Learning to coordinate without communication in multi-user multi-armed bandit problems

Orly Avner Shie Mannor

May 1, 2015

Abstract

We consider a setting where multiple users share multiple channels modeled as a multi-user multi-armed bandit (MAB) problem. The characteristics of each channel are initially unknown and may differ between the users. Each user can choose between the channels, but her success depends on the particular channel as well as on the selections of other users: if two users select the same channel their messages collide and none of them manages to send any data. Our setting is fully distributed, so there is no central control and every user only observes the channel she currently uses. As in many communication systems such as cognitive radio networks, the users cannot communicate among themselves so coordination must be achieved without direct communication. We develop algorithms for learning a stable configuration for the multiple user MAB problem. We further offer both convergence guarantees and experiments inspired by real communication networks.

1 Introduction

The inspiration for this paper comes from the world of distributed multi-user communication networks, such as cognitive radio networks. These networks consist of a set of communication channels with different characteristics, and independent users whose goal is to transmit over these channels as efficiently as possible.

Modern networks, such as cognitive radio networks, must cope with several challenges. First and foremost, the networks’ distributed nature prohibits any form of central control. In addition, many users operate on an “ad hoc” basis, preventing them from forming inter-user communication. In fact, they probably do not even know how many users share their network.

On top of these issues, that concern the coordination of multiple users, the channel characteristics may be initially unknown, and differ between users. The first assumption entails a learning approach, while the consequences of the different characteristics are discussed in detail in the sequel.
1.1 Cognitive radio networks

Cognitive Radio Networks (CRNs), introduced in [21], have attracted considerable attention in recent years. The idea that lies at the heart of CRNs is that advanced sensing mechanisms and increased computation power may enable radio devices to dramatically improve their performance in terms of resource utilization, resilience and more. Networks of such users are usually dynamic and stochastic, giving rise to many interesting problems [1, 14]. In our paper we focus on developing a sensing and transmission scheme that enables users to learn a stable, orthogonal configuration without communicating directly.

1.2 Multi-armed bandits

A well known framework for learning in CRNs is the classical Multi-Armed Bandit (MAB) model. MABs offer a simple, intuitive framework for learning the characteristics of a number of unknown options in an online manner, while balancing exploration and exploitation. A MAB problem consists of a single user repeatedly choosing between arms with different characteristics, that are initially unknown. After every round, the user acquires a reward that depends on the arm she chose. Her goal in most setups is to maximize the expected sum of rewards acquired over time.

The channels of a CRN are naturally cast as the arms of a bandit, as first suggested in [15], with different performance measures (bandwidth, ack signals, bit rate) serving as the reward.

Many papers propose solutions for the stochastic MAB problem (see, e.g., [4, 6, 12]) and its adversarial version (see, e.g., [5]), but they all assume a single user is sampling the arms of the bandit.

However, this assumption does not apply in multi-user networks. In the multi-user MAB model, users compete over the arms of the same bandit. As a result, they are bound to experience collisions (i.e., multiple users sampling the same arm), unless they employ some form of collision avoidance or coordination mechanism. Collisions in communication networks result in performance degradation, corresponding to reward loss in the MAB model. In order to meet the goal of reward maximization, the presence of multiple users must be addressed. We survey several approaches to this issue in Section 1.4.

1.3 Extension of the CRN-MAB setting

The novelty introduced in our paper lies in the combination of bandit learning, multiple users, different reward distributions for different users and no direct communication. The combination of these last two demands - different distributions and no direct communication, poses a real challenge.

As explained in detail in Section 2.3 and in Section 2.4 the only thing we can guarantee in terms of network behavior in this setup is stability. In a dynamic, distributed network, stability should not to be taken for granted, and it is of great value. Once a network has reached a stable configuration, users can
focus on utilizing its resources, rather than engaging in coordination or learning efforts; a stable network is more robust and efficient.

Reaching stability is a nontrivial task, since users must learn their channel characteristics while coordinating their actions with the other users, based on very limited observations.

1.4 Previous work

We now present several approaches to our problem, coming from different areas and disciplines.

Our problem may be viewed as an assignment problem, i.e., maximum weight matching in a weighted bipartite graph. Users correspond to agents, channels to tasks, and rewards are simply the complementary of the costs of graph edges. Several papers have been published on the distributed assignment problem, but to the best of our knowledge none of them offers a solution for our problem. The well-known Hungarian method [15] requires full knowledge of the graph (i.e., channel characteristics) and assumes the existence of central control. The Bertsekas auction algorithm [8] frees us from the need for central control, at the cost of direct communication between nodes. The classical Gale-Shapley algorithm [11] solves the problem of finding a stable marriage configuration, but does not take the need to learn into account. Some papers have actually applied it to CRNs, but not in the learning context [9, 20]. Another work on distributed stable marriage, that makes use of a variant of the Gale-Shapley algorithm, is [10]. While it is quite foreign to our problem, the potential defined in the paper is helpful in our analysis. Another noteworthy work in this context is [2]. The authors address the challenge of limiting communication between nodes to a minimum, and propose two communication models. Nevertheless, they allow more communication than we would like, and their formulation does not consider learning. Two additional results that deal with distributed stable marriage offer lower bounds and state that some form of information exchange is inevitable when solving such problems [13, 17].

