Journal of Machine Learning Research 21 (2020) 1-23 Submitted 11/19; Published 4/20

Multi-Player Bandits: The Adversarial Case

Pragnya Alatur PRAGNYA.ALATUR@GMAIL.COM

Kfir Y. Levy KFIRYLEVY@TECHNION.AC.IL
Faculty of Electrical Engineering
Technion - Israel Institute of Technology
Haifa, 3200003, Israel

Andreas Krause KRAUSEA@ETHZ.CH
Department of Computer Science
ETH Zurich
8092 Zürich, Switzerland

Editor: Ohad Shamir

Abstract

We consider a setting where multiple players sequentially choose among a common set of actions (arms). Motivated by an application to cognitive radio networks, we assume that players incur a loss upon colliding, and that communication between players is not possible. Existing approaches assume that the system is stationary. Yet this assumption is often violated in practice, e.g., due to signal strength fluctuations. In this work, we design the first multi-player Bandit algorithm that provably works in arbitrarily changing environments, where the losses of the arms may even be chosen by an adversary. This resolves an open problem posed by Rosenski et al. (2016).

Keywords: Multi-Armed Bandits, Multi-Player Problems, Online Learning, Sequential Decision Making, Cognitive Radio Networks

1. Introduction

The Multi Armed Bandit (MAB) problem is a fundamental setting for capturing and analyzing sequential decision making. Since the seminal work of Robbins (1952) there has been a plethora of research on this topic (Cesa-Bianchi and Lugosi, 2006; Bubeck and Cesa-Bianchi, 2012; Lattimore and Szepesvári, 2018), addressing both the stochastic and adversarial MAB settings. In the *stochastic* setting it is assumed that the environment is stationary, namely that except for noisy fluctuations, the environment does not change over time. The *adversarial* setting is more general, and enables to capture dynamic (arbitrarily changing) environments.

Most existing work on MABs considers a single player who sequentially interacts with the environment. Nevertheless, in many real world scenarios, the learner also *interacts with other players*, either collaboratively or competitively. One such intriguing multi-player setting arises in cognitive radio networks, where multiple broadcasters (players) share a common set of transmission channels (arms). In this setting, players incur an extra loss upon colliding (transmitting on the same channel), and communication between players is generally not possible. This challenging setting has recently received considerable attention, see Avner and Mannor (2014); Rosenski et al. (2016); Bistritz and Leshem (2018).

Despite impressive progress on multi-player Bandit problems, existing work only addresses the stochastic setting where the environment is stationary. Yet, this may not capture common phenomena in cognitive radio networks, such as channel breakdowns or signal strength fluctuations due to changing environmental conditions.

In this work, we address the adversarial multi-player MAB setting, and provide the *first efficient algorithm with provable guarantees.* This resolves an open problem posed by Rosenski, Shamir, and Szlak (2016). Assuming that K players choose among a set of N arms, we provide an efficient method with *two* variants that ensure respective regret bounds of $\tilde{O}(K^{4/3}N^{2/3}T^{2/3})$, and $\tilde{O}(K^{4/3}N^{1/3}T^{2/3})$[1].

Our key algorithmic technique is to imitate the idealized case where there is full communication between the players. Then, to address the no-communication constraint, we enforce the players to keep the same arms within long periods of time (blocks). This gives them the chance to coordinate among themselves via a simple protocol that uses collisions as a primitive, yet effective manner of coordination. We suggest two different coordination schemes, yielding two different guarantees.

Related Work: The stochastic multi-player MAB problem has been extensively investigated in the past years. The majority of work on this topic assumes that players may communicate with each other (Lai et al., 2008; Liu and Zhao, 2010; Vakili et al., 2013; Liu et al., 2013; Avner and Mannor, 2016, 2018). The more realistic "no-communication" setting is discussed in (Anandkumar et al., 2011; Avner and Mannor, 2014; Rosenski et al., 2016; Bistritz and Leshem, 2018). Avner and Mannor (2014) are the first to provide regret guarantees for the "no-communication" stochastic setting, establishing a bound of $O(T^{2/3})$. This has been later improved by Rosenski et al. (2016), who establish a constant regret (*independent* of T) for the case where there exists a fixed gap between mean losses. Recently, Bistritz and Leshem (2018) have explored a more challenging setting, where each player has a different loss vector for the arms. They provide an algorithm that ensures $O(\log^2 T)$ regret for this setting.

The stochastic setting where the number of players may change throughout the game is addressed by Rosenski et al. (2016), where a regret bound of $O(\sqrt{T})$ is established. Avner and Mannor (2014) also discuss this case and provide an algorithm that in some scenarios ensures an $O(T^{2/3})$ regret.

Different multi-player adversarial MAB settings are explored in Awerbuch and Kleinberg (2008); Cesa-Bianchi et al. (2016). Nevertheless, these works allow players to communicate, and do not assume a "collision loss". There also exists rich literature on Combinatorial bandit settings Uchiya et al. (2010); Audibert et al. (2013); Combes et al. (2015), where several players may fully communicate to jointly choose a set of arms in each round. Nevertheless, these works do not address the "no communication" setup. In a contemporary unpublished work (Bubeck et al. (2019)), a regret of $O(\sqrt{T})$ is obtained for the case of two-players MAB setting. However, their paper does not establish any guarantees for the general multi-players MAB setting. Conversely, we provide an efficient algorithm which holds for the general multi-player setting.

