FULLY DISTRIBUTED BANDIT ALGORITHM FOR THE JOINT CHANNEL AND RATE SELECTION PROBLEM IN HETEROGENEOUS COGNITIVE RADIO NETWORKS

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF

MASTER OF SCIENCE

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

By

Alireza Javanmardi

December 2020

Fully distributed bandit algorithm for the joint channel and rate selection problem in heterogeneous cognitive radio networks
By Alireza Javanmardi
December 2020

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Cem Tekin(Advisor)

Sinan Gezici

Elif Uysal

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

To my mother

ABSTRACT

FULLY DISTRIBUTED BANDIT ALGORITHM FOR THE JOINT CHANNEL AND RATE SELECTION PROBLEM IN HETEROGENEOUS COGNITIVE RADIO NETWORKS

Alireza Javanmardi
M.S. in Electrical and Electronics Engineering
Advisor: Cem Tekin
December 2020

We consider the problem of the distributed sequential channel and rate selection in cognitive radio networks where multiple users choose channels from the same set of available wireless channels and pick modulation and coding schemes (corresponds to transmission rates). In order to maximize the network throughput, users need to be cooperative while communication among them is not allowed. Also, if multiple users select the same channel simultaneously, they collide, and none of them would be able to use the channel for transmission. We rigorously formulate this resource allocation problem as a multi-player multi-armed bandit problem and propose a decentralized learning algorithm called Game of Thrones with Sequential Halving Orthogonal Exploration (GoT-SHOE). The proposed algorithm keeps the number of collisions in the network as low as possible and performs almost optimal exploration of the transmission rates to speed up the learning process. We prove our learning algorithm achieves a regret with respect to the optimal allocation that grows logarithmically over rounds with a leading term that is logarithmic in the number of transmission rates. We also propose an extension of our algorithm which works when the number of users is greater than the number of channels. Moreover, we discuss that Sequential Halving Orthogonal Exploration can indeed be used with any distributed channel assignment algorithm and enhance its performance. Finally, we provide extensive simulations and compare the performance of our learning algorithm with the state-of-the-art which demonstrates the superiority of the proposed algorithm in terms of better system throughput and lower number of collisions.

Keywords: Cognitive radio, multi-armed bandits, decentralized algorithms, regret bounds.

ÖZET

HETEROJEN BİLİŞSEL RADYO AĞLARINDA MÜŞTEREK KANAL VE ORAN SEÇİMİ PROBLEMİ İÇİN TÜMÜYLE MERKEZİ OLMAYAN HAYDUT ALGORİTMASI

Alireza Javanmardi
Elektrik ve Elektronik Mühendisliği, Yüksek Lisans
Tez Danışmanı: Cem Tekin
Aralık 2020

Bilişsel radyo ağlarında ağ çıktısını enbüyüklemek için her kullanıcının kablosuz kanal, modülasyon ve kodlama şeması (aktarım hızı) seçtiği, merkezi olmayan dinamik oran ve kanal seçimi problemi ele alınmıştır. Kullanıcıların işbirliği yaptığı, ancak, kendi aralarında koordinasyon ve haberleşme yapmadığı ve sistemdeki kullanıcı sayısının bilinmediği varsayılmıştır. Bu problem çoklu-oyunculu çok-kollu haydut olarak modellenmiş ve Sıralı İkiye Bölmeli Dikey Keşifli Taht Oyunları (GoT-SHOE) isimli merkezi olmayan öğrenme algoritması önerilmiştir. Önerilen algoritma oyundaki çarpışmaları mümkün olduğunca az tutar ve hızlı öğrenmek için aktarım hızının neredeyse en uygun keşfini yapar. Öğrenme algoritmamızın en uygun tahsise göre pişmanlığının artışının zamanda logaritmik olduğu kanıtlanmıştır. Ayrıca kullanıcı sayısının kanal sayısından büyük olduğu durumlarda algoritmamızın nasıl çalıştırılabileceği de incelenmiştir. Ek olarak, Sıralı İkiye Bölmeli Dikey Keşif metodunun herhangi bir merkezi olmayan kanal ataması algoritmasıyla nasıl kullanılabileceği ve bu durumda sunduğu performans artışı tartışılmıştır. Son olarak, öğrenme algoritmamızın başarısı simülasyonlar üzerinden en gelişmiş diğer metotlarla karşılaştırılmış ve algoritmamızın başarılı iletilen veri miktarını yüksek oranda arttırdığı ve çarpışma sayılarını azalttığı gösterilmiştir.

Anahtar sözcükler: Bilişsel radyo ağları, çok kollu haydut problemleri, merkezi olmayan algoritmalar, pişmanlık sınırları.

Acknowledgement

First of all, I would like to thank my advisor, Assist. Prof. Dr. Cem Tekin, for giving me the opportunity to work under his supervision and also for his support and encouragement during this M.Sc. at Bilkent University.

I would like to acknowledge the thesis jury members, Prof. Dr. Sinan Gezici and Prof. Dr. Elif Uysal for their valuable time and insightful feedback. It was very kind of them participating in my thesis despite their busy schedule.

I wish to show my deepest gratitude to my beloved friends Daniyal, Soheil, and Mahsa. I can not imagine how my life would be without them for the past two years. We had a delightful, enjoyable, and memorable time together and I hope this friendship lasts forever.

I am grateful to the rest of my friends, both Iranian and internationals who made my life in Turkey much more pleasant. I am also thankful to the members of our research group (CYBORG), Alp, Andi, and Kubilay for the fantastic conversations we had. Especially, I would like to thank Dr. Muhammad Anjum Qureshi who was like an elder brother to me.

Last but not the least, I am indebted to my father Abbas, my sister Farnaz and my brother Armin. To me, they are the most valuable persons in the world and one of the most important reasons why I am pursuing this path. Likewise, I would like to thank the other family members of mine.

This work was supported by the Scientific and Technological Research Council of Turkey (TUBITAK) under Grant 116E229.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The multi-armed bandit (MAB) problem is a type of sequential optimization problem, where in each round, a decision-maker pulls an arm (action) among multiple possible arms (action space) enforced by a specific policy and receives a random reward [1, 2]. The objective of the decision-maker is to maximize the expected cumulative reward while the arms' reward distributions are not known in advance. In each round, the decision-maker has to either explore the action space in order to enhance her belief about the barely-known actions or exploit her knowledge to select the empirically optimal action. In order to maximize the expected cumulative reward, the decision-maker needs to find a balance between exploration and exploitation [3].

The multi-player multi-armed bandit (MPMAB) problem is an extension to the MAB problem, where in each round, multiple decision-makers (players) select arms simultaneously from a shared action space. In such a problem, the objective is to maximize the sum of players' expected cumulative rewards. Below we give a detailed review of the different types of the MPMAB problem:

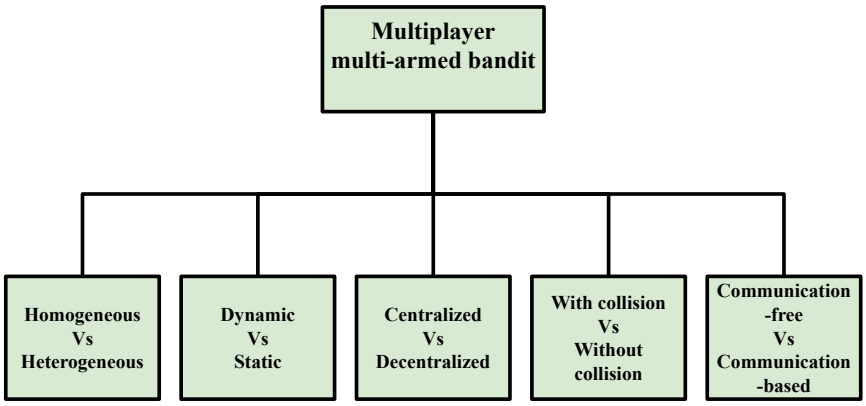


Figure 1.1: Variants of the MPMAB problem.

- **Centralized vs. Decentralized:** In the centralized MPMAB problem, a central learner selects the arms of the players. This scenario can be viewed as a single-player MAB problem with multiple plays [4]. Such a central learner does not exist in the decentralized MPMAB and each player has to select her arm individually [5].

- **Communication-based vs. communication-free:** Players might be able to exchange messages to share their information and achieve coordination [6]. However, in many cases, direct communication among players is not allowed [7].

- **Homogeneous vs. heterogeneous:** In the homogeneous setting, the expected reward of an arm is the same for all the players [8, 9], while in the heterogeneous setting, it may be different for different players [10, 11].

- **With collision vs. without collision:** Many works assume that multiple players can not use the same arm simultaneously and if they do so, they collide and all of them get zero rewards [7,8,10]. In the collision-free setting, however, multiple players can obtain non-zero rewards from selecting the same arm at the same time [12, 13].

- **Dynamic vs. static:** In the dynamic setting, the players may enter and leave throughout the game [8,14]. Obviously, there must be some restriction on the entering and leaving rates of the players in order to guarantee the

performance of such algorithms. In the static setting, however, the players start the game all at once and remain there till the end [5].

Similar to the single-player MAB and based on the type of the reward generation, MPMAB can be categorized into stochastic MPMAB (where the series of rewards of each arm are drawn from a specific and a priori unknown distribution [15]) and adversarial MPMAB (where no statistical assumptions are made about the reward generation process [16]).

1.1 Our Contribution

In this thesis, we rigorously formulate the resource allocation problem in the cognitive radio network (CRN) as an MPMAB problem. Two classes of users are available in such a network: primary users (PUs) who are the owner of the frequency bands (channels), and secondary users (SUs) who may opportunistically use those channels when they are not occupied by PUs.[1] It is assumed that SUs are able to use a geolocation database to get a list of channels free from PUs [17].

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.

In this MPMAB problem, (channel, rate) pairs are considered as *arms* and SUs as *players* or *learners*. In each round, each player selects a (channel, rate) pair for a packet transmission, and then, receives a random reward which indicates that either the transmission is successful or unsuccessful. The expected reward of each (channel, rate) pair is given as the chosen rate times the packet successful

[1] These parts of the spectrum that are not used by PUs are known as *white spaces*.

transmission probability, which is also referred to as the *throughput*. We target to maximize the overall system throughput, which is calculated as the summation of the throughputs of the individual users.