The papers closest to ours in spirit are those dealing with multi-user MABs. There has been work on the case of reward distributions that do not vary between users, such as [3] and [7]. The latter introduces an algorithm that is able to cope with a variable number of users. Another paper, that addresses different reward distributions for different users, is [16]. Here, the authors solve the challenge by employing the Bertsekas auction algorithm. This approach enables users to reach a reward-maximizing solution, at the price of direct, frequent communication between themselves.

To this end, we would like to point out that communication between users is undesirable not only because of its price in terms of network resources and time. Once users depend on communication, they are more vulnerable to intentional attacks that may disrupt it, as well as noise bursts that are common in CRNs. In addition, they need to have some basic knowledge about each other in order to set up such a communication protocol - knowledge that our algorithm does not require.
2 Model and formulation

2.1 System and users

We model a communication network with $K$ channels, servicing $N$ independent users. Our work is based on the assumption that $K \geq N$, which is reasonable since without it, a time division based approach is necessary. In this system, time is slotted and users’ clocks are synchronized.

The communication network consists of $K$ channels, where only one user can transmit over a certain channel during a single time slot. Each transmission yields a reward, which we assume to be stochastic.

The users are a group of $N$ independent, selfish agents. Their observations are local, consisting only of the history of their actions and rewards. In addition, they do not know the number of users they share a network with. There is no central control managing their use of the network, and they do not have direct communication with each other.

A key characteristic of our model is that the expected reward a channel yields depends not only on the identity of the channel, but also on the identity of the user. Formally, the rewards of the channels are Bernoulli random variables with expected values $\{\mu_{n,k}\}$, where $n \in \{1, \ldots, N\}$ and $k \in \{1, \ldots, K\}$.

We model the users’ sharing resources through the representation of the communication network by a single bandit. This means that two users, $n$ and $m$, attempting to access the same channel $k$ at the same time $t$, will experience a collision. In our model, the result of a collision is complete loss of communication for that time slot for the colliding users, i.e., zero reward. A user $n$ that accesses a channel $k$ alone during a certain time slot will receive a reward drawn i.i.d. from a Bernoulli distribution with expected value $\mu_{n,k}$. Throughout the paper, we use the term configuration to refer to a mapping of users to channels.

2.2 Limited coordination

In an effort to keep our model faithful to real world CRNs, we limit the coordination between users to a minimum. Thus, users can only transmit in a channel of their choice, or sense the spectrum range and receive binary feedback regarding all channels $\{1, \ldots, K\}$ at time $t$:

- “0”: no transmission in channel
- “1”: transmission in channel

2.3 Reward maximizing solution

We adopt a system-wide view for characterizing the optimal solution. The goal is therefore to maximize the sum of rewards over all users, over time. The solution that achieves this goal is an orthogonal configuration, in which each user focuses on a single channel, and channels are sampled by no more than one
The assignment of users to channels is chosen so as to maximize the sum of rewards:

\[ R^* = \max_{\pi \in \mathcal{C}} \sum_{n=1}^{N} \mu_{n,\pi(n)}, \tag{1} \]

where \( \mathcal{C} \) is the set of all possible permutations of subsets of size \( N \) chosen without replacement from the set \( \{1, \ldots, K\} \).

However, reaching such a solution requires direct information exchange between users. Assume a channel \( k \) maximizes the expected reward of two different users \( m \) and \( n \), but \( \mu_{m,k} > \mu_{n,k} \). To maximize the system-wide reward, user \( n \) must step down and choose a different channel. However, since there is no central control, then without explicit information exchange regarding the values of \( \mu_{m,k} \) and \( \mu_{n,k} \), \( m \) and \( n \) cannot decide which of them should hold on to channel \( k \).

Since the reward-maximizing solution is not attainable in our setup, due to the limited information exchange, we focus on convergence to a stable, orthogonal configuration.

### 2.4 Stable marriage solution

Our goal is to develop policies that will lead users to a stable configuration. Such a configuration may be perceived as an absorbing state - from which there are no deviations. We employ the notion of stable marriage to formally define stability:

**Definition 1.** A Stable Marriage Configuration (SMC) is an assignment of users to channels such that no two users are willing to swap channels. Formally, for a pair of users \( n,m \):

\[ S_1 \triangleq (\mu_{n,a_n} < \mu_{n,a_m}) \quad \text{user } n \text{ would like to swap} \]

\[ S_2 \triangleq (\mu_{m,a_m} \leq \mu_{m,a_n}) \quad \text{user } m \text{ is willing like to swap}, \]

where \( a_m \) and \( a_n \) are the users’ current actions. In an SMC,

\[ S_1 \land S_2 = 0 \quad \forall n,m. \]

### 2.5 Goal

Given a system with \( K \) channels and \( N \) users, allowing only limited communication as described in Section 2.2, our goal is to reach a configuration that is orthogonal: no two users use the same channel, and an SMC, according to Definition 1.
3 Coordination protocol

We now turn to describing our coordination protocol, that balances the limitations of Section 2.2, with the users’ need for information exchange. We maintain this balance by introducing a signalling mechanism between pairs of users. At predefined time slots, a user \( n \) wishing to occupy a channel \( k \) may transmit in that channel to express her wish.

In order to make sure that this signal is received by user \( m \), who currently occupies channel \( k \), we propose a frame-based protocol. We note that in order to implement this protocol, users need to be able to transmit and receive at the same time. This is a reasonable requirement in modern communication systems.