2. Background and Setting

2.1. Background

The N-armed bandit setting can be described as a repeated game over T rounds between a *single* player and an adversary. At each round $t \in [T]$ (we denote $[N] := \{1, \ldots, N\}$, for any $N \in \mathbb{Z}^+$),

[1] Using $\tilde{O}(\cdot)$ we ignore logarithmic factors in T, N.

the player chooses an arm $I^t \in [N]$ and the adversary independently chooses a loss for each arm $l_i^t \in [0,1],\ \forall i \in [N]$. Then, the player incurs the loss of the chosen arm $l_{I^t}^t$, and gets to view the loss of this arm only (bandit feedback). The goal of the player is to minimize the *regret*, defined as $R_T := \sum_{t=1}^T l_{I^t}^t - \min_{i\in[N]} \sum_{t=1}^T l_i^t$. We are interested in learning algorithms that ensure an expected regret which is *sublinear* in T, here expectation is with respect to possible randomization in the player's strategy as well as in the choices of the adversary.

The seminal work of Auer et al. (2002) presents an algorithm that achieves an optimal regret bound of $O(\sqrt{TN\log N})$ for this setting. Their algorithm, called EXP3, devises an unbiased estimate of the loss vector in each round, $\left\{\widetilde{l}_i^t\right\}_{i\in[N]}$. These are then used to pick an arm in each round by sampling $I^t \propto \exp(-\eta \sum_{\tau=1}^{t-1} \widetilde{l}_i^\tau)$, for any arm $i \in [N]$.

2.2. K-Player MAB Setting

We consider a repeated game of T rounds between K players and an adversary in the N-armed bandit setting. For now assume that each player has a unique *rank* in $[K]$, and that each player knows her own rank (but does not necessarily know the rank of other players)[2]. We also refer to the player with rank k as “player k". Now at each round $t \in [T]$,

1. each player $k \in [K]$ chooses an arm $I_k^t \in [N]$
2. the adversary independently chooses a loss for each arm $l_i^t \in [0,1],\ \forall i \in [N]$
3. for each player $k \in [K]$ one of two cases applies,

   Collision: if another player chose the same arm, i.e., $\exists m \neq k$ such that $I_k^t = I_m^t$, then player k gets to know that a collision occurred, and incurs a loss of 1.
   No Collision: if there was no collision, player k incurs the loss of the chosen arm $l_{I_k^t}^t$, and gets to view the loss of this arm only (bandit feedback).

We emphasize that at each round all players play *simultaneously*. We further assume that communication between players is not possible. Finally, note that the ability to distinguish between collision and non-collision is a reasonable assumption when modeling cognitive radio networks and was also used in previous work, e.g. by Rosenski et al. (2016).

Our assumption is that the players are *cooperative* and thus their goal is to obtain low regret together with respect to the K *distinct* best arms in hindsight. Let $C_k^t \in \{0,1\}$ be an indicator for whether player k collided at time t ($C_k^t = 1$) or not ($C_k^t = 0$). With this, we define the regret R_T after T rounds as follows:

$$R_T := \underbrace{\sum_{t=1}^T \sum_{\substack{k=1,\\ C_k^t=0}}^K l_{I_k^t}^t}_{\text{no collisions}} + \underbrace{\sum_{t=1}^T \sum_{k=1}^K C_k^t}_{\text{collisions}} - \min_{\substack{i_1,\ldots,i_K\in[N]\\ i_m\neq i_n,\forall m\neq n}} \sum_{t=1}^T \sum_{k=1}^K l_{i_k}^t .$$

We are interested in learning algorithms that ensure an expected regret that is sublinear in T.

[2] As we show in Section 3, such ranking can be achieved by running a simple procedure at the beginning of the game (see Algorithm 2).

Staying quiet: For simplicity, we assume that a player may choose to stay *quiet*, i.e., not choose an arm, in any given round. By staying quiet she does not cause any collisions, but she will still suffer a loss of 1 for that round. This is a reasonable assumption in cognitive radio applications, as a user may choose to not transmit anything. In Appendix A.3 we show how to relax this assumption.

Adversary: Our analysis focuses on *oblivious* adversaries, meaning that the adversary may know the strategy of the players, yet he is limited to choosing the loss sequence before the game starts.

Further assumptions: We assume that every player knows T, the number of arms N, the number of players K and that $K < N$ and $N < T$. Furthermore, we assume that the set of players is fixed and no player enters or exits during the game. Using a standard doubling technique we may extend our method to the case where T is unknown.

3. Multi-Player MABs

In this section, we present our algorithm for the N-armed bandit setting with K players. We first discuss an idealized setting in which players can fully communicate and then build our K-player communication-free algorithm on top of that. For ease of exposition we mainly discuss a variant of our method ensuring a regret of $\tilde{O}(K^{4/3}N^{2/3}T^{2/3})$ (see Thm. 2). We then explain how to improve this rate to $\tilde{O}(K^{4/3}N^{1/3}T^{2/3})$ (see Thm. 4) by using a more sophisticated coordination mechanism.