We list the challenges faced in the above multi-user resource allocation problem as follows:

(i) Having a central controller that assigns (channel, rate) pairs to the users induces significant costs in terms of time and energy which in turn is not applicable in CRN. Moreover, in realistic scenarios, SUs may be unable or unwilling to communicate with each other.

(ii) While players are aware of their action space (i.e., the set of channels and the set of transmission rates), the number of SUs in the network is unknown.

(iii) 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.

(iv) Inspired by the classical ALOHA protocol, multiple SUs are not able to use the same channel at the same time.

As a result, we need to design a decentralized communication-free algorithm that enables SUs to jointly maximize the overall system throughput in such a heterogeneous setting. In addition, since the 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. Hence, it is necessary for any such algorithms to keep the number of collisions as low as possible.

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. Maximizing the overall system throughput is equivalent to minimizing the regret.

Lastly, as the number of (channel, rate) pairs increases, learning becomes more challenging because there are more options to explore. Since the number of SUs is unknown, each user has to learn the quality of all the channels adequately. However, this is not the case for the transmission rates and each user is merely required to know the optimal rate for each channel. In the single-user case exploring all the rates sufficiently enough for any channel results in regret that scales linearly in the number of rates, while sequential elimination of the suboptimal rates results in regret that is logarithmic in the number of rates [18].

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 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.

1.2 Related Works

1.2.1 Related Works with MPMAB

1.2.1.1 homogeneous setting:

A centralized MPMAB problem where the decisions of the users are made by a central agent is studied in [4, 19, 20]. A decentralized setting where users are allowed to communicate with each other is considered in [5,6,21], while [5,7,8,14] assume a fully distributed scenario. In particular, in [14], exploration is done by employing an orthogonalization technique which orthogonalizes the players with respect to the channels in order to minimize the number of collisions during the learning process.

Authors in [9] propose a decentralized UCB-based algorithm that requires the knowledge of the number of users in the network. Unlike the mentioned works, the case where the players are not provided with collision feedback is considered in [15]. While all the previous works do not allow multiple players to use the same channel simultaneously, [12, 13] consider the case when multiple players can obtain non-zero rewards from the same arm at the same time. All of the aforementioned works focus on the stochastic MPMAB whereas the adversarial case is addressed in [16, 22].

1.2.1.2 heterogeneous setting:

A decentralized heterogeneous MPMAB problem was introduced in [6] and enhanced in [23]. In both works, however, it is assumed that users are able to communicate their selected arms with each other. This assumption was relaxed in [24] where users are merely required to be able to sense all the channels without knowing which channel was selected by whom. A fully-distributed algorithm, known as *Game of Thrones* (GoT), is proposed in [10] and [25]. This algorithm solves a special case of the well-known distributed assignment problem [26] by

using collision and reward feedback. An improved algorithm with a better convergence time than GoT is proposed in [11, 27]. 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.

1.2.2 Related Works with best arm identification in MAB

Apart from MPMAB, 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., [18, 28–30]). Specifically, authors in [18] 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.

1.2.3 Comparison with the related works

This work can be seen as an extension of [17] to the decentralized, communication-free, and heterogeneous multi-user network where multiple users can not use the same channel at the same time. Also, it can be considered as an extension of [25] to the case where the outcome of transmission for each user depends on the chosen rate as well as the selected channel. The proposed algorithm learns channels and optimal rates together while keeping the number of collisions as low as possible by using the orthogonalization idea proposed in [14].

Moreover, as our reward signal is binary, we only require 1-bit feedback

Table 1.1: Comparison of Our Work with Prior Works

Property/Algorithm	Musical Chairs [8]	Trekking approach [14]	GoT [25]	**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)

(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 [17,31]. The differences between our work and the related works are summarized in Table 1.1.

The rest of the thesis is organized as follows. The MPMAB problem is defined in Chapter 2. The learning algorithm is proposed in Chapter 3, and its analysis is given in Chapter 4. In Chapter 5, we propose an extension to our algorithm for the case that the number of users is greater than the number of channels and also we integrate the proposed learning algorithm into an existing state-of-the-art MPMAB algorithm. Numerical results for the proposed scheme are provided in Chapter 6, followed by concluding remarks in Chapter 7.

Chapter 2

Problem Formulation

2.1 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 [8, 14], 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 [8, 14, 25], we focus on the case where $K \geq N$ in the remainder of this Chapter.[2] An example of channel allocation problem with $N = 5$ and $K = 5$ is provided in Figure 2.1.

[1] For a positive integer N, $[N] := \{1, \ldots, N\}$.

[2] The case where $N > K$ is discussed in Chapter 5

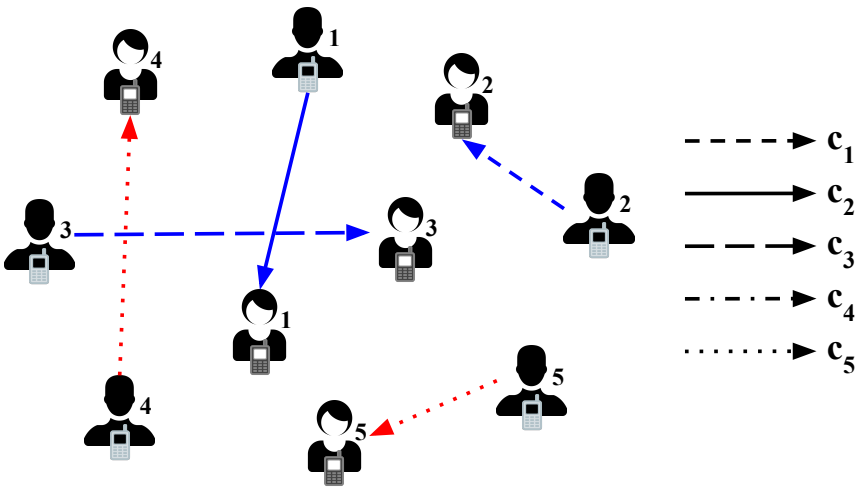


Figure 2.1: The system model of a network with 5 users ($N = 5$) and 5 channels ($K = 5$). Different channels are represented by different dash types while their colors determine whether they are collision-free (blue) or not (red). Here, users 1, 2, and 3 transmit over channels 2, 1, and 3 respectively. Users 4 and 5 face collision on channel 5 while channel 4 is unused.

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.

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 strategy profile $\mathbf{a}$ as:

$$\eta_i(\mathbf{a}) = \begin{cases} 0 & |\,\mathcal{N}_i(\mathbf{a})\,| > 1 \\ 1 & \text{otherwise} \end{cases} \tag{2.1}$$

where $\mathcal{N}_i(\mathbf{a}) := \{n : c_n = i\}$ is the set of users who select channel i in strategy profile $\mathbf{a}$.

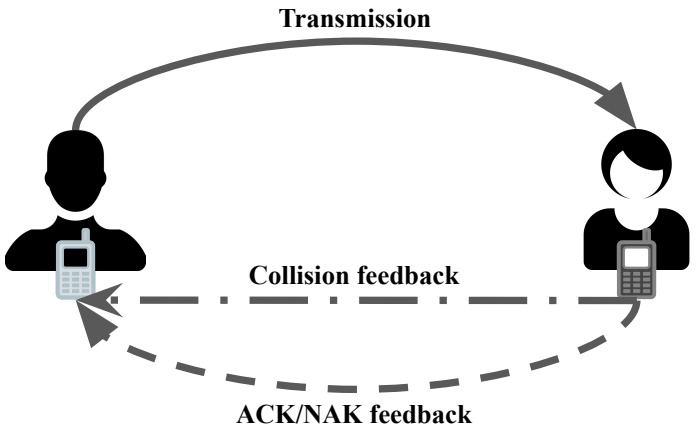


Figure 2.2: The transmitter transmits a packet on the selected channel with the selected rate and receives two feedback after that.

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:

$$v_n(\mathbf{a}(t)) := r_{n,a_n(t)}(t)\eta_{c_n(t)}(\mathbf{a}(t)) \ . \tag{2.2}$$

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 (see Figure 2.2). 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

$$\gamma^*_{n,c} := \underset{\gamma \in \mathcal{R}}{\operatorname{argmax}}\, \mu_{n,(c,\gamma)} \tag{2.3}$$

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 [18], we define

$$H_{n,c} := \max_{\gamma \neq \gamma^*_{n,c}} \frac{I_\gamma}{(\mu_{n,(c,\gamma^*_{n,c})} - \mu_{n,(c,\gamma)})^2} \tag{2.4}$$

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.

2.2 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 (2.2). The (pseudo) regret over period T is defined as:

$$\mathrm{Reg}(T) := \sum_{t=1}^{T} \sum_{n=1}^{N} g_n(\mathbf{a}^*) - \sum_{t=1}^{T} \sum_{n=1}^{N} g_n(\mathbf{a}(t)) \tag{2.5}$$

where

$$\mathbf{a}^* := \underset{\mathbf{a} \in \mathcal{A}}{\operatorname{argmax}} \sum_{n=1}^{N} g_n(\mathbf{a}). \tag{2.6}$$

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* $[c_n]_{n\in[N]}$*.*

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

$$\mu_{n,(c_n,\gamma)} \leq \mu_{n,(c_n,\gamma^*_{c_n})}, \quad \forall \gamma \in \mathcal{R} \ . \tag{2.7}$$

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 (2.7), we obtain that $g(\mathbf{a}) \leq g(\tilde{\mathbf{a}})$. □

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

Chapter 3

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 [25] 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 [25]. Unlike [25], 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

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$(T - T_e - T_g, c_n^*, \gamma_{n,c_n^*,b})$

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.

3.1 Sequential Halving Orthogonal Exploration

This phase consists of T_e number of rounds. In comparison to [25] 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 [14] and [18]. Flowchart and pseudocode of this phase are given in Figure 3.1 and Algorithm 2 respectively.