The following explanation is best understood by observing Figure 1. Our protocol divides time into super frames of length \( T_{SF} = 2 + 2(K - 1) \). Each super frame begins with a pair of time slots, \( S_1 \) and \( S_2 \), during which a single signalling user is coordinated for the entire super frame, as described in Algorithm 1. These are followed by \( K - 1 \) mini-frames of 2 time slots each, denoted by \( S_3 \) and \( S_4 \). Each of these mini-frames corresponds to one channel on the initiator’s list of preferred channels. Thus, a single super frame enables one user to go over her entire preference list and signal other users, suggesting they swap channels with her. The time slots marked \( S_4 \) allow users not participating in the coordinating process during a certain mini-frame to sample their current channel and proceed with the learning-while-transmitting process. Thus, all but two users (initiator and responder) gather a sample during each mini-frame, resulting in at least \( K - 2 \) samples for each of the users, except for the initiator, over each super frame. For an exact description of our coordination protocol, see Figure 2.

While this may seem like much coordination, the protocol is very simple to implement, and is indeed lightweight when compared to other protocols. It is important to note that all transmissions performed over the course of these coordination attempts can also serve as informative samples for the learning process. In order to reduce clutter in our presentation and implementation of the algorithm, we do not make use of these samples.

4 The CSM-MAB algorithm

We now turn to a full description of our algorithm, the Coordinated Stable Marriage Multi-Armed Bandit (CSM-MAB) algorithm. We propose a user-level algorithm for a fully distributed system, whose goal is described in Section 2.5. When all users in the network apply CSM-MAB, the assignment of users to channels is guaranteed to be orthogonal, and converges to an SMC.

Our algorithm begins with a start up phase, during which users transmit and sense to detect collisions, in order to reach an initial orthogonal configuration (line 1). This phase follows the lines of the CFL algorithm introduced in [19], and converges quickly. Once an initial orthogonal configuration has been reached, users start executing the CSM-MAB algorithm, described in Algorithm 1.
Figure 1: Coordination protocol - frame structure.

Figure 2: Coordination protocol flow for a single super frame. Events (transmission or sensing) appearing on the same line take place simultaneously.
As explained in Section 3, time is divided into Super Frames (SFs). At the beginning of each SF, users individually create a list of channels they prefer over their current action (line 4). Channels are assigned values according to their UCB indices, calculated using the well known formula

\[ I_{n,k}(t) = \hat{\mu}_{n,k} + \sqrt{\frac{2 \ln t}{s_{n,k}}}, \]  

where \( \hat{\mu}_{n,k} \) is the empirical mean of the reward acquired by user \( n \) on channel \( k \) up till time \( t \) and \( s_{n,k} \) is the number of times she sampled arm \( k \) up till time \( t \).

Next, the users attempt to coordinate an initiator: every user who would like to improve upon her current choice of channel presents herself as the initiator with a probability of \( \epsilon = \frac{1}{N} \) (lines 5-11). An agreed initiator for the current SF emerges if and only if the number of non-zero entries in \( S_2 \) is exactly 1 (the value of \( \epsilon \) is chosen in order to maximize the probability of this occurring). Assuming a single initiator is agreed upon, all users observe her current channel, based on \( S_1 \). They will need this knowledge to decide whether to accept a swapping suggestion, if they are signalled during the SF.

The initiator proceeds to signal other users, based on her ranking of the different channels (lines 13-22). Each responder (i.e., signalled user) checks whether swapping channels with the initiator will improve her situation, based on her own UCB ranking of the channels. For the sake of collective benefit, if the index of the initiator’s channel is exactly equal to the index of responder’s current choice, she will also agree to swap. Once a responder agrees, a swap takes place. No more signalling attempts are made till the end of the SF, and users simply continue sampling their chosen channels. If the responder refuses, the initiator will approach the next-best channel on her list. She will continue the process until she (a) finds a partner that agrees to swap; or (b) exhausts her list of potential channels to swap to.

If the initiator would like to switch to a vacant channel (this knowledge is obtained from inspecting the vector \( S_1 \), see Section 3), then there is no need for signalling, and the initiator simply updates her chosen action.

All users except for the initiator and the responder gather a sample for the learning process with each mini-frame (lines 23-24). The initiator does not gather samples at all during the entire super frame, as she is occupied with approaching other users, and responders do not gather samples during the single mini-frame in which they are approached.

5 Analysis

We will now show that the CSM-MAB meets the goals defined in Section 2.5. Our main theoretical result is stated in Theorem 1.

**Theorem 1.** Consider a system with \( K \) channels and \( N \) users, with channel rewards characterized by the matrix \( \mu \). Applying CSM-MAB (Algorithm 2) by all users will result in convergence to an orthogonal SMC: For all \( \delta > 0 \) there
Algorithm 1 CSM-MAB algorithm