3.1. Idealized, Communication-Enabled Setting

In this setting, the players can fully communicate and thus, they can coordinate their choices to avoid collisions, resulting in a collision loss of 0, i.e., $\sum_{t=1}^{T}\sum_{k=1}^{K} C_k^t = 0$. In this case, the K players would behave as a single player who chooses K distinct arms in each round and aims to obtain low regret with respect to the K best arms in hindsight.

Such a hypothetical player (let us call her *K-Metaplayer*) chooses K distinct arms $I^t := \{I_1^t, ..., I_K^t\}$ in each step t and receives the losses of these arms[3]. Her regret after T rounds is $R_T^{\text{meta}} := \sum_{t=1}^{T}\sum_{k=1}^{K} l_{I_k^t}^t - \min_{\substack{i_1,...,i_K \in [N] \\ i_m \neq i_n, \forall m \neq n}} \sum_{t=1}^{T}\sum_{k=1}^{K} l_{i_k}^t$. We will see soon show a simple adaptation of the EXP3 algorithm ((Auer et al., 2002)) yields low regret for the K-Metaplayer. First, let us define the set of *meta-arms* $\mathcal{M}$ as follows:

$$\mathcal{M} := \left\{ \{i_1, ..., i_K\} \subseteq [N] \middle| i_m \neq i_n \text{ for any } m \neq n \right\} .$$

We further define the loss $\mathbf{l_I^t}$ of a meta-arm $I \in \mathcal{M}$ at time t as $\mathbf{l_I^t} := \sum_{k \in I} l_k^t$.

From this, it is immediate that the best meta-arm in hindsight w.r.t. losses $(\mathbf{l_I^t})_{I \in \mathcal{M}, t \in [T]}$ consists of the K best arms in hindsight w.r.t. $(l_i^t)_{i \in [N], t \in [T]}$.

With these definitions, one can view the K-Metaplayer as a player who chooses one meta-arm in each step, receives that meta-arm's loss and aims to obtain low regret with respect to the best meta-arm in hindsight. This exact setting was described and analyzed by Uchiya et al. (2010), who present and analyze an adaptation of EXP3 to obtain a low-regret K-Metaplayer algorithm.

Feedback model: In Alg. 1 we describe and analyze a variant of the above setting. We assume a more restrictive bandit feedback, where the Metaplayer gets to view only a *single arm chosen*

[3] Actually, as we will soon see, we analyze a slightly different setting where the Metaplayer gets to view only a single arm chosen uniformly at random from I^t.

Algorithm 1 K-Metaplayer algorithm (Input: η)

```
Input: $\eta$
for $t = 1$ to $T$ do
    Set cumulative loss estimate $\widetilde{\mathbf{L^t_I}} = \sum_{\tau=1}^{t-1} \sum_{i \in I} \widetilde{l_i^\tau}$, for all meta-arms $I \in \mathcal{M}$
    Set probability $p^t(I) = \frac{e^{-\eta \widetilde{\mathbf{L^t_I}}}}{\sum_{J \in \mathcal{M}} e^{-\eta \widetilde{\mathbf{L^t_J}}}}$, for all meta-arms $I \in \mathcal{M}$
    Sample meta-arm $I^t = \{I_1^t, ..., I_K^t\}$ at random according to $P^t = (p^t(I))_{I \in \mathcal{M}}$
    Pick one of the $K$ arms $J^t \sim \text{Uniform}(I_1^t, ..., I_K^t)$
    Choose arms $I_1^t, ..., I_K^t$ in the game, suffer losses $l_{I_1^t}^t, ..., l_{I_K^t}^t$ and observe $l_{J^t}^t$
    Set loss estimate $\widetilde{l_i^t} = K \cdot \frac{l_i^t}{\sum_{i \in I \in \mathcal{M}} p^t(I)} \cdot \mathbb{I}_{\{J^t = i\}}$, for all arms $i \in [N]$
end for
```

uniformly at random (u.a.r.) among $I^t := \{I_1^t, ..., I_K^t\}$. As we show later, this serves as a building block for our algorithm in the more realistic *no-communication* setting. The following Lemma states the guarantees of Algorithm 1,

Lemma 1 *Employing the K-Metaplayer algorithm (Alg. 1) with* $\eta = \sqrt{\frac{\log \binom{N}{K}}{KTN}}$ *guarantees a regret bound of* $\mathbb{E}[R_T^{meta}] \leq 2\sqrt{KTN \log \binom{N}{K}} \leq 2K\sqrt{TN \log N}$. *We defer the proof to Appendix A.1.*

Note that the above bound is worse by a factor of $\sqrt{K}$ compared to the bound appearing in Uchiya et al. (2010). This is since we consider a more restrictive feedback model (i.e., viewing a single arm rather than K arms in each round).

Together as one K-Metaplayer Let us turn our attention back to the K players in an idealized setting with *full communication*. How do the players need to play in order to behave as the K-Metaplayer in Alg. 1?

We suggest to do so by assigning roles as follows: Player 1 takes the role of a global *coordinator*, who decides which arm each of the K players should pick. She samples K arms in each step using the metaplayer algorithm, chooses one out of those K u.a.r. for herself and assigns the rest to the other players. She then communicates to the other players what arms she has chosen for them. Players $2, ..., K$ simply behave as *followers* and accept whatever arm the coordinator chooses for them. With this, they are playing exactly as the metaplayer from Algorithm 1 and their regret would be bounded according to Lemma 1.