3.1.1 Channel Allocation

Channel allocation is inspired from [14], 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

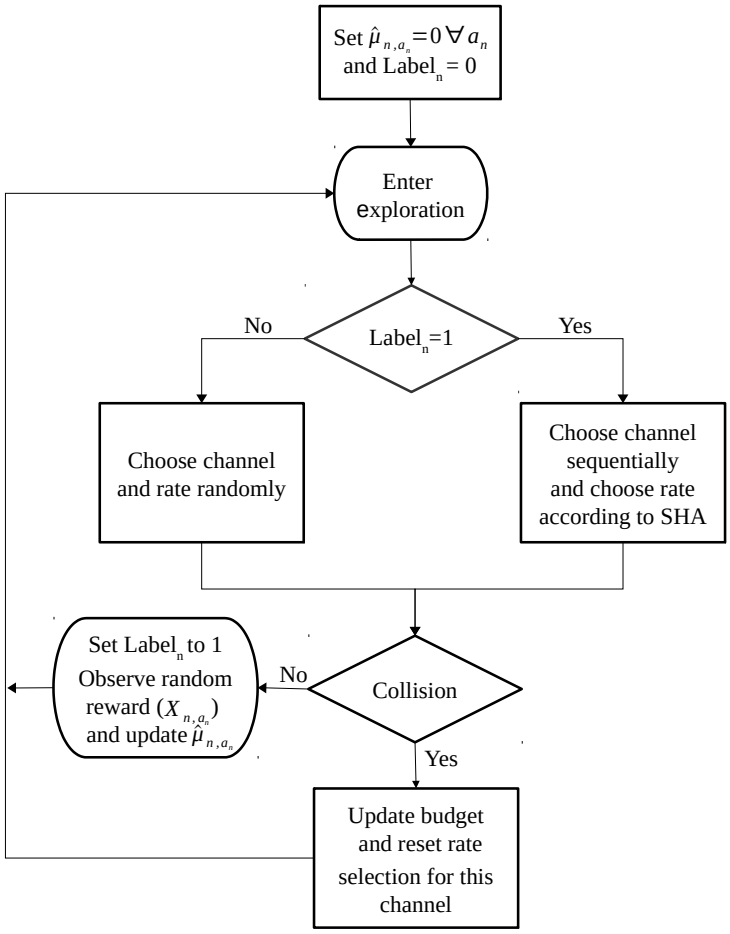


Figure 3.1: Flowchart of Sequential Halving Orthogonal Exploration.

rounds, she simply selects channels sequentially in each round, i.e., $c_n(t+1) = c_n(t) \mod K+1$. Take a look at Figure 3.2 to see how this orthogonalization idea works.

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:

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

$$\sum_{c \in \mathcal{K}} T_{SS,n,c} = T_{SS,n}. \tag{3.1}$$

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	c_3	c_4	c_1	c_2	c_3	c_4	c_1	c_2
	c_4	c_2	c_1	c_3	c_2	c_3	c_4	c_1
	c_4	c_4	c_2	c_3	c_4	c_1	c_2	c_3
	c_1	c_2	c_3	c_4	c_1	c_2	c_3	c_4

Figure 3.2: An illustration of the player orthogonalization in a network with 4 users ($N = 4$) and 4 channels ($K = 4$). While users 1 and 4 enter the SS sub-phase in round 1, users 2 and 3 find collision-free channels in rounds 5 and 3 respectively.

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

$$\left\lfloor \frac{T_{SS,n}}{K} \right\rfloor \leq T_{SS,n,c} \leq \left\lceil \frac{T_{SS,n}}{K} \right\rceil. \tag{3.2}$$

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 afterward. Compared to the random exploration used in [25], 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.

3.1.2 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 the RS sub-phase, she will select 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 [18].

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:

$$\text{Budget}_{n,c}(\tau_{n,c}(t)) = \left\lfloor \frac{T_e - \tau_{n,c}(t)}{K} \right\rfloor . \tag{3.3}$$

According to Sequential Halving algorithm in [18], 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., $\mathcal{R}_{n,c,0} = \mathcal{R},\ \forall (n, c) \in \mathcal{N} \times \mathcal{K}$.

Meanwhile, user n might face collisions with other users that are in the RS sub-phase. If such an 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.[1] Let $\gamma_{n,c,b}$[2] 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.

3.2 Game of Thrones (GoT)

The pseudocode of this phase is given in Algorithm 4. This phase consists of T_g number of rounds. Similar to the GoT phase in [25], the strategy space of this phase consists of K channels with their corresponding estimated best rates. The

[1] 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.

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

empirical estimates are used as deterministic utilities for the GoT dynamics, i.e.,

$$u_n(\mathbf{a}) := \hat{\mu}_{n,(c_n,\gamma_{n,c_n,b})}\eta_{c_n}(\mathbf{a}). \tag{3.4}$$

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.

3.3 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.

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**

$c_n(t+1) \sim U(1, \ldots, K), \quad \gamma_n(t+1) \sim U(1, \ldots, R)$

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

$t \leftarrow t + 1$

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 $\bar{\mu}_n, \bar{\gamma}_n^*$

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 $\gamma_n(t+1), \mathcal{R}_{n,c_n(t+1),s}$

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 $t \leq T_g$ **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:

$$[\bar{c}_n, C/D] \rightarrow \begin{cases} [c_n, C] & \frac{u_n}{u_{n,max}} \epsilon^{u_{n,max} - u_n} \\ [c_n, D] & 1 - \frac{u_n}{u_{n,max}} \epsilon^{u_{n,max} - u} \end{cases}$$

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
$c_n^* = \text{argmax}_{i \in \mathcal{K}} F_n(i, \gamma_{n,i,b})$
return c_n^*

Algorithm 5 Exploitation (EXP)

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

Chapter 4

Regret Analysis

In this chapter, we analyze the regret of GoT-SHOE.

4.1 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 $P^{\epsilon}_{\omega} = P^{\epsilon,\bar{\boldsymbol{\mu}}(\omega)}$ 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 [25, Theorem 2], if for a given $\omega \in \Omega$ and $\epsilon \in (0,1)$, $\phi \geq \sum_n u_{n,max,\omega} - J_1$, and the Markov chain (Z, P^ϵ_ω) is ergodic, 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 [25, 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 [25, Lemma 5], for this small enough $\epsilon_\omega > 0$ we have:

$$P_{g,\omega} := \Pr\left(\sum_{t\in\mathcal{G}} \mathbb{1}(z(t) = z^*_\omega) \leq (1-\rho)\pi_{z^*_\omega,\omega} T_g |\bar{\boldsymbol{\mu}}(\omega)\right) \leq B_0 ||\varphi||_{\pi_\omega} e^{-\frac{\rho^2 \pi_{z^*_\omega,\omega} T_g}{72 T_{m,\omega}(\frac{1}{8})}} \tag{4.1}$$

where $T_{m,\omega}(\frac{1}{8})$ is the mixing time of $(Z, P^{\epsilon_\omega}_\omega)$ 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.,

$$\varphi_i = \begin{cases} \frac{1}{K^N} & \text{if } i = [\mathbf{a}, C^N] \text{ for some } \mathbf{a} \in \mathcal{A}' \\ 0 & \text{otherwise.} \end{cases} \tag{4.2}$$

For any $i \in \mathcal{N}$, let $v_i = \{v_{i,j}\}_{j\in\mathcal{K}} \in \mathbb{R}_+^K$ represent a vector of expected rewards. We make the following assumption throughout the analysis in order to ensure that $T_m(\frac{1}{8})$ and $||\varphi||_\pi$ are almost surely bounded.

Assumption 1. *Let* $\Xi(\delta) := \{(v_1, \ldots, v_N) \in \mathbb{R}_+^{NK} : |v_{n,a_n} - \mu_{n,a_n}| \leq \delta, \forall a_n \in \mathcal{A}'_n, \forall n \in \mathcal{N}\}$ *be a compact set. We assume that there exists* $\Delta_0 > 0$ *such that for any* $\boldsymbol{v} := (v_1, \ldots, v_N) \in \Xi(\Delta_0)$ *and* $\epsilon \in (0,1)$*, the Markov chain* $(Z, P^{\epsilon,\boldsymbol{v}})$ *is ergodic.*

Note that GoT dynamics may fail to converge when ergodicity of the induced Markov chain does not hold. In addition, $T_{m,\omega}(\frac{1}{8})$ may exhibit divergent behavior around these extreme cases. Assumption 1 ensures that when expected rewards for (channel, channel's estimated best rate) pairs are accurately estimated by all

users in the exploration phase, GoT will operate far away from these extreme cases. Note that the positivity of expected rewards ensures existence of such Δ_0. We can simply set $\Delta_0 > 0$, such that $\Delta_0 < \min_{n,a_n} \mu_{n,a_n}$.

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. *Let* $\Delta := \min\{\Delta_0, \frac{2(J_1-J_2)}{5N}\}$ *and* $\Xi := \Xi(\Delta)$. *Conditioned on the event* $\bar{\boldsymbol{\mu}} \in \Xi \cap \Psi_{T_e}$, $T_m(\frac{1}{8})$ *and* $||\varphi||_\pi$ *are almost surely bounded.*

Proof. For a fixed $\epsilon > 0$ and $\boldsymbol{v} \in \Xi$, let $T_{m,\boldsymbol{v},\epsilon}(\frac{1}{8})$ represent the mixing time with an accuracy of $\frac{1}{8}$ and $\pi_{z^*,\boldsymbol{v},\epsilon}$ represent the stationary probability of state z^* of the Markov chain $(M, P^{\boldsymbol{v},\epsilon})$. Let $\epsilon(\boldsymbol{v}) > 0$ be such that $(1-\rho)\pi_{z^*,\boldsymbol{v},\epsilon(\boldsymbol{v})} > \frac{1}{2}$ is satisfied (see [25, Eqns. A.20∼A.21]). Such an $\epsilon(\boldsymbol{v})$ exists since according to Lemma 3 when $\Delta < \frac{2(J_1-J_2)}{5N}$, then the unique optimal allocation is also $\mathbf{a}^*$ in the GoT phase and the gap between the best and second best allocation in the GoT phase is at least $\frac{(J_1-J_2)}{5}$. Let $\epsilon_{\min} = \min_{\boldsymbol{v}\in\Xi\cap\Psi_{T_e}} \epsilon(\boldsymbol{v}) > 0$ since $\Xi \cap \Psi_{T_e}$ is finite. It can be shown via a coupling argument that $T_{m,\boldsymbol{v},\epsilon_{\min}}(\frac{1}{8}) \leq \ln 8/(\epsilon_{\min}^{\phi}/(K-1))^{2N}$ for all $\boldsymbol{v} \in \Xi \cap \Psi_{T_e}$. Therefore, $\max_{\boldsymbol{v}\in\Xi\cap\Psi_{T_e}} T_{m,\boldsymbol{v},\epsilon_{\min}}(\frac{1}{8}) \leq \ln 8/(\epsilon_{\min}^{\phi}/(K-1))^{2N}$. Since the Markov chain is ergodic for $\boldsymbol{v} \in \Xi$, we also have $||\varphi||_{\pi_{z^*,\boldsymbol{v},\epsilon_{\min}}} < \infty$. Therefore, $\max_{\boldsymbol{v}\in\Xi\cap\Psi_{T_e}} ||\varphi||_{\pi_{z^*,\boldsymbol{v},\epsilon_{\min}}} < \infty$. □

Finally, let $A := B_0 \times ||\varphi||_\pi$.