1: \( a_n(0) \leftarrow \text{apply\_CFL}(K) \)
2: \textbf{for all} frames \( t \) \textbf{do}
3: \hspace{1em} \textbf{if} \ \mod (t, T_{SF}) \equiv 1 \ \textbf{then} \ \{ \text{Beginning of SF} \}
4: \hspace{2em} \text{list} \leftarrow \text{rank\_channels}(a_n(t-1), \hat{\mu}_n, s_n)
5: \hspace{2em} \textbf{if} \ \text{list} \neq 0 \ \textbf{then} \ \{ \text{User} n \ \text{would like to change channels} \}
6: \hspace{3em} \text{flag}_n \leftarrow \text{rand}(\text{Bernoulli}, \epsilon)
7: \hspace{3em} \textbf{if} \ \text{flag}_n = 1 \land \text{flag}_i = 0 \ \forall \ i \neq n \ \textbf{then} \ \{ \text{Only} n \ \text{raised a flag} \}
8: \hspace{4em} \text{initiator} = n \ \{ \text{User} n \ \text{is the initiator for this SF} \}
9: \hspace{4em} \text{pref} = 1 \ \{ \text{Swapping preference is initialized to 1} \}
10: \hspace{2em} \textbf{end if}
11: \hspace{2em} \textbf{end if}
12: \hspace{1em} \textbf{else}
13: \hspace{2em} \textbf{if} \ \text{initiator} = n \land \text{pref} > 0 \ \textbf{then} \ \{ n \ \text{is the initiator, list not exhausted yet} \}
14: \hspace{3em} \text{response} \leftarrow \text{propose\_swap}(\text{list}(\text{pref}))
15: \hspace{3em} \textbf{if} \ \text{response} = 1 \ \textbf{then} \ \{ \text{Responder agreed} \}
16: \hspace{4em} \text{a}(t) \leftarrow \text{swap}(a_n(t), \text{list}(\text{pref}))
17: \hspace{4em} \text{pref} \leftarrow 0
18: \hspace{3em} \textbf{else}
19: \hspace{4em} \text{pref} \leftarrow \text{pref} + 1 \ \{ \text{Advance to next best channel} \}
20: \hspace{3em} \textbf{end if}
21: \hspace{2em} \textbf{end if}
22: \hspace{1em} \textbf{end if}
23: \hspace{1em} \text{r}_n(t) \leftarrow \text{execute\_action}(a_n(t))
24: \hspace{1em} \text{update\_stats}(r_n(t), \hat{\mu}_{n,a_n(t)}, s_{n,a_n(t)})
25: \textbf{end for}

\textbf{note}: \ \hat{\mu}_{n,k} \ \text{is the empirical mean of the reward for user} n \ \text{on channel} k; \\
\ \ \ \ \text{s}_{n,k} \ \text{is the number of times user} n \ \text{has sampled arm} k.
exists $T(\delta)$ such that for all time slots $t > T$, the probability of the system’s being in an SMC is at least $1 - \delta$.

The proof of Theorem 1 consists of two aspects: orthogonality and stability. The first part is easy to verify.

**Proposition 1.** The actions of users applying CSM-MAB are orthogonal (i.e., there is at most one user sampling each channel) for all $t > t_0$ with probability of at least $1 - \delta_0$.

**Proof.** Based on Theorem 1 of [19], the initial configuration reached after running the CFL algorithm is orthogonal with probability 1. The authors provide an upper bound on the distribution of stopping times, $\tau$:

$$P[\tau > k] = \alpha e^{-\gamma k},$$

where $\alpha$ and $\gamma$ are some positive constants. The expected stopping time is therefore upper bounded by $\frac{\alpha e^{-\gamma}}{1 - e^{-\gamma}}$. Thus, setting $t_0 \triangleq 2\frac{\alpha e^{-\gamma}}{1 - e^{-\gamma}}$, the probability of not having reached an orthogonal configuration is at most $\delta_0 \triangleq e^{-2\frac{\alpha e^{-\gamma}}{1 - e^{-\gamma}}}$.

Once the system reaches an orthogonal configuration, every deviation from this configuration is coordinated: a user does not switch to an occupied channel without having coordinated the switch, as defined in Algorithm 1. \qed

### 5.1 Stability and potential

Showing that our system converges to a stable solution is more involved. We begin by defining a potential function for the problem. For any user $n \in \{1, \ldots, N\}$, the potential at time $t$ is defined as follows:

$$\phi_n(t) \triangleq \sum_{k=1}^{K} 1 \{ \mu_{n,k} > \mu_{n,a_n(t-1)} \},$$

where $a_n(t-1)$ is the action taken by user $n$ in the previous time step. In words, the potential is the number of channels user $n$ prefers over her current choice. The system-wide potential is the sum of potentials over all users:

$$\Phi(t) \triangleq \sum_{n=1}^{N} \phi_n(t)$$

An illustration of the potential appears in Tables 1 and 2.

In terms of potential, a configuration is an SMC if no two users can swap channels and decrease their potential by doing so. We note that a stable configuration does not necessarily correspond to zero system-wide potential, since not all users might be able to achieve zero potential simultaneously, depending on network parameters. Also, a system may have several stable configurations, each characterized by a different potential. Nevertheless, observing a system’s
Table 1: Table of users’ channel rankings (first row represents best channel, last row represents worst). Cells highlighted in yellow represent user’s current choice.

|   | U1 | U2 | U3 |
|---|---|---|---|
| 1 | 1 | 2 | 4 |
| 2 | 2 | 1 | 1 |
| 3 | 4 | 3 | 2 |
| 4 | 3 | 4 | 3 |

Table 2: User potentials corresponding to the configuration in Table 1.

| φ1 | φ2 | φ3 |
|----|----|----|
| 3  | 1  | 0  |

potential does provide an indication regarding stability: once a system reaches a stable configuration, its potential will no longer change.

We prove convergence to an SMC by using the potential function, considering three aspects:

1. The maximal potential of a system with $K$ channels and $N$ users is finite and equal to $N(K - 1)$.
2. The potential $\Phi(t)$ is monotonously non-increasing with high probability.
3. Until an SMC is reached, changes in potential are bound to happen within finite time.

We formalize and prove these statements in the sequel.