Note that the coordinator samples K arms but receives feedback only for *one* of them. This is the reason behind the feedback model considered in Alg. 1. Also, note that in this case, the coordinator is the only player that actually "learns" from the feedback. All other players follow the coordinator and ignore their loss observations.

3.2. Real, Communication-Free Setting

In the previous section, we described and analyzed an idealized setting where all players can fully communicate and can therefore act as a single metaplayer. Then we have shown that by assigning Player 1 the role of a global *coordinator*, and the rest of the players being *followers*, we can exactly imitate the metaplayer algorithm. This strategy however, requires full communication. Here, we show

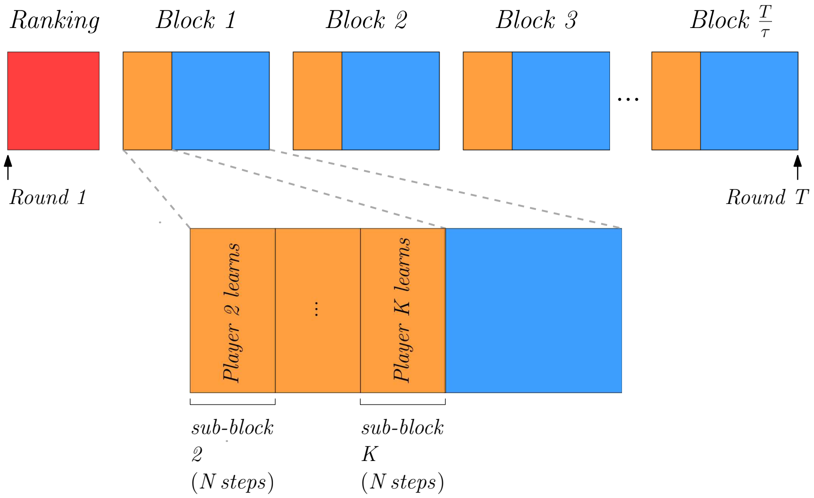


Figure 1: Illustration of the K-player algorithm. The upper part illustrates the timeline of the algorithm and the lower part shows the close-up view of a single block in the algorithm. *Coordinate* phases are marked in orange and *Play* phases are shown in blue. At the beginning of the algorithm, the players compute a ranking (red). This will be discussed further below.

how to build on these ideas to devise an algorithm for the realistic “no-communication" setting. Our C&P (Coordinate & Play) algorithm is depicted in Figure 1, as well as in Alg. 3, and 4. Its guarantees are stated in Theorem 2 and at the end of this section we discuss an efficient implementation of our method.

Our method builds on top of the idealized scheme, with two additional ideas.

Infrequent switches: In order to give players the opportunity to coordinate, we prevent them from frequently switching their decisions. Concretely, as is described in Fig 1, instead of sampling K arms in each round, the coordinator (as well as the followers) keeps the same K arms for a block of τ consecutive rounds. The coordinator (Alg. 3) runs a *blocked version* of the K-metaplayer algorithm (Alg. 1): In each block, the coordinator samples an arm according to Alg. 1, but stays on that arm for the entire block. Then she feeds the *average* loss of that arm back into Alg. 1 to update her loss estimates. While these blocks enable coordination, they cause degradation to the regret guarantees (Dekel et al., 2012). We elaborate on that in the analysis.

Coordinate and Play: We depict the timeline of our algorithm in Figure 1. As can be seen, we divide each block into two phases: *Coordinate* phase (orange), and *Play* phase (blue). At the beginning of each block, the coordinator picks K arms according to the blocked version of the K-metaplayer algorithm. Then, during *Coordinate*, the coordinator assigns an arm to each of the $K-1$ followers. Thus, the Coordinate phase is further divided into $K-1$ sub-blocks $2, ..., K$ (Fig. 1, bottom part). At sub-block k, the k'th follower gets assigned to an arm by a protocol that uses collisions as a primitive, yet efficient, manner of coordination.

This protocol (see Alg. 3, and Alg. 4) is very simple: during sub-block k, the coordinator stays on the arm for player k (a follower). Player k tries out all arms in a round-robin fashion, until she collides with the coordinator. At this point, player k learns her arm and the coordinator can repeat this procedure with the other players. While player k is trying to find her arm, all other followers will stay quiet. Since each follower needs at most N trials, all followers will have learned their arms after $(K-1)\cdot N$ rounds.

After *Coordinate*, each player has learned her arm. During *Play*, all players stay on their arms for the remaining steps of the block. At the end of the block, the coordinator uses the feedback she has collected in order to update her loss estimates.

If T is not divisible by τ, the players will play for $\lfloor \frac{T}{\tau} \rfloor$ blocks and choose arms uniformly at random for the remaining steps. This will increase the regret by at most $K\tau$.

Ranking: So far we assumed that the players have unique ranks in $[K]$. They can compute the ranking by using a scheme that we adopt from Rosenski et al. (2016). The idea is playing a "Musical Chairs game" on the *first* K arms $\{1,\ldots,K\}$ for T_R rounds: A player chooses arms uniformly at random until she chooses an arm i without colliding. At this point, that player becomes the owner of arm i and will receive the rank i. This player i then just stays on arm i for the remaining of the T_R rounds. We will set T_R in a way that the ranking completes successfully with high probability.