4.2 Main Result

Our main result is given in the following theorem.

Theorem 1. *Fix* $\rho \in (0, 1/2)$. *For all* $\Delta < \min\{\Delta_0, \frac{2(J_1-J_2)}{5N}\}$, $\phi \geq N(1+\Delta)-J_1$,[1] *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*

[1] *Since* N *and* J_1 *are not known to the users, the upper bound* $\phi \geq K$ *will also work.*

$$
\begin{aligned}
T_e \geq & \left\lceil \frac{\log(\eta/3K)}{\log(1-1/4K)} \right\rceil \\
& + \max\left(\left\lceil 8KH_{\max} \log_2 R \log(\frac{18NK\log_2 R}{\eta}) \right\rceil + K, \right. \\
& \left. \left\lceil \frac{K\log_2 R}{2\Delta^2 \frac{R-1}{R}} \log\left(\frac{12NKe^{2\Delta^2 \log_2 R}}{\eta} \right) \right\rceil \right)
\end{aligned} \tag{4.3}
$$

and GoT length

$$
T_g \geq \left\lceil \frac{72T_m(1/8)}{\rho^2 \pi_{z^*}} \log(\frac{3A}{\eta}) \right\rceil, \tag{4.4}
$$

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

$$
Reg(T) \leq N(T_e + T_g). \tag{4.5}
$$

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 (4.3) and (4.4) 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 [27], 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 Chapter 6 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 (4.3) and (4.4) 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 $\overline{Reg}(T) = O(\log(T))$.*

4.3 Facts and Lemmas for the Regret Analysis

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

$$T_{SS,\min} := \min_{n\in\mathcal{N}} T_{SS,n}. \tag{4.6}$$

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

$$T_{RS,\max} := \max_{n\in\mathcal{N}} T_{RS,n}. \tag{4.7}$$

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 $1-\eta_1$.*

Proof. 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 [14]. According to [14, 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 $1-\eta_1$. □

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.,*

$$|\hat{\mu}_{n,a_n} - \mu_{n,a_n}| < \Delta. \tag{4.8}$$

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 $\Delta < \frac{2(J_1-J_2)}{5N}$, 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

$$\sum_{n=1}^{N} \hat{\mu}_{n,a_n^*} - \operatorname*{argmax}_{\boldsymbol{a}\in\tilde{\mathcal{A}}\setminus\{\boldsymbol{a}^*\}} \sum_{n=1}^{N} \hat{\mu}_{n,a_n} > \frac{(J_1-J_2)}{5}.$$

Proof. $\forall \mathbf{a} \in \tilde{\mathcal{A}}$, 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 $\Delta < \frac{2(J_1 - J_2)}{5N}$. We also have

$$\sum_{n=1}^{N} \hat{\mu}_{n,a_n^*} - \underset{\mathbf{a} \in \tilde{\mathcal{A}} \setminus \{\mathbf{a}^*\}}{\operatorname{argmax}} \sum_{n=1}^{N} \hat{\mu}_{n,a_n} \geq J_1 - N\Delta - J_2 - N\Delta \geq \frac{(J_1 - J_2)}{5}.$$

□

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*

$$T_1(\eta_2) := \left\lceil 8KH_{\max} \log_2 R \log\left(\frac{6NK \log_2 R}{\eta_2}\right)\right\rceil + K \tag{4.9}$$

and

$$T_2(\eta_2) := \left\lceil \frac{K \log_2 R}{2\Delta^2 \frac{R-1}{R}} \log\left(\frac{4NKe^{2\Delta^2 \log_2 R}}{\eta_2}\right)\right\rceil. \tag{4.10}$$

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 $1 - \eta_2$.

Proof. 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 $\forall(n,c) \in \mathcal{N}\times\mathcal{K}$, $\text{Budget}_{n,c}(\tau_{n,c}(T_e)) \geq \lfloor\frac{T_{SS,\min}}{K}\rfloor \geq 8H_{\max}\log_2 R\log(\frac{6NK\log_2 R}{\eta_2})$. Thus, according to [18, Theorem 4.1], we have:

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

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

$$\Pr(E_1|\mathcal{E}) \leq \frac{\eta_2}{2} \text{ and } \Pr(\bar{E}_1|\mathcal{E}) \geq 1-\frac{\eta_2}{2}\ . \tag{4.11}$$

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 $\kappa := \Big\lceil\frac{1}{2\Delta^2}\log(\frac{4NK}{\eta_2})\Big\rceil$ observations of $(c,\gamma_{n,c,b})$ pairs, $\forall c\in\mathcal{K}$. We can write:

$$\begin{aligned}
\Pr(E_{2,n}|B_n) &= \Pr(\exists c\in\mathcal{K}: |\hat{\mu}_{n,(c,\gamma_b)} - \mu_{n,(c,\gamma_b)}| \geq \Delta|B_n)\\
&\leq \sum_{c=1}^{K}\Pr(|\hat{\mu}_{n,(c,\gamma_b)} - \mu_{n,(c,\gamma_b)}| \geq \Delta|B_n)\\
&= \sum_{c=1}^{K}\sum_{j=\kappa}^{\infty}\Pr(|\hat{\mu}_{n,(c,\gamma_b)} - \mu_{n,(c,\gamma_b)}| \geq \Delta, V_{n,(c,\gamma_b)}=j|B_n)\\
&= \sum_{c=1}^{K}\sum_{j=\kappa}^{\infty}\Pr(|\hat{\mu}_{n,(c,\gamma_b)} - \mu_{n,(c,\gamma_b)}| \geq \Delta|V_{n,(c,\gamma_b)}=j, B_n)\\
&\qquad\qquad \times\Pr(V_{n,(c,\gamma_b)}=j|B_n)\\
&\leq \sum_{c=1}^{K}\sum_{j=\kappa}^{\infty}2e^{-2j\Delta^2}\Pr(V_{n,(c,\gamma_b)}=j|B_n) \qquad (4.12)\\
&\leq \sum_{c=1}^{K}2e^{-2\kappa\Delta^2}\sum_{j=\kappa}^{\infty}\Pr(V_{n,(c,\gamma_b)}=j|B_n) \leq 2Ke^{-2\kappa\Delta^2}
\end{aligned}$$

where (4.12) follows from Hoeffding's inequality. Hence, using the union bound we obtain $\Pr(E_2|\cap_n B_n) \leq 2NKe^{-2\kappa\Delta^2} \leq \eta_2/2$. Next, we will show that B_n happens under event $\mathcal{E}$ for all n. According to Sequential Halving algorithm

in [18], 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 [18], 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

$$\Pr(\bar{E}_2 | \mathcal{E}) \geq 1 - \eta_2/2 \ . \tag{4.13}$$

Combining (4.11) and (4.13) by a union bound, we get $\Pr(\bar{E}_1 \cap \bar{E}_2 | \mathcal{E}) \geq 1 - \eta_2$. □

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 (4.1), 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

$$P_g \leq A e^{-\frac{\rho^2 \pi_{z^*} T_g}{72 T_m(\frac{1}{8})}} \leq A e^{-\frac{\rho^2 \pi_{z^*} T_g(\eta_3)}{72 T_m(\frac{1}{8})}} \leq \eta_3.$$

□

4.4 Proof of Theorem 1

Proof. 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 $\Delta < \Delta_0$, we are ensured that the mixing time of the GoT dynamics is bounded almost surely under this event (see Fact 1). Therefore, according to Lemma 5, if GoT phase is run for $T_g(\frac{\eta}{3})$ number of rounds, this 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 $N(T_e + T_g)$. □

Chapter 5

Extensions

5.1 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 the 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.

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	c_1	c_1	c_1	c_2	c_v	c_v	c_1	c_2
	c_2	c_1	c_2	c_2	c_2	c_1	c_2	c_v
	c_2	c_2	c_v	c_v	c_1	c_2	c_v	c_v
	c_1	c_1	c_2	c_1	c_2	c_v	c_v	c_1

Figure 5.1: An illustration of the player orthogonalization using the virtual channel in a network with 4 users ($N = 4$) and 2 channels ($K = 2$). Users 1, 2, 3, and 4 enter the SS sub-phase in rounds 3, 6, 2, and 4, respectively.

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 channels 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. An example of such an orthogonalization procedure is provided in Figure 5.1.

However, N is assumed to be unknown in our problem. Similar to [8], 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 [8, 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.

5.1.1 Fairness

Besides efficient use of the available resources for the case when $N > K$, an alternative approach is to let all the users benefit from the resources equally. An instance of such fair scheduling is provided 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 phase and the users have to remain in the exploration phase for the whole 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 be 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*. Designing more

Algorithm 6 OALA-SHOE

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

practical fair scheduling (e.g., proportional fair scheduling) is an interesting topic for future work.

5.2 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 the Online Auction based Learning Algorithm (OALA) proposed in [11, 27]. 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 [27].

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). 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 learned knowledge is exploited 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 Chapter 4 can also be derived for OALA-SHOE. The advantage of

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**
    $\text{State}_n = assigned$
  end if
  $t \leftarrow t + 2^b + 1$
end while
return c_n^* : the assigned channel

SHOE over random explorations is shown in Chapter 6 via numerical simulations.