5.2 Added challenge of learning

Since users’ decisions are guided by UCB indices, and stability is examined with respect to the true reward distribution, users do not always update their choice of channels in a way that matches the ground truth. Thus, the system potential may occasionally increase, due to users’ exploration or inaccurate statistics. In our proof we show that learning does not prevent users from ultimately converging to a stable configuration.

5.3 Proof of Theorem

We begin with a lemma that ensures the monotonicity of the potential.

Lemma 1. For all times $t$ for which $t > \frac{16K}{\Delta_{\min}} \ln t$, if a change in potential occurs, it is a decrease, with probability of at least $1 - 2t^{-4}$.

$\Delta_{\min}$ is a distribution dependent constant. In the appendix we derive an upper bound on the minimal time for which the condition above holds:

$$t_{\min} \leq \frac{M - 1 - \sqrt{(M - 1)^2 - 4M}}{2}, \quad (5)$$
where $M \triangleq \frac{16K}{\Delta_{\min}}$. This bound will enable us to use $t_{\min}$ in the proof.

Next, we introduce a lemma that concerns the ability of a single user to reach the position of the initiator.

**Lemma 2.** If $\phi_n(t) > 0$ for some user $n$, then her probability of becoming the next initiator is at least $\epsilon (1 - \epsilon)^{N - 1}$.

Using Lemma 2 we show another result:

**Lemma 3.** If the system is not in an SMC at some time $t$, then a change in the potential will occur within no more than $t'(\delta_1)$ time slots with probability of at least $1 - \delta_1$.

The exact dependency of $t'$ on $\delta_1$ appears in the appendix, as do the proofs of all lemmas.

The probability of the system’s reaching an SMC within $\tau \triangleq t'N (K - 1)$ time slots after time $t_{\min}$ is at least

$$P_{\text{SMC}} \triangleq \left(1 - \delta_1\right) \left(1 - 2t_{\min}^{-1}\right)^{N(K-1)}.$$

We model the convergence to an SMC using a Markov chain. Let $S_t$ denote the state of the system at time $t$:

$$S_t = \begin{cases} 0 & \text{if in SMC}, \\ 1 & \text{else}. \end{cases}$$

The following holds for the chain’s transition probability:

$$\mathbb{P}[S_t+\tau = 1|S_{t_{\min}} = 0] \geq P_{\text{SMC}},$$

and also

$$\mathbb{P}[S_T = 0|S_{t_{\min}} = 0] \leq (1 - P_{\text{SMC}})^{\left\lfloor \frac{T - t_{\min}}{\tau} \right\rfloor} \forall T > t_{\min} + \tau.$$ 

Defining $\delta \triangleq (1 - P_{\text{SMC}})^{\left\lfloor \frac{T - t_{\min}}{\tau} \right\rfloor}$ completes the proof, and inverting yields

$$T = t_{\min} + \tau \frac{\ln \delta}{\ln (1 - P_{\text{SMC}})}.$$

6 Experiments

In order to demonstrate the merits of our algorithm, we implemented a simulation of a distributed multi-user communication network. The users in our network are synchronized, and time is slotted.

In this network, users cannot communicate with each other directly. However, they can sense the entire frequency range (i.e., listen to all channels). In
addition, they may transmit over a channel of their choice, updating this choice with each time slot.

A user $n$ transmitting over a channel $k$ receives a binary reward, drawn i.i.d. from a Bernoulli distribution with parameter $\mu_{n,k}$. This can be viewed as a form of the classic binary symmetric channel. As far as the different values of the reward parameters go, we ran experiments in two different modes:

1. random: the $\mu_{n,k}$’s are drawn uniformly and independently from the interval $[0,1]$.
2. real-world: users are divided into clusters, and each cluster has a preferred group of channels. This represents a scenario in which users sharing a cluster are geographically close, and experience an interference in part of the frequency range. In real-world wireless communication systems, an agent that does not belong to the network but is transmitting in its vicinity will often cause a similar phenomenon.

We present results obtained in an experiment with $K = 12$ channels and $N = 10$ users. The users are divided into two clusters. Users 1-5 belong to one cluster, and experience an interference in the frequency range of channels 7-12. Users 6-10, on the other hand, experience similar results over the entire frequency range. Our experiments last $T = 120000$ time slots, and results are averaged over 50 repetitions.

Our experiments show several interesting results. First, we compare the performance of our coordinated approach to an approach that applies a learning algorithm directly, without addressing the presence of multiple users. Such an approach may sometimes yield reasonable results when there are few users, but is likely to break down as the network load increases. We show this by comparing the convergence of both policies to a stable state: Figure 3 shows the cumulative number of times our policy “hits” an SMC, together with the number of times a classic UCB policy does so. Our algorithm, CSM-MAB, achieves this very often, while the direct implementation of UCB struggles, especially as time advances. Once user statistics start converging, users are less inclined to move between channels, and the lack of coordination hinders their ability to co-exist. We note that this result was obtained in a slightly longer experiment than others.

Our next result examines the convergence to different SMCs over several repetitions of a certain setup. In this case, the set of SMCs consists of 305 configurations. Naturally, the size of this set depends on the number of users, $N$, the number of channels, $K$, and also on the specific realization of the $\mu_{n,k}$’s. Figure 4 shows that the periods of time users spend in unstable configurations decrease as the experiment advances, and users move between different SMCs, depending on the realization.

