case," *J. Mach. Learn. Res.*, vol. 21, no. 77, pp. 1–23, 2020.
[30] S. Bubeck, Y. Li, Y. Peres, and M. Sellke, "Non-stochastic multi-player multi-armed bandits: Optimal rate with collision information, sublinear without," 2019, *arXiv:1904.12233*. [Online]. Available: http://arxiv.org/abs/1904.12233
[31] K. Jamieson and R. Nowak, "Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting," in *Proc. 48th Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2014, pp. 1–6.
[32] A. Garivier and E. Kaufmann, "Optimal best arm identification with fixed confidence," in *Proc. Conf. Learn. Theory*, 2016, pp. 998–1027.
[33] D. Russo, "Simple Bayesian algorithms for best arm identification," in *Proc. Conf. Learn. Theory*, 2016, pp. 1417–1418.
[34] I. Bistritz, T. Baharav, A. Leshem, and N. Bambos, "My fair bandit: Distributed learning of max-min fairness with multi-player bandits," in *Proc. Int. Conf. Mach. Learn.*, Nov. 2020, pp. 930–940.



Alireza Javanmardi received the B.Sc. degree in electrical engineering from Shiraz University, Shiraz, Iran, in 2017, and the M.Sc. degree in electrical and electronics engineering from Bilkent University, Ankara, Turkey, in 2020. He is currently pursuing the Ph.D. degree with the Department of Computer Science, University of Tübingen, Tübingen, Germany. His research interests include cognitive communications, multiagent learning, and multi-armed bandit problems.

Muhammad Anjum Qureshi (Member, IEEE) received the B.Sc. degree in electrical and electronics engineering from UET, Taxila, Pakistan, in 2005, the master's degree from CASE, Islamabad, Pakistan, in 2010, and the Ph.D. degree from the Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey, in 2020, under the supervision of Dr. Cem Tekin. His research interests include machine learning, wireless communications, and multi-armed bandit problems. He received the Alper Atalay Award for Best Paper in IEEE-SIU 2017 and the Third Best Student Paper Award in IEEE-SIU 2018.

Cem Tekin (Senior Member, IEEE) received the B.Sc. degree in electrical and electronics engineering from Middle East Technical University, Ankara, Turkey, in 2008, and the M.S.E. degree in electrical engineering: systems, the M.S. degree in mathematics, and the Ph.D. degree in electrical engineering: systems from the University of Michigan, Ann Arbor, MI, USA, in 2010, 2011, and 2013, respectively. From February 2013 to January 2015, he was a Post-Doctoral Scholar with the University of California, Los Angeles, CA, USA. He is currently an Associate Professor with the Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey. His research interests include cognitive communications, reinforcement learning, multiarmed bandit problems, and multiagent systems. He received numerous awards, including the Fred W. Ellersick Award for the Best Paper in MILCOM 2009 and the Distinguished Young Scientist (BAGEP) Award of the Science Academy Association of Turkey in 2019.