For the auction phase to converge to the optimal allocation, the following conditions should hold (for details, see [27]):

1. At the end of 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 $T_A = \left\lceil 4K^2N\left(\frac{\max_{n,c,\gamma}\mu_{n,c,\gamma}}{\Delta_{min}} + \frac{1}{N}\right)\left(2^b+1\right)\right\rceil$ number

of rounds.

Chapter 6

Numerical results

In this chapter, 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.,

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

6.1 Experiment 1: GoT-SHOE

We compare GoT-SHOE with the following benchmarks:

- *Game of Thrones* (GoT): A naive extension of the algorithm in [25], 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 of their algorithm in order to have a fair comparison (see [25, Corollary 1]).
- *Game of Thrones with Trekking* (GoT-Trek): A variant of the algorithm in [25], which uses the idea of player orthogonalization [14] for channel selection. For each (user, channel) pair, rate selection is done in a round-robin

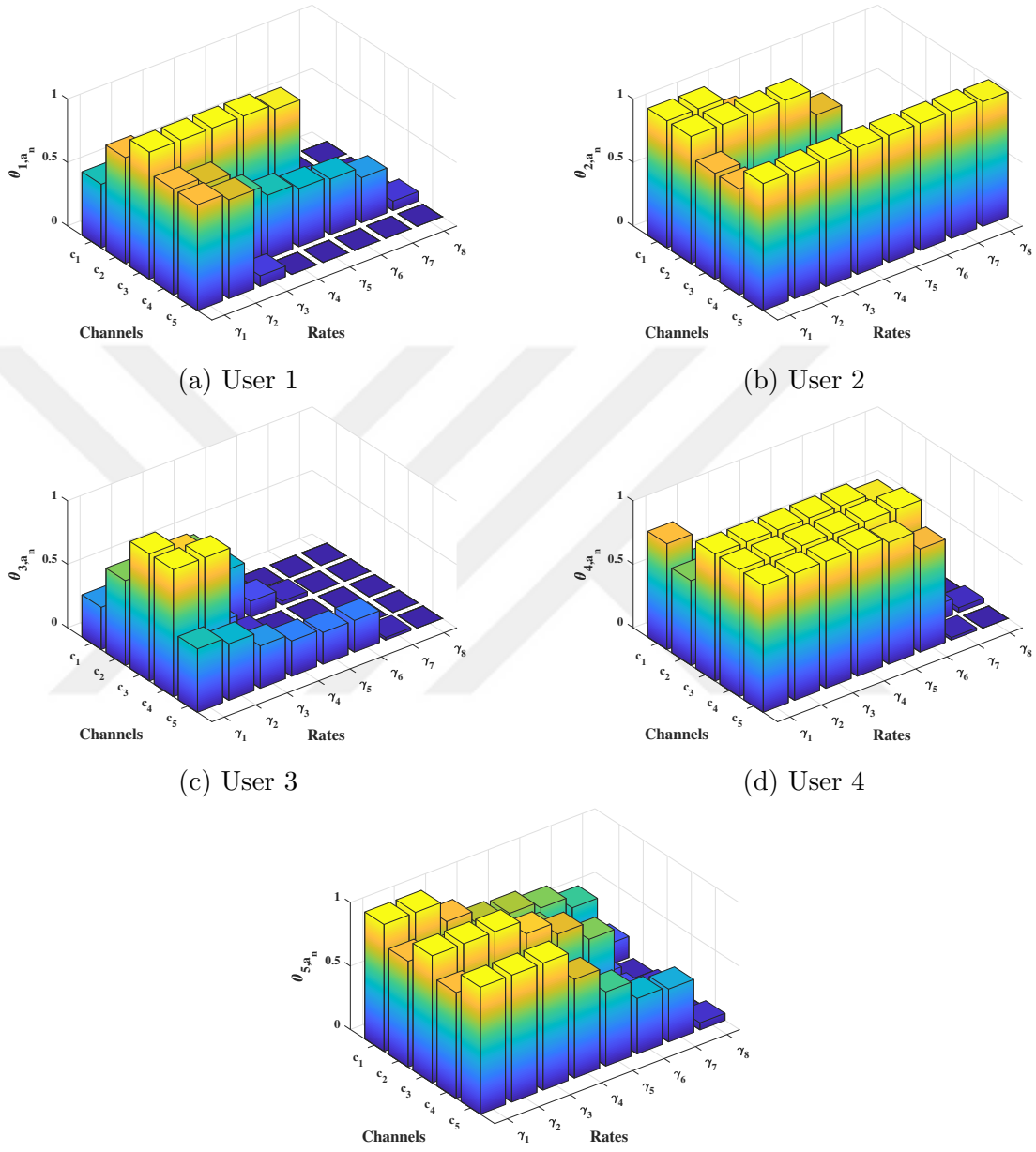
fashion. This algorithm enhances the exploration phase of GoT by employing the trekking strategy in [14], and thus, serves as a good competitor benchmark.

6.1.1 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 and their values are given in the Figure 6.1. It is evident that for every user, $\theta_{n,(c_n,\gamma_n)}$ is a non-increasing function of γ_n. Normalized throughputs or the expected rewards (μ_{n,a_n}'s) of different (channel, rate) pairs are given in Figure 6.2 for each user. For this setup, the optimal allocation is $a_1 = (c_3, \gamma_5),\ a_2 = (c_5, \gamma_8),\ a_3 = (c_2, \gamma_3),\ a_4 = (c_4, \gamma_6),\ a_5 = (c_1, \gamma_7)$.

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.

Figure 6.3 demonstrates that GoT-SHOE significantly outperforms GoT and GoT-Trek both in terms of regret and accuracy. Random channel selections in GoT results 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, Figure 6.3b 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, by using GoT-SHOE, users were able to select the optimal assignment about 78% of the time.



(a) User 1

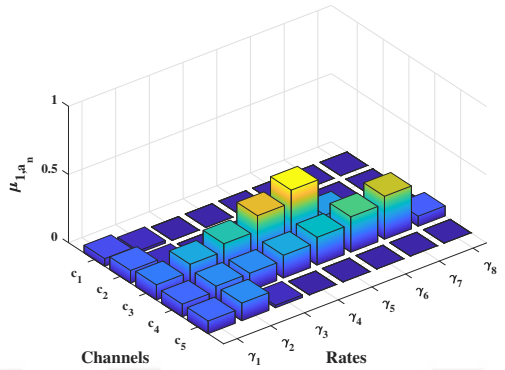
(b) User 2

(c) User 3

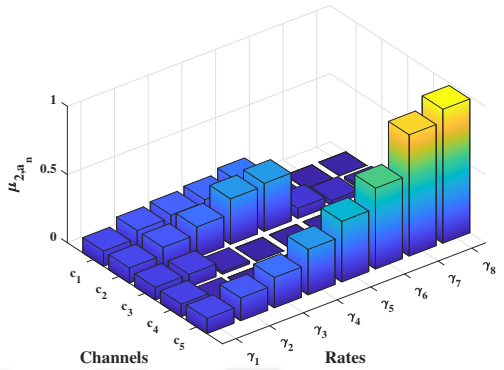
(d) User 4

(e) User 5

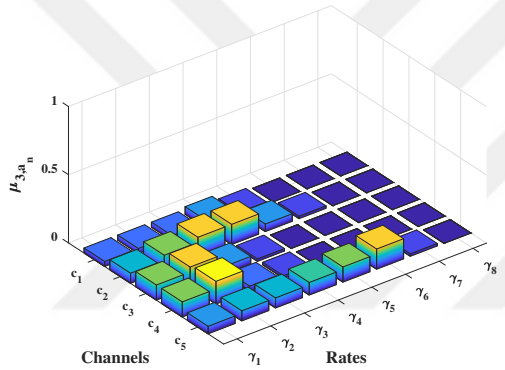
Figure 6.1: Users' packet successful transmission probabilities (θ_{n,a_n}'s) for different (channel, rate) pairs. Note that, $\theta_{n,[c_n\gamma]}$ should be a non-increasing function of γ for any given channel c_n.