Theorem 2 *Suppose that the K players use our C&P Algorithm. Meaning, they first compute a ranking using Algorithm 2 with $T_R = K \cdot e \cdot \log T$. Afterwards, player 1 will act as coordinator and play according to Algorithm 3. The other players will behave as followers and run Algorithm 4. Then, the expected regret of the K players is bounded as follows,*

$$\mathbb{E}[R_T] \;\leq\; 4K^{4/3}N^{2/3}(\log N)^{1/3}T^{2/3} + 2K^2 \cdot e \cdot \log T \;,$$

for block size $\tau = \left(\frac{K^2 NT}{\log N}\right)^{1/3}$ and $\eta = \sqrt{\frac{\log\binom{N}{K}}{\frac{T}{\tau}KN}}$.

Remark: In the variant that we present and analyze above, the coordinator requires N steps in each sub-block in order to communicate an arm in $[N]$ to the follower. Nevertheless, one requires roughly $\log_2 N$ bits to communicate a number in $[N]$. This observation can be used to design a coordination mechanism that requires only $\log_2 N$ rounds in each sub-block. This leads to an improved regret bound of $\tilde{O}(K^{4/3}N^{1/3}T^{2/3})$. See Section. 3.3 for more details.

Proof [Proof of Theorem 2] By setting the length of the ranking phase $T_R = K \cdot e \cdot \log T$, the ranking completes after T_R rounds with probability at least $1 - \frac{K}{T}$ (see Appendix A.2 for the derivation).

Case 1: Ranking unsuccessful With probability at most $\frac{K}{T}$, the players do not succeed in computing a ranking. The worst regret that they could obtain in the game is KT.

Case 2: Ranking successful In the idealized setting with communication from the previous Section 3.1, the Coordinate phase would not be necessary. In that case, Algorithms 3 and 4 together are just the result of applying the blocking technique to the K-Metaplayer algorithm 1. This can be analyzed using the following Theorem from Dekel et al. (2012),

Algorithm 2 C&P Ranking

```
Input: $T_R$
for $t = 1$ to $T_R$ do
    Choose arm $r \sim$ Uniform$(1, ..., K)$
    if I did not collide then    ▷ My rank is $r$
        Choose arm $r$ for the remaining of the $T_R$ rounds and return.
    end if
end for
```

Algorithm 3 C&P Coordinator algorithm

```
Input: $\eta$, block size $\tau$
for block $b = 1$ to $\frac{T}{\tau}$ do
    Choose K arms according to the metaplayer
    Set cumulative loss estimate $\widetilde{\mathbf{L^b_I}} = \sum_{t=1}^{b-1} \sum_{i \in I} \widetilde{l^t_i}$, for all meta-arms $I \in \mathcal{M}$
    Set probability $p^b(I) = \frac{e^{-\eta \widetilde{\mathbf{L^b_I}}}}{\sum_{J \in \mathcal{M}} e^{-\eta \widetilde{\mathbf{L^b_J}}}}$, for all meta-arms $I \in \mathcal{M}$
    Choose meta-arm $\widetilde{J^b} = \{\widetilde{J^b_1}, ..., \widetilde{J^b_K}\}$ at random according to $P^b = (p^b(I))_{I \in \mathcal{M}}$
    Let $\widetilde{I^b} = (\widetilde{I^b_1}, ..., \widetilde{I^b_K})$ be a uniform random permutation of $\widetilde{J^b}$
    Coordinate
    for sub-block $r = 2$ to $K$ do    ▷ Each sub-block has exactly $N$ steps
        Choose $I^t_1 = \widetilde{I^b_r}$ in steps $t$ until collision
        After collision, choose $I^t_1 = \widetilde{I^b_1}$ for the remaining steps $t$ of sub-block $r$
    end for
    Play
    Choose arm $I^t_1 = \widetilde{I^b_1}$ for remaining steps $t$ of block $b$
    Feed average loss of arm $\widetilde{I^b_1}$ back to the metaplayer
    Set $\widehat{l^b_i} = \sum_{t=(b-1)\cdot\tau+1}^{b\cdot\tau} \mathbb{I}_{\{I^t_1 = i\}} \cdot l^t_i$, for all arms $i \in [N]$
    Set loss estimate $\widetilde{l^b_i} = K \cdot \frac{\frac{1}{\tau}\widehat{l^b_i}}{\sum_{i \in I \in \mathcal{M}} p^b(I)} \cdot \mathbb{I}_{\{\widetilde{I^b_1} = i\}}$, for all arms $i \in [N]$
end for
```

Algorithm 4 C&P Follower algorithm

```
Input: block size $\tau$, rank $r$
for block $b = 1$ to $\frac{T}{\tau}$ do
    Coordinate
    Stay quiet during sub-blocks $2, ..., r-1$    ▷ Each sub-block has exactly $N$ steps
    During sub-block $r$, explore arms in a round-robin fashion until collision. $\widetilde{I^b_r}$ is the arm on
        which the collision occurred. Choose $\widetilde{I^b_r}$ for remaining steps of sub-block $r$.
    Stay quiet during remaining sub-blocks $r+1, ..., K$
    Play
    Choose $I^t_r = \widetilde{I^b_r}$ for remaining steps $t$ of block $b$
end for
```

Theorem 3 *(Dekel et al. (2012)) Let $\mathcal{A}$ be a bandit algorithm with expected regret bound of $R(T)$. Then using the blocked version of $\mathcal{A}$ with a block of size τ gives a regret bound of $\tau R(T/\tau) + \tau$.*

The term τ above accounts for the additional regret in case T is not divisible by τ. Since we have K players, we will replace that term by $K\tau$. Hence, by applying the above theorem to the regret bound from Lemma 1, we obtain that the regret of the K players in a setting with communication would be $C \cdot T^{1/2}\tau^{1/2} + K\tau$ for $C = 2\sqrt{KN \log \binom{N}{K}} \leq 2K\sqrt{N \log N}$.