The number of times users change their choice of channel is also of interest. Since our goal is stability, we would like the number of policy changes to be small. Figure 5 shows the cumulative number of changes per user, over time. Different users have different patterns, depending on the realization but more importantly on the difficulty of their problem: users that have small differences
between channel characteristics will need more samples in order to differentiate between them, and will therefore experience more policy changes.

Finally, to complement our proof, we provide a visualization of the temporal evolution of the system potential, averaged over several repetitions, in Figure 6. As shown in the proof, the potential decays on average. The shaded area around the plot represents the variance over iterations, which also decays over time. As explained in Section 5.1, the potential does not necessarily decay to zero, but rather to a constant value that represents the potential of one of the SMCs.

7 Conclusion

We have presented an extension of the multi-user MAB problem, for the case of different reward distributions between the users, together with limited information exchange. Using a specialized signalling method, our algorithm enables multiple users to learn network characteristics and converge to an orthogonal configuration that is also a stable marriage. We provide a theoretical analysis of our algorithm’s performance, based on the notion of system potential. Finally, we present the results of an experimental setup and examine different aspects of our approach’s performance.

In the future we intend to extend our work to a dynamic scenario, both in terms of channel characteristics and number of users. The latter should be straightforward due to the minimal inter-dependency of users, while the former will require some adjustment of the learning algorithm.

References

[1] I.F. Akyildiz, L. Won-Yeol, M.C. Vuran, and S. Mohanty. A survey on spectrum management in cognitive radio networks. Communications Magazine, IEEE, 46(4):40–48, April 2008.
Figure 4: Convergence to SMC for different realizations: horizontal axis shows time, vertical axis shows numbering of realizations. White pixels represent unstable configurations, other colors correspond to different SMCs. As time goes by, longer stretches of time are spent in SMCs.

[2] N. Amira, R. Giladi, and Z. Lotker. Distributed weighted stable marriage problem. In Structural Information and Communication Complexity, pages 29–40. Springer, 2010.

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

[4] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. Machine learning, 47(2):235–256, 2002.

[5] P. Auer, N. Cesa-Bianchi, Y. Freund, and R.E. Schapire. The nonstochastic multiarmed bandit problem. SIAM Journal on Computing, 32(1):48–77, 2002.

[6] P. Auer and R. Ortner. UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem. Periodica Mathematica Hungarica, 61(1):55–65, 2010.

[7] O. Avner and S. Mannor. Concurrent bandits and cognitive radio networks. In European Conference on Machine Learning, 2014.

[8] D.P. Bertsekas. The auction algorithm: A distributed relaxation method for the assignment problem. Annals of operations research, 14(1):105–123, 1988.
Figure 5: Number of changes in users’ choice of channels. On the left - a single realization, jumps in plot represent action change points. On the right, we plot the number of changes, averaged over 50 repetitions. User 10 (yellow, top plot) experiences more changes than other users, since the expected value of the reward she receives from channels 2, 8 and 11 is very similar.

Figure 6: Decay of system potential over time, averaged over 50 repetitions. The lines in the left graph represent different realizations, and the shaded area in the right graph represents variance.

[9] K. Cohen, A. Leshem, and E. Zehavi. Game theoretic aspects of the multi-channel aloha protocol in cognitive radio networks. Selected Areas in Communications, IEEE Journal on, 31(11):2276–2288, 2013.

[10] P. Floréen, P. Kaski, V. Polishchuk, and J. Suomela. Almost stable matchings by truncating the gale–shapley algorithm. Algorithmica, 58(1):102–118, 2010.

[11] D. Gale and L.S. Shapley. College admissions and the stability of marriage. American mathematical monthly, pages 9–15, 1962.

[12] A. Garivier and O. Cappé. The KL-UCB algorithm for bounded stochastic bandits and beyond. In Conference On Learning Theory, pages 359–376, 2011.
[13] Y.A. Gonczarowski and N. Nisan. A stable marriage requires communication. *arXiv preprint arXiv:1405.7709*, 2014.

[14] S. Haykin. Cognitive radio: brain-empowered wireless communications. *Selected Areas in Communications, IEEE Journal on*, 23(2):201 – 220, February 2005.

[15] W. Jouini, D. Ernst, C. Moy, and J. Palicot. Multi-armed bandit based policies for cognitive radio’s decision making issues. In *Signals, Circuits and Systems (SCS), 2009 3rd International Conference on*, pages 1–6. IEEE, 2010.

[16] D. Kalathil, N. Nayyar, and R. Jain. Decentralized learning for multi-player multi-armed bandits. In *51st IEEE Conference on Decision and Control*, 2012.

[17] A. Kipnis and B. Patt-Shamir. A note on distributed stable matching. In *IEEE International Conference on Distributed Computing Systems*, 2009.

[18] H.W. Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[19] D.J. Leith, P. Clifford, V. Badarla, and D. Malone. WLAN channel selection without communication. *Computer Networks*, 2012.

[20] A. Leshem, E. Zehavi, and Y. Yaffe. Multichannel opportunistic carrier sensing for stable channel access control in cognitive radio systems. *Selected Areas in Communications, IEEE Journal on*, 30(1):82–95, 2012.

[21] J. Mitola and G.Q. Maguire. Cognitive radio: making software radios more personal. *Personal Communications, IEEE*, 6(4):13 –18, August 1999.
A  A note on rewards

Although we do not address reward maximization explicitly in the paper, it is implicitly taken into account through the use of UCB indices for ranking. By construction, stability is the main goal when applying CSM-MAB, but the fact that users maintain an orthogonal configuration and are guided by UCB indices, contributes to a reward-oriented solution. This is especially evident when comparing the performance of CSM-MAB to that of classical UCB \[4\], in cases in which the number of arms and the number of users are very close. In such scenarios, classical UCB often fails to reach an orthogonal configuration, resulting in substantial reward loss. Figure 7 presents the cumulative reward over time for a scenario in which $N=K=6$, exemplifying this phenomenon.