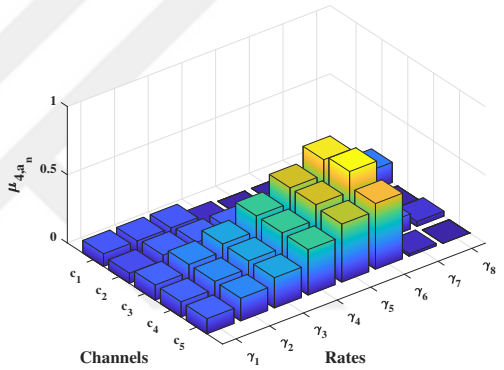
(a) User 1



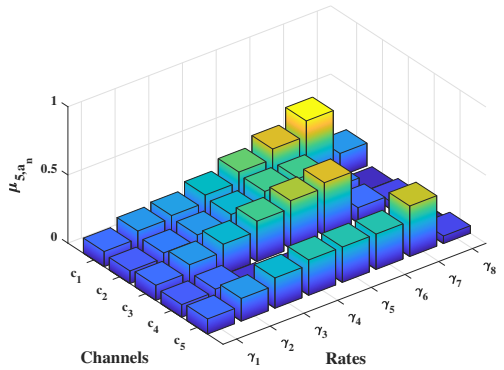
(b) User 2



(c) User 3

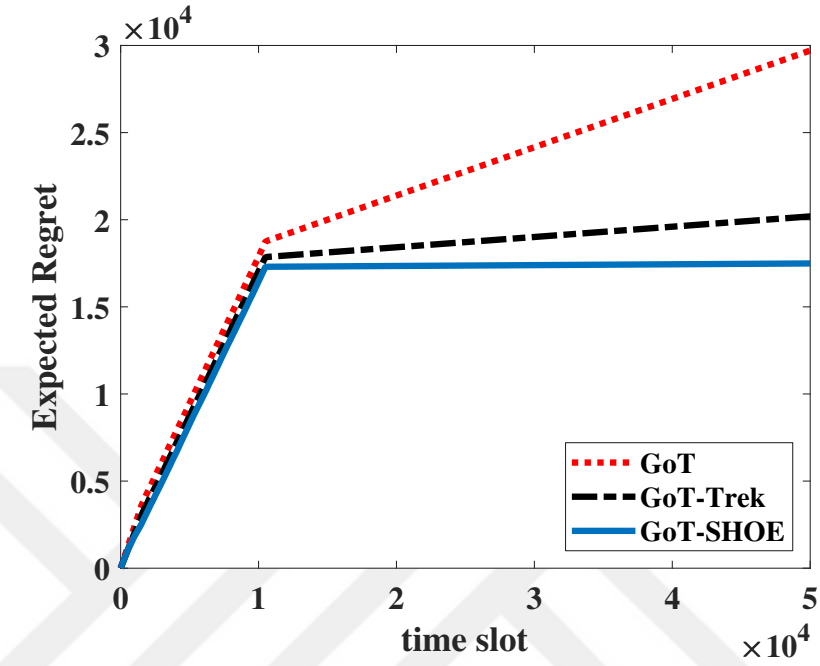


(d) User 4

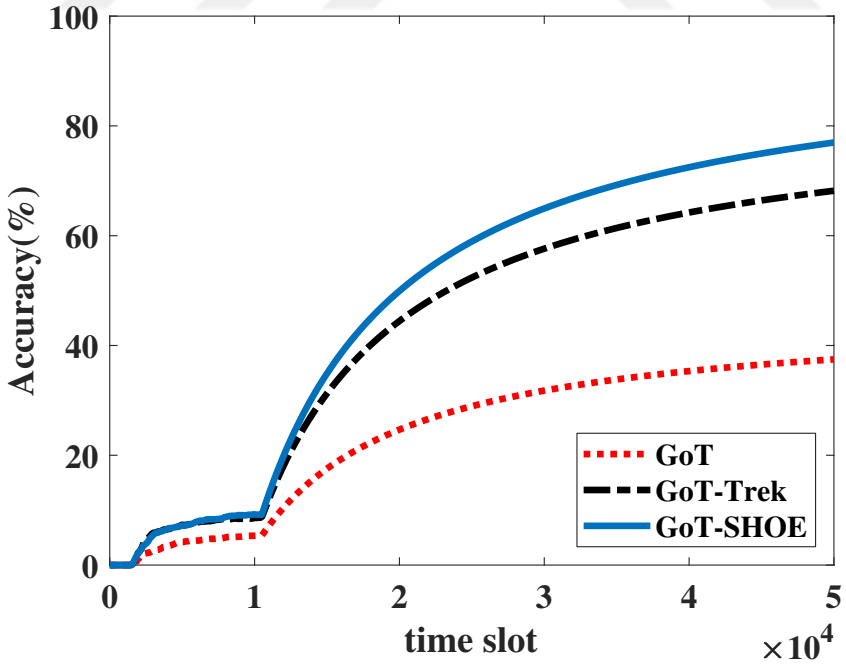


(e) User 5

Figure 6.2: Users' normalized throughputs (or expected rewards μ_{n,a_n}'s where $\mu_{n,a_n} = \frac{\gamma_n}{\gamma_R}\theta_{n,a_n}$) for different (channel, rate) pairs.



(a) Regret



(b) Accuracy

Figure 6.3: Comparison of GoT-SHOE with GoT and GoT-Trek under the baseline configuration.

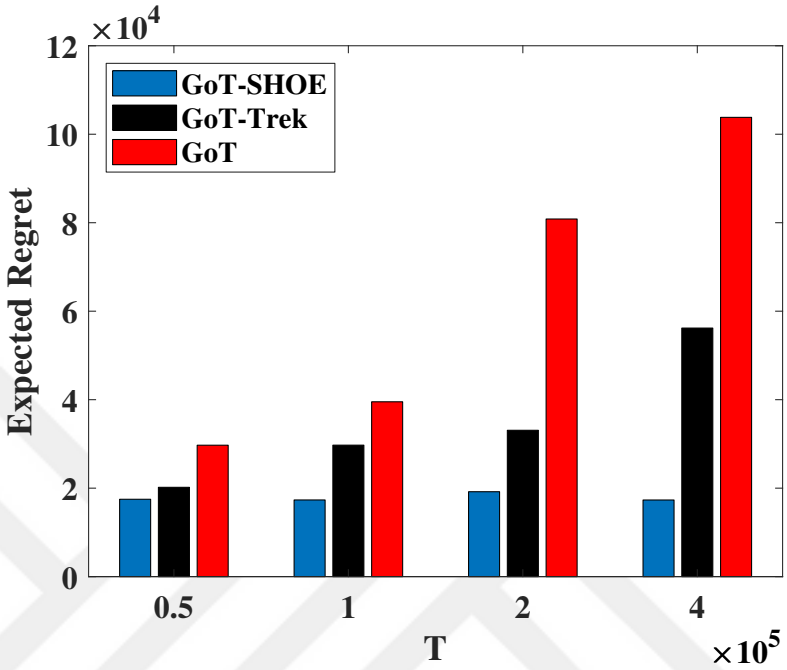


Figure 6.4: Comparison of GoT-SHOE with GoT and GoT-Trek for different time horizons.

6.1.2 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):** Figure 6.4 compares the expected regret of the algorithms for different time horizons ($T \in \{0.5, 1, 2, 4\} \times 10^5$). Such an observation was quite obvious in advance. In figure 6.3a, it is observed that the expected regret of our algorithm (almost) does not change after the GoT phase while the expected regret of the GoT-Trek and the GoT algorithms increase with a small and a high slope respectively. Therefore, for the fixed exploration and GoT length ($T_e + T_g = 1500 + 9000 = 10500$), we expect to see (almost) the same expected regret for GoT-SHOE and higher expected regret 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 Figure 6.5. 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 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 Figure 6.5c that the 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 Figure 6.6a 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 Figure 6.6b 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.

- **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 $\frac{5N}{T_g}$. Figure 6.6c 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 Figure 6.6d.

- **Different number of rates (R):** The set of rates for the 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) $R = 2 : \mathcal{R} = \{6, 54\}$.

  Figure 6.6e 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 increase from 2 to 8.

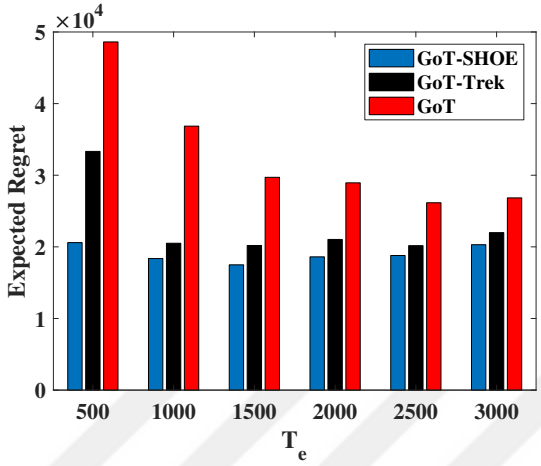
- **When the number of users has to be estimated (i.e., $N \leq K$ is not known a priori):** Here N is 5, and each user performs $T_{\text{est}} = 1200$ random exploration to estimate N, then she spends $T_e = 1500$ rounds in SHOE (the rest of the parameters are similar to the baseline configuration). 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:

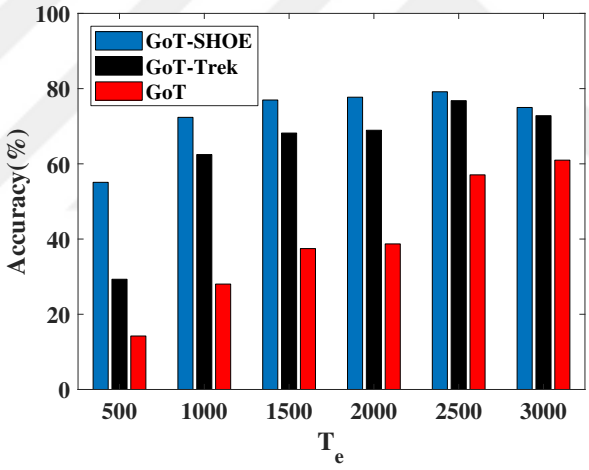
  (i) $K = 2 < N$,

  (ii) $K = 5 = N$.

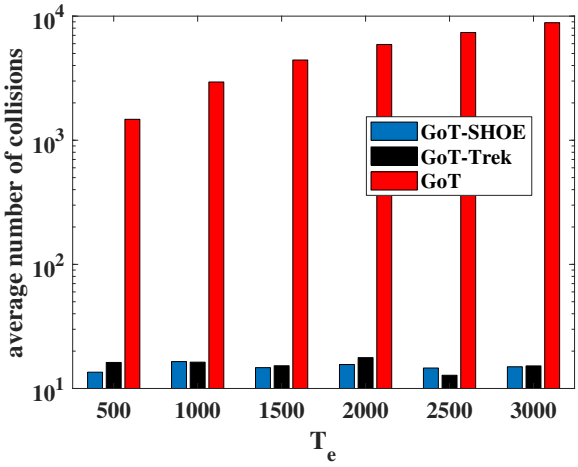
  In both cases, it is observed that the users can successfully estimate the number of users in the system and GoT-SHOE outperforms its competitors (see Figure 6.6f).



(a) Regret

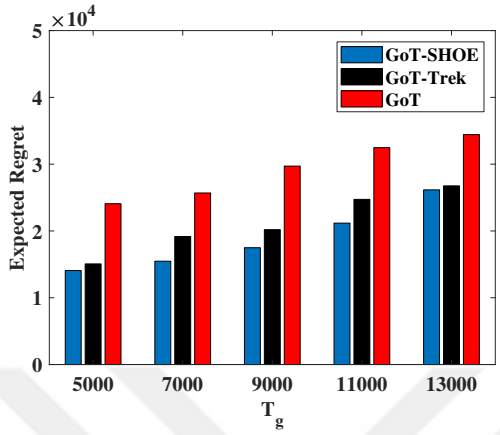


(b) Accuracy

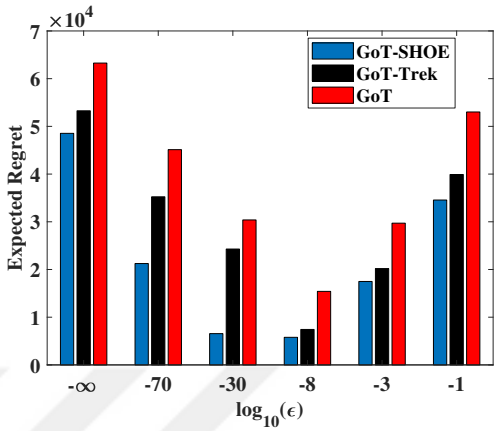


(c) Average number of collisions

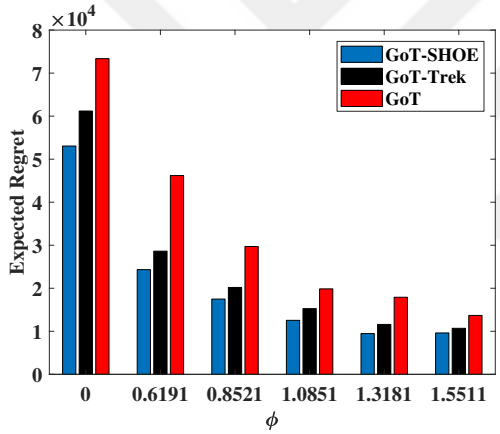
Figure 6.5: Comparison of GoT-SHOE with GoT and GoT-Trek for different exploration lengths.