In the real setting without communication, the Coordinate phase is needed and takes $(K-1)\cdot N$ steps. During the Coordinate phase in one block, each player adds at most $(K-1)\cdot N$ to the total regret, either by staying quiet (loss 1) or by not choosing the optimal arm (round-robin exploration). Thus, the Coordinate phase increases the total regret by at most $\frac{T}{\tau}\cdot(K-1)\cdot N\cdot K$.

Finally, the ranking algorithm adds $K\cdot T_R = K^2\cdot e\cdot\log T$ to the regret. Put together, the expected regret of the K players, assuming that ranking was successful (we denote this event by $\mathcal{S}$), is bounded as follows:

$$\mathbb{E}[R_T|\mathcal{S}] \;\le\; \underbrace{C\cdot T^{1/2}\tau^{1/2} + K\tau}_{\text{Thm. 3 + Lemma 1}} + \underbrace{\frac{T}{\tau}\cdot K^2 N}_{\text{Coordinate}} + \underbrace{K^2\cdot e\cdot\log T}_{\text{Ranking}} \;\le\; \tilde{O}(K^{4/3}N^{2/3}(\log N)^{1/3}T^{2/3})\;,$$

where in the second line we use $C = 2K\sqrt{N\log N}$ which holds by Lemma 1; we also take $\tau = \left(\frac{K^2\cdot N\cdot T}{\log N}\right)^{1/3}$ and $\eta = \sqrt{\frac{\log\binom{N}{K}}{\frac{T}{\tau}KN}}$. Furthermore, we use $K < T$.

Combining the results from cases 1 and 2 with $T_R = K\cdot e\cdot\log T$, gives the following bound:

$$\mathbb{E}[R_T] \;=\; \underbrace{Pr[\mathcal{S}]}_{\le 1}\cdot\mathbb{E}[R_T|\mathcal{S}] + \underbrace{Pr[\mathcal{S}^c]}_{\le\frac{K}{T}}\cdot\underbrace{\mathbb{E}[R_T|\mathcal{S}^c]}_{\le K\cdot T} \;\le\; 4K^{4/3}N^{2/3}(\log N)^{1/3}T^{2/3} + 2K^2 e\log T\;,$$

where $\mathcal{S}$ denotes the event where ranking is successful, and $\mathcal{S}^c$ is its complement. ■

Remark: So far we assumed that the players need to stay quiet during the Coordinate phase, but this assumption is actually not necessary. In Appendix A.3 we show how to relax this assumption.

Efficient Implementation: As the number of meta-arms is exponential $|\mathcal{M}| = \Theta(N^K)$, sampling a meta-arm directly according to Alg. 3 can be very slow. Nevertheless, as we show in Section 4, we can reduce the computational complexity to $O(NK)$. The key insight that enables an efficient implementation is that the sampling distribution may be described as a K-DPP. This allows us to use powerful sampling mechanisms to reduce the computational cost to $O(NK)$ in any block. Similarly, computing the marginal probability $\sum_{i\in I\in\mathcal{M}} p^b(I)$ for any fixed arm $i\in[N]$ can be done in $O(NK)$.

3.3. Improved Coordination Scheme and Regret Bounds

In section 3, we discussed our C&P algorithm which achieves low regret for the K-player setting. C&P is a blocked algorithm and consists of a *Coordinate* and a *Play* phase in each block (see Fig. 1). In this section, we will discuss a more efficient scheme to shorten the *Coordinate* phase. Concretely, we will show how to reduce the size of each sub-block in *Coordinate* from N to $\lceil\log_2(N+1)\rceil$ rounds.

In C&P the *Coordinate* part between coordinator and player k (follower) in block b works as follows:

1. Coordinator chooses arm $\widetilde{I}^b_k \in [N]$ for player k
2. In sub-block k, the coordinator stays on arm $\widetilde{I}^b_k$ until a collision. At the same time, player k explores arms in a round-robin fashion until collision.

The collision happens on arm $\widetilde{I_k^b}$ and at this point, player k learns her arm. Using the round-robin exploration of arms, player k will learn her arm in at most N steps.

Instead of doing a naive round-robin exploration on the arms, we can have a more efficient coordination scheme using binary encoding. The idea is that in order to encode any number in $[N]$ (we assume $N > 1$), we need at most $\lceil \log_2(N+1) \rceil$ bits. This enables to reduce the length of each sub-block from N to $\lceil \log_2(N+1) \rceil$. Next we elaborate on this efficient coordination scheme.