![Figure 7: Total reward - classical UCB (red) vs. CSM-MAB (blue).](image)

B  Supplementary material: proofs

B.1  Proof of Lemma 1

We would like to show that for all values of $t$ for which $t > \alpha \ln t$, the probability that the potential decreases every time it changes is at least $1 - \frac{4}{t} - \frac{4}{\alpha}$, where $\alpha = \frac{32K}{\Delta_{max}}$.

Given that a change in potential occurs at time $t$, it is guaranteed to result in a potential decrease if it benefits both users. This will happen if both users’ indices, that guide their decisions, are accurate w.r.t the true distribution.

Since we condition on a change in potential,

$$P[\Phi_{Dec}] = 1 - P[\Phi_{Inc}]$$

Let us upper bound $P[\Phi_{Inc}]$. For a user $n$ switching from arm $j$ to arm $i$ at time $t$, $\mu_{n,i} < \mu_{n,j}$,

$$P[\Phi_{Inc}] = P[I_{n,i}(t) \geq I_{n,j}(t) \cap \mu_{n,i} < \mu_{n,j}]$$

18
where \( I_{n,i}(t) \) is user \( n \)'s UCB index of arm \( i \) at time \( t \), defined in the paper in [2]. Following the proof of Theorem 1 of [4],

\[
\Pr[\Phi_{\text{Inc}}] = \Pr[\hat{\mu}_{n,i}(t) + c_{t,s_{n,i}} \geq \hat{\mu}_{n,j}(t) + c_{t,s_{n,j}} \cap \mu_{n,i} < \mu_{n,j}]
\leq 2t^{-4},
\]

provided that

\[
s_{n,i} \geq \frac{8 \ln t}{\Delta_{i,j}^2(n)}, \tag{6}
\]

where \( s_{n,i} \) is the number of times user \( n \) has sampled arm \( i \) up till time \( t \) and \( \Delta_{i,j}(n) \triangleq \mu_{n,i} - \mu_{n,j} \). If condition (6) does not hold, then the UCB index “misleads” user \( n \), causing her to mistakenly favor arm \( i \), despite its lower expected reward. Switching from arm \( j \) to arm \( i \) will result in an increase in potential. However, once she acquires another sample of arm \( i \), its index will decrease. In the meantime, the index of arm \( j \) will increase due to the passing time, and the indices will ultimately reflect the correct preference. This will result in a potential decrease.

The extreme value for (6), i.e., the largest number of required samples, corresponds to the minimal value of \( \Delta_{i,j}(n) \). Let us define:

\[
\Delta_n \triangleq \min_{i,j \in \{1, \ldots, K\}, \ i \neq j} [\mu_{n,i} - \mu_{n,j}]
\]

\[
\Delta_{\text{min}} \triangleq \min_{n \in \{1, \ldots, N\}} \Delta_n
\]

Thus, when all arms have been sampled at least

\[
s_{\text{min}} \triangleq \frac{8 \ln t}{\Delta_{\text{min}}^2} \tag{7}
\]

times, the probability of an increase in potential is extremely small.

In order to allow for the coordination protocol, users do not gather informative samples in every time slot. Instead, they gather at least \( K - 2 \) samples in each super frame, whose length is \( T_{\text{SF}} = 2 + 2(K - 1) = 2K \).

Therefore, taking into account the fact that the sampling condition in (7) must apply for all arms, the condition on \( t \) is

\[
t > K \frac{T_{\text{SF}}}{K - 2} s_{\text{min}} = \frac{16K^2}{(K - 2) \Delta_{\text{min}}^2} \ln t > \frac{16K}{\Delta_{\text{min}}^2} \ln t. \tag{8}
\]

For all times \( t \) for which condition (8) holds, if a change in potential occurs, it is a decrease, with probability of at least \( 1 - 2t^{-4} \).

In our application of this lemma we will use a quantity \( t_{\text{min}} \), which is an upper bound on the minimal value of \( t \) for which (8) holds. We derive this bound using a lower bound on the logarithmic function:

\[
\ln x \geq \frac{x - 1}{x + 1} \quad \forall x > 1.
\]
We incorporate this lower bound into an equation based on (8):
\[
t_{\min} = \frac{16K}{\Delta_{\min}^2} \ln t_{\min} \geq \frac{16K}{\Delta_{\min}^2} \frac{t_{\min} - 1}{t_{\min} + 1}.
\]

Denoting \( M \triangleq \frac{16K}{\Delta_{\min}} \), we continue:
\[
t_{\min} \geq M \frac{t_{\min} - 1}{t_{\min} + 1}
\]
\[
t_{\min}^2 + (1 - M) t_{\min} + M \geq 0.
\]

Our conclusion is that \( t_{\min} \leq M - 1 - \sqrt{(M-1)2-4M} \). Since this expression is finite, we may now use it for our proof.

**B.2 Proof of Lemma 2**

The probability of a specific user becoming the initiator when there are \( \ell \) interested users is
\[
P_s(\epsilon, \ell) \triangleq P \{ \text{specific initiator} | \ell \ \text{interested} \} = \epsilon (1 - \epsilon)^{\ell - 1} \ \forall \ell \in \{1, \ldots, N\}.
\]
The probability is minimized when all \( N \) users would like to tap, yielding the bound \( \epsilon (1 - \epsilon)^{N-1} \).