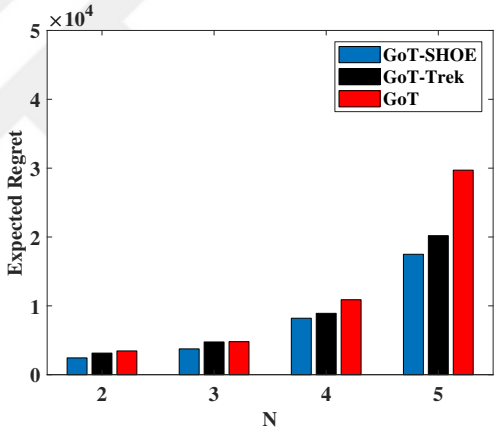
(a) different T_g



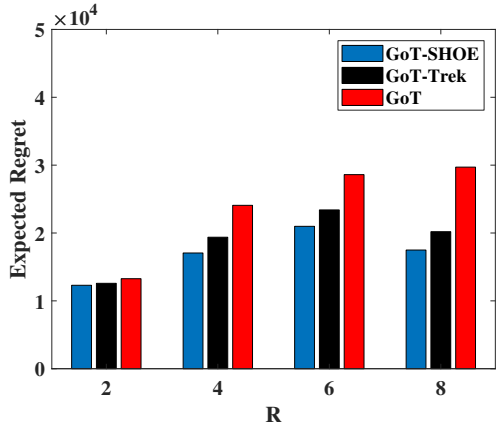
(b) different ϵ



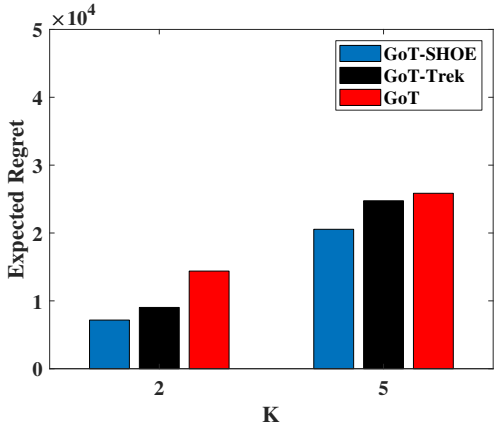
(c) different ϕ



(d) different N



(e) different R



(f) estimate N

Figure 6.6: Comparison of the expected regrets of GoT-SHOE, GoT and GoT-Trek under different parameters.

6.2 Experiment 2: OALA-SHOE

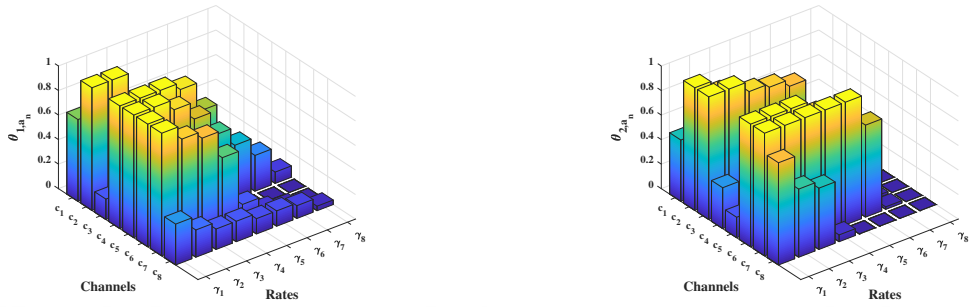
We compare OALA-SHOE with the following benchmarks:

- OALA: A naive extension of the algorithm in [11], 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 [27], which uses the idea of player orthogonalization [14] 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 [14], 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. The packet successful transmission probabilities ($\theta_{n,(c_n,\gamma_n)}$'s) and the normalized throughputs (μ_{n,a_n}'s) for different users are given in Figure 6.7 and Figure 6.8 respectively. 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. Figure 6.9a 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, Figure 6.9b 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.

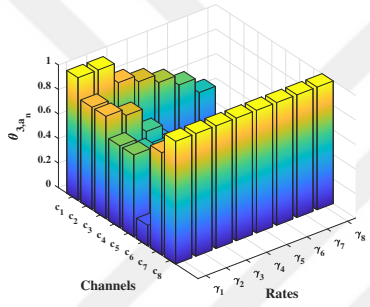
Table 6.1: The average number of colliding players per time slot in exploration phases.

Algorithm	OALA	OALA-Trek	OALA-SHOE
Average	4.8591	0.0057	0.0051

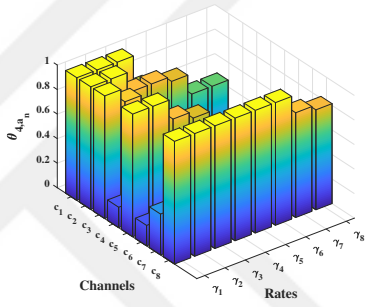


(a) User 1

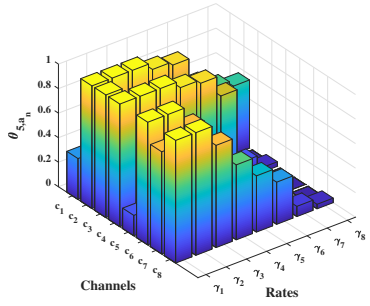
(b) User 2



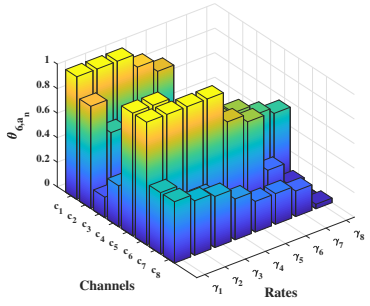
(c) User 3



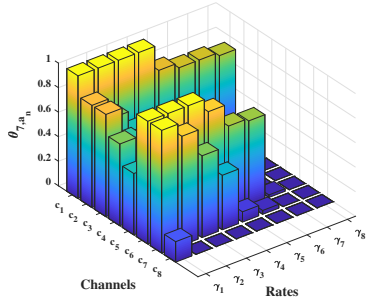
(d) User 4



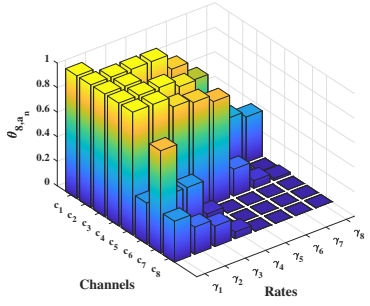
(e) User 5



(f) User 6

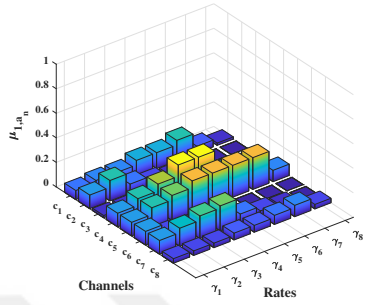


(g) User 7

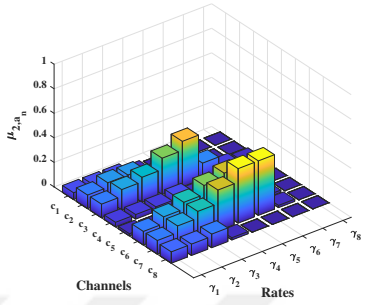


(h) User 8

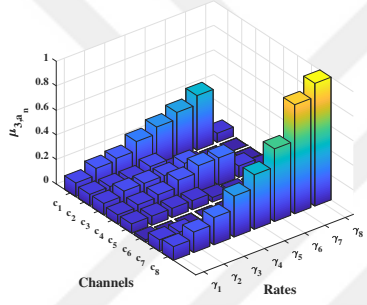
Figure 6.7: Users' packet successful transmission probabilities (θ_{n,a_n}'s) for different (channel, rate) pairs. Note that, $\theta_{n,[c_n\gamma]}$ should be a non-increasing function of γ for any given channel c_n.