Based on the above idea, we propose the following coordination scheme using sub-blocks of size $\lceil \log_2(N+1) \rceil$: In a sub-block k of block b, the coordinator first computes the binary representation of $\widetilde{I_k^b} \in [N]$, which is the arm that she has chosen for player k. Then, coordinator and player k iterate over arms $1, ..., \lceil \log_2(N+1) \rceil$ in an increasing order. In step $i \in \{1, ..., \lceil \log_2(N+1) \rceil\}$ of the iteration, if the binary representation of $\widetilde{I_b^k}$ has a "1" at position i, the coordinator will choose arm i, otherwise she will stay quiet. Player k will choose arm i in step i of the iteration and interpret a collision (no collision) as a "1" ("0"). At the end of sub-block k, player k can reconstruct the binary number to obtain the value of $\widetilde{I_k^b}$ from that.

With this, we reduce the size of each sub-block in *Coordinate* from N to $\lceil \log_2(N+1) \rceil$ rounds. The next theorem demonstrates the affect of this coordination scheme on the regret of C&P.

Theorem 4 *Suppose that the K players use our C&P Algorithm with the improved coordination scheme we describe above. Meaning, they first compute a ranking using Algorithm 2 with $T_R = K \cdot e \cdot \log T$. Afterwards, player 1 will act as coordinator and play according to Algorithm 3. The other players will behave as followers and run Algorithm 4. Then, the expected regret of the K players is bounded as follows,*

$$\mathbb{E}[R_T] \;\leq\; 7K^{4/3}N^{1/3}(\log N)^{2/3}T^{2/3} + 2K^2 e \log T \;,$$

for block size $\tau = \left(\frac{4 \cdot K^2 \cdot \log N \cdot T}{(\log 2)^2 \cdot N}\right)^{1/3}$ and $\eta = \sqrt{\frac{\log \binom{N}{K}}{\frac{T}{\tau} KN}}$.

Proof [Proof of Theorem 4] As the following analysis is based on the proof for Theorem 2, we recommend the reader to take a look at that proof first.

In the original C&P the regret accounted for the *Coordinate* phases was $\frac{T}{\tau}K^2N$. With the improvement, the regret for *Coordinate* becomes $\frac{T}{\tau}K^2\lceil \log_2(N+1) \rceil \;\leq\; 2 \cdot \frac{T}{\tau}K^2 \log_2 N$ (for $N > 1$). Plugging this into the expression for $\mathbb{E}[R_T|\mathcal{S}]$ (see Theorem 2), we obtain:

$$\begin{aligned}\mathbb{E}[R_T|\mathcal{S}] \;&\leq\; \underbrace{C \cdot T^{1/2}\tau^{1/2} + K\tau}_{\text{Thm. 3 + Lemma 1}} + \underbrace{2\frac{T}{\tau}K^2 \log_2 N}_{\text{Coordinate}} + \underbrace{K^2 \cdot e \cdot \log T}_{\text{Ranking}} \\ &\leq 7K^{4/3}N^{1/3}(\log N)^{2/3}T^{2/3} + K^2 \cdot e \cdot \log T \;,\end{aligned}$$

where in the second line we use $C = 2K\sqrt{N \log N}$ (by Lemma 1), $\tau = \left(\frac{4 \cdot K^2 \cdot \log N \cdot T}{(\log 2)^2 \cdot N}\right)^{1/3}$, $\eta = \sqrt{\frac{\log \binom{N}{K}}{\frac{T}{\tau} KN}}$ and the log base change $\log_2(N) = \frac{\log N}{\log 2}$. Finally, we also use $K < T$.

As in Theorem 2, combining the results for the two cases, i.e. $\mathcal{S}$ and $\mathcal{S}^c$ with $T_R = K \cdot e \cdot \log T$, we obtain the following improved regret bound:

$$\mathbb{E}[R_T] \;=\; \underbrace{Pr[\mathcal{S}]}_{\leq 1} \cdot \mathbb{E}[R_T|\mathcal{S}] + \underbrace{Pr[\mathcal{S}^c]}_{\leq \frac{K}{T}} \cdot \underbrace{\mathbb{E}[R_T|\mathcal{S}^c]}_{\leq K \cdot T} \;\leq\; 7K^{4/3}N^{1/3}(\log N)^{2/3}T^{2/3} + 2K^2 e \log T \;.$$

■

4. Efficient sampling from the K-Metaplayer's distribution using K-DPPs

In this section, we will discuss how the coordinator can efficiently sample K arms and compute marginal probabilities in Alg. 3 using K-DPPs. We will first give some background on DPPs and K-DPPs before explaining how to use them for our case.

DPPs (Determinantal Point Processes) are probabilistic models that can model certain probability distributions of the type $\mathcal{P} : 2^{\mathcal{Y}} \rightarrow [0,1]$, where $\mathcal{Y} = [N]$ and $2^{\mathcal{Y}}$ is the power set of $\mathcal{Y}$.[4] Hence, a DPP samples subsets over a ground set $\mathcal{Y}$. In general, a DPP $\mathcal{P}$ is specified by a Kernel matrix (see definition 2.1 of Kulesza and Taskar (2012)). L-Ensembles are a specific type of DPPs and we will focus only on those since this is what we will need for the coordinator algorithm. An L-Ensemble DPP $\mathcal{P}$ is defined by a $N \times N$-Kernel matrix L as follows (see definition 2.2 of Kulesza and Taskar (2012)):

$$\mathcal{P}(\mathbf{Y} = Y) \propto \det(L_Y) \quad (Y \subseteq \mathcal{Y}, \mathbf{Y} \text{ is a random variable specifying the outcome of the DPP.})$$

L_Y is the submatrix of L obtained by keeping only the rows and columns indexed by Y. The only restriction on L is that it needs to be symmetric and positive semidefinite.