**B.3 Proof of Lemma 3**

If the system has not reached an SMC, then according to Definition 1, the conditions \( S_1 \), \( S_2 \) hold for at least one pair of users \( n, m \).

According to the definition of Algorithm 1, if \( S_1 \) holds, then user \( n \) will add the channel user \( m \) is sampling to his list of preferred channels with a probability of at least \( 1 - \delta \). Following arguments similar to those presented in the proof of Lemma 1, \( \delta < 2t^{-4} \). If \( S_2 \) holds, user \( m \) will accept user \( n \)'s swap proposal, assuming her statistics are accurate. This, once again, happens with a probability of at least \( 1 - \delta \). Once users \( n \) and \( m \) swap channels, the potential will change.

In the worst case (i.e., largest \( t' \)), user \( m \)'s channel will be the last channel on user \( n \)'s list, and all users higher on the list will decline user \( n \)'s swap proposals. If user \( n \) approaches a different user (whose channel is ranked higher than \( m \)'s), and that user agrees to swap, the potential will also change.

What is left to prove is that the time it shall take user \( n \) to receive the privilege of being initiator is finite. Once \( n \) is appointed the initiator, it will take no more than \( K - 1 \) mini-frames, i.e., \( 2 (K - 1) \) time slots, until she approaches user \( m \) and a swap takes place.

There are two different cases - if \( n, m \) are the only unstable pair, then they will be the only ones interested in becoming the initiators. Furthermore, if only one of them is dissatisfied, then there will only be one user interested
in initiating. In the notation of Lemma 2, this corresponds to $\ell = 2$ or $\ell = 1$, respectively. The probability of exactly one of them becoming the initiator is $P_{1,2} = \min \{\epsilon, 2\epsilon (1 - \epsilon)\}$.

If there are additional unstable pairs, there will be more nominees for initiating. However, not all super frames necessarily result in a decrease in potential - if the initiator only targets channels occupied by “satisfied” users, all her attempts will be rejected. Therefore, we need to address the worst case scenario, in which all $N$ users attempt to initiate, but only one of them is in a position that will actually result in a swap. Based on Lemma 2, the probability of that user emerging as the single initiator is at least $\epsilon (1 - \epsilon)^{N-1}$, for a single super frame. This probability is smaller than $P_{1,2}$ for all $\epsilon, N$, and is therefore the lower bound for the probability of a single initiator with actual capacity for a decrease in potential.

In a time interval of length $t'$, $C = \left\lfloor \frac{t'}{T_{SF}} \right\rfloor$ super frames are initiated. Based on the discussion above, the probability that a single initiator with actual capacity for a decrease in potential does not emerge in a certain SF is less than $1 - \epsilon (1 - \epsilon)^{N-1}$, and the probability that a single initiator does not emerge in the interval is less than $P_C \triangleq \left(1 - \epsilon (1 - \epsilon)^{N-1}\right)^C$. As $t' \to \infty$, so does $C$, and the probability of a single potential-decreasing initiator never emerging decays to zero.

Binding the two aspects of this lemma together, we have that the probability of a single initiator with actual capacity for coordinating a switch emerging in an interval of length $t'$ is at least $1 - P_C$. The probability of a swap between users whose actions do not correspond to a stable configuration is at least $(1 - 2t^{-4})^2$. The combined result: if the system is not in an SMC at time $t$, then a change in the potential will occur within no more than $t'$ time slots with probability of at least $(1 - P_C) (1 - 2t^{-4})^2$, where $P_C \triangleq \left(1 - \epsilon (1 - \epsilon)^{N-1}\right)^{\left\lfloor \frac{t'}{T_{SF}} \right\rfloor}$.

Let us re-write the result for the sake of clarity: if the system is not in an SMC at time $t$, then a change in the potential will occur within no more than $t'(\delta_1)$ time slots with probability of at least $1 - \delta_1$. Developing the previous expression for the probability of a change in potential:

\[
(1 - P_C) (1 - 2t^{-4})^2 = (1 - P_C) (1 - 4t^{-4} + 4t^{-8}) \\
\geq (1 - P_C) (1 - 4t^{-4}) \\
= 1 - P_C - 4t^{-4} + 4P_Ct^{-4} \\
\geq 1 - P_C - 4t^{-4} \\
\geq 1 - P_C - 4t_{\min}^{-4}.
\]

From now on, we denote $\delta_1 = P_C + 4t_{\min}^{-4}$. Using this, we can derive an expression
for $t' (\delta_1)$:

$$P_C = \delta_1 - 4t_{\text{min}}^{-4}$$

$$\left(1 - \epsilon (1 - \epsilon)^{N-1}\right)^\left[\frac{t'}{T_{\text{SF}}}ight] = \delta_1 - 4t_{\text{min}}^{-4}$$

$$\frac{t'}{T_{\text{SF}}} \ln \left(1 - \epsilon (1 - \epsilon)^{N-1}\right) = \ln \left(\delta_1 - 4t_{\text{min}}^{-4}\right)$$

$$t' = T_{\text{SF}} \frac{\ln \left(\delta_1 - 4t_{\text{min}}^{-4}\right)}{\ln \left(1 - \epsilon (1 - \epsilon)^{N-1}\right)}.$$