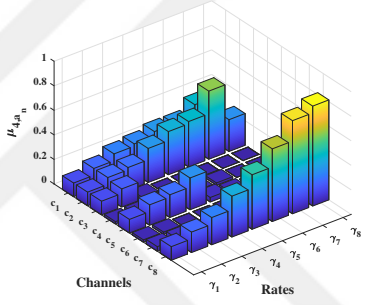
(a) User 1



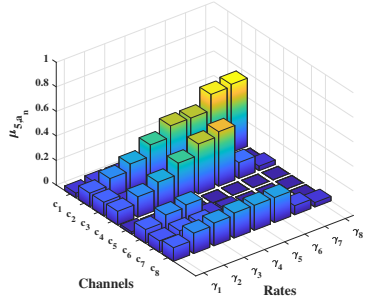
(b) User 2



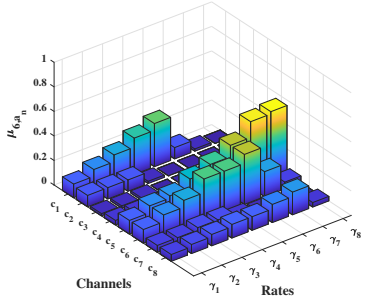
(c) User 3



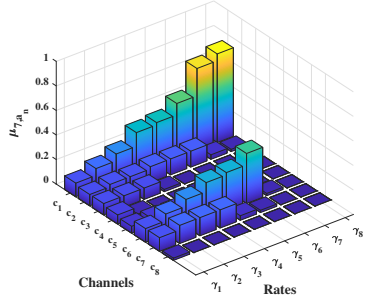
(d) User 4



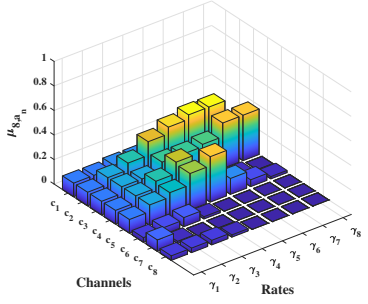
(e) User 5



(f) User 6

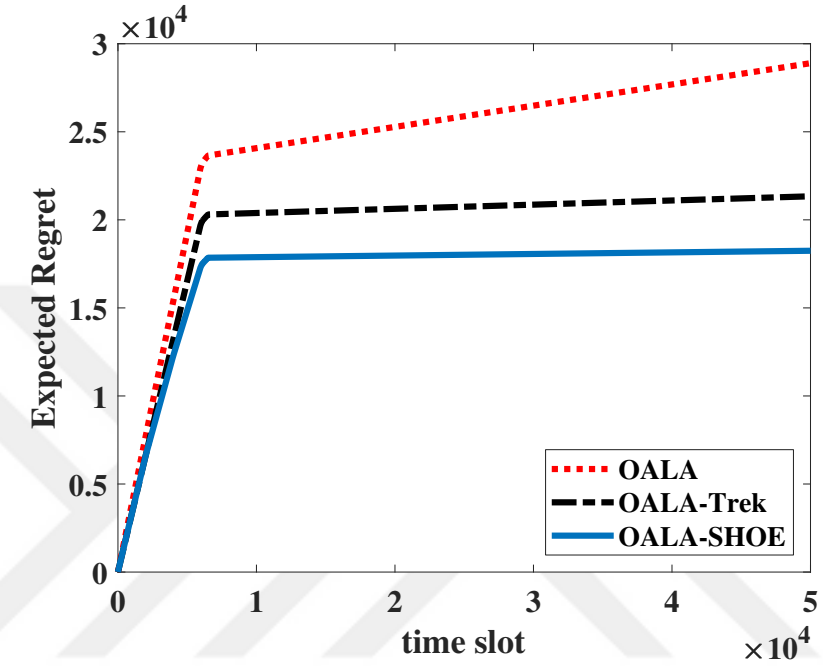


(g) User 7

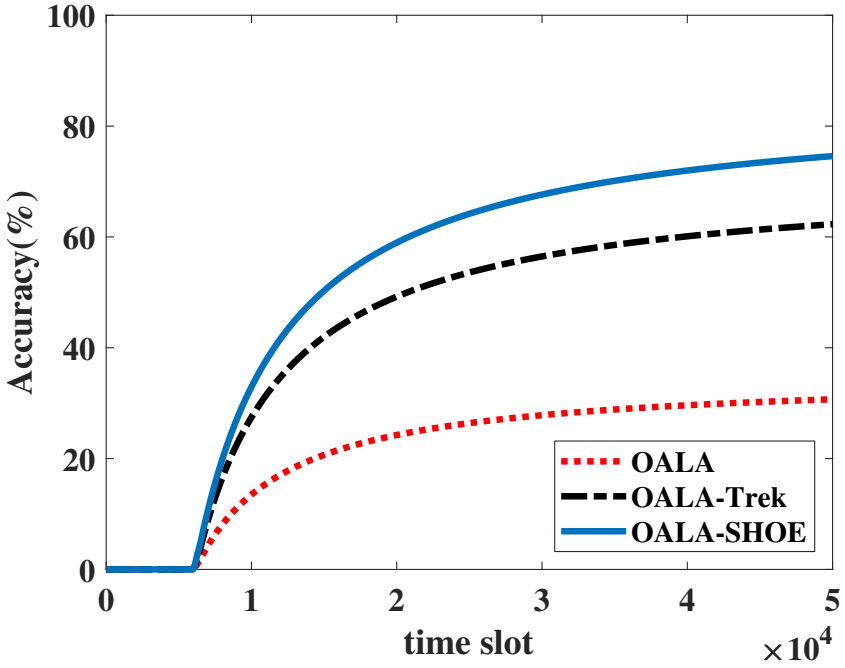


(h) User 8

Figure 6.8: Users' normalized throughputs (or expected rewards μ_{n,a_n}'s) for different (channel, rate) pairs.



(a) Regret



(b) Accuracy

Figure 6.9: Comparison of OALA-SHOE with OALA and OALA-Trek.

Chapter 7

Conclusion and Future Work

In this thesis, we proposed a decentralized learning algorithm for dynamic rate and channel adaptation over a shared spectrum. To the best of our knowledge, the proposed algorithm is the first one that addresses the joint channel and rate selection problem in a fully distributed multi-user network. Our algorithm uses the ideas of orthogonal exploration and sequential halving to keep the number of collisions low as well as learning 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. Moreover, we considered the case where the number of users is greater than the number of channels and proposed an extension of our algorithm that works under this assumption. 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 [17]). 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.

Bibliography

[1] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933.

[2] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Adv. Appl. Math.*, vol. 6, no. 1, pp. 4–22, 1985.

[3] 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.

[4] V. Anantharam, P. Varaiya, and J. Walrand, "Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays-part i: I.i.d. rewards," *IEEE Transactions on Automatic Control*, vol. 32, no. 11, pp. 968–976, 1987.

[5] 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, 2011.

[6] D. Kalathil, N. Nayyar, and R. Jain, "Decentralized learning for multiplayer multiarmed bandits," *IEEE Trans. Inf. Theory*, vol. 60, no. 4, pp. 2331–2345, 2014.

[7] O. Avner and S. Mannor, "Concurrent bandits and cognitive radio networks," vol. 8724, 04 2014.

[8] J. Rosenski, O. Shamir, and L. Szlak, "Multi-player bandits–a musical chairs approach," in *Proc. 33rd Int. Conf. Mach. Learn.*, pp. 155–163, 2016.

[9] L. Besson and E. Kaufmann, "Multi-player bandits revisited," in *Proc. Algorithmic Learn. Theory*, vol. 83, pp. 56–92, 2018.

[10] 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.

[11] 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, 2019.

[12] C. Tekin and M. Liu, "Online learning in decentralized multi-user spectrum access with synchronized explorations," in *Proc. 2012 IEEE Military Commun. Conf.*, pp. 1–6, 2012.

[13] M. Bande and V. V. Veeravalli, "Multi-user multi-armed bandits for uncoordinated spectrum access," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, pp. 653–657, 2019.

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

[15] G. Lugosi and A. Mehrabian, "Multiplayer bandits without observing collision information," *arXiv preprint arXiv:1808.08416v1*, 2018.

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

[17] 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, 2015.

[18] Z. Karnin, T. Koren, and O. Somekh, "Almost optimal exploration in multi-armed bandits," in *Proc. 30th Int. Conf. Mach. Learn.*, vol. 28, pp. 1238–1246, 2013.

[19] Y. Gai, B. Krishnamachari, and R. Jain, "Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation," in *Proc. 2010 IEEE Symposium on New Frontiers in Dynamic Spectrum (DySPAN)*, pp. 1–9, 2010.

[20] J. Komiyama, J. Honda, and H. Nakagawa, "Optimal regret analysis of thompson sampling in stochastic multi-armed bandit problem with multiple plays," in *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pp. 1152–1161, JMLR.org, 2015.

[21] 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, 2010.

[22] P. Alatur, K. Y. Levy, and A. Krause, "Multi-player bandits: The adversarial case," *J. Mach. Learn. Research*, vol. 21, no. 77, pp. 1–23, 2020.

[23] 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, 2018.

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

[25] I. Bistritz and A. Leshem, "Game of thrones: Fully distributed learning for multiplayer bandits," *Mathematics of Operations Research*, 2020.

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

[27] S. M. Zafaruddin, I. Bistritz, A. Leshem, and D. Niyato, "Multiagent autonomous learning for distributed channel allocation in wireless networks," in *Proc. 20th IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, pp. 1–5, 2019.

[28] 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.*, pp. 1–6, 2014.

[29] A. Garivier and E. Kaufmann, "Optimal best arm identification with fixed confidence," in *Proc. Conf. Learn. Theory*, pp. 998–1027, 2016.

[30] D. Russo, "Simple Bayesian algorithms for best arm identification," in *Proc. Conf. Learn. Theory*, pp. 1417–1418, 2016.

[31] 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, 2018.

Appendix A

Notation table

Table A.1: Notation Table

Notation	**Explanation**
$\mathcal{N}$, N	Set of users, number of users.
$\mathcal{K}$, K	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 (channel-rate pair) 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.

Continued on next page

Table A.1 – *Notation Table (continued)*

Notation	**Explanation**
$\tilde{\mathcal{A}}$	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 $\mathbf{a}$.
$X_{n,a_n}(t)$	Bernoulli random variable representing 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 specified 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 $\mathbf{a}$.
$\mathrm{Reg}(T)$, $\overline{\mathrm{Reg}}(T)$	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.
$\varepsilon_{n,a_n}(t)$	Dither value added by the user n to $\hat{\mu}_{n,a_n}$.
$u_n(\mathbf{a})$	Utility of user n in strategy profile $\mathbf{a}$.
T	Total number of rounds
T_e	Length of the exploration phase
T_g	Length of the GoT phase
$\mathcal{A}'_n$	Set of available actions of user n at the beginning of the GoT phase
C, D	Content State, discontent State
Z	State Space, $Z = \prod_n(\mathcal{A}'_n \times \mathcal{M})$, where $\mathcal{M} = \{C, D\}$
$T_m(\frac{1}{8})$	Mixing time of state space Z with an accuracy of $\frac{1}{8}$
π	Stationary distribution of Z

Continued on next page

Table A.1 – *Notation Table (continued)*

Notation	**Explanation**
φ_i	The probability distribution of the state i at the beginning of the GoT phase
z^*	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.
$\tau_{n,c}(t)$	Time 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.
$\text{Budget}_{n,c}(\tau_{n,c}(t))$	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