K-DPPs define probability distributions over subsets of size K, while the outcome set of a DPP can have any size. A K-DPP $\mathcal{P}^K$ is specified by a $N \times N$-Kernel matrix L as follows (see definition 5.1 of Kulesza and Taskar (2012)):

$$\mathcal{P}^K(\mathbf{Y} = Y) = \frac{\det(L_Y)}{\sum_{Y' \subseteq \mathcal{Y}, |Y'| = K} \det(L_{Y'})}$$

As before, L needs to be positive and semidefinite. For DPPs and K-DPPs, sampling and marginalization can be done efficiently. Because of this, K-DPPs were appealing to us as they would allow us to efficiently sample a set of exactly K *distinct* arms, which is what we need for the coordinator. We will now see how we can model the coordinator's probability distribution over meta-arms as a K-DPP, i.e. we will determine how L needs to be set.

Let us first recall the coordinator's probability for meta-arms. For this, let $\widetilde{L_i^b} = \sum_{\tau=1}^{b-1} \widetilde{l_i^\tau}$ denote the cumulative loss estimate for any arm $i \in \mathcal{Y}$ in block b. And let $\widetilde{\mathbf{L_I^b}} = \sum_{i \in I} \widetilde{L_i^b}$ be the cumulative loss estimate for any meta-arm $I \in \mathcal{M}$ in block b. The probability that the coordinator chooses $I \in \mathcal{M}$ in block b, is:

$$p^b(I) = \frac{e^{-\eta \widetilde{\mathbf{L_I^b}}}}{\sum_{J \in \mathcal{M}} e^{-\eta \widetilde{\mathbf{L_J^b}}}} \qquad \text{(see in Alg. 3)}$$

[4] In general, $\mathcal{Y}$ does not need to be discrete. For more information on the continuous case, please refer to Kulesza and Taskar (2012).

For our K-DPP, $\mathcal{Y} = [N]$ is the ground set and $\mathcal{M}$ the set of outcomes. For block b, let the $N \times N$-Kernel matrix L^b be defined as follows:

$$L^b_{i,j} = \begin{cases} e^{-\eta \widetilde{L^b_i}}, i = j \\ 0, \qquad\qquad i \neq j \text{ (diagonal matrix)} \end{cases}$$

Clearly, L is symmetric and positive definite. Hence, it induces the following K-DPP $\mathcal{P}^K$:

$$\begin{aligned} \mathcal{P}^K(\mathbf{Y} = I) &\propto \det(L^b_I) && (\mathbf{Y} \text{ is the random variable specifying the K-DPP's outcome, } I \in \mathcal{M}) \\ &= \prod_{i \in I} L^b_{i,i} && (L^b_I \text{ is a diagonal matrix}) \\ &= e^{-\eta \sum_{i \in I} \widetilde{L^b_i}} \\ &= e^{-\eta \widetilde{\mathbf{L^b_I}}} && (\text{by definition of } \widetilde{\mathbf{L^b_I}}) \end{aligned}$$

Note that a K-DPP samples *subsets* of size K, i.e. $\mathbf{Y}$ does not contain any element twice and its size is K. Since the probabilities need to sum up to one, we conclude:

$$\begin{aligned} \mathcal{P}(\mathbf{Y} = I) &= \frac{e^{-\eta \widetilde{\mathbf{L^b_I}}}}{\sum_{J \in \mathcal{M}} e^{-\eta \widetilde{\mathbf{L^b_J}}}} \\ &= Pr[\widetilde{J^b} = I] && (\text{Coordinator's probability of choosing meta-arm } I) \end{aligned}$$

Cost for sampling a meta-arm Kulesza and Taskar (2012) describe in their Algorithm 1 how to sample from a *general* DPP. In a general DPP, the outcome can be any subset of $\mathcal{Y}$, its size is not necessarily equal to K. Their algorithm consists of two phases:

1. Sample eigenvectors of L^b. This determines the size of the DPP outcome.

2. Use the sampled eigenvectors to actually choose a subset of $\mathcal{Y}$.

In order to obtain a K-DPP algorithm, they replace phase 1 with an algorithm that samples *exactly* K eigenvectors (see Alg. 6). This then fixes the size of the outcome to K, which is what we want in a K-DPP. Algorithm 5 describes the algorithm for sampling from a K-DPP.

Since our matrix L^b is diagonal, its eigendecomposition is very simple: The eigenvalues are the diagonal elements of L^b, the eigenvectors are the standard basis vectors. This means, that in phase 2 of Alg. 5, we would simply end up choosing the indexes of the eigenvectors computed in phase 1. Thus, we can actually finish after phase 1.

Algorithm 6 describes how to sample exactly K eigenvectors. Since this part requires $O(NK)$, we conclude that sampling a meta-arm in any block can be done in $O(NK)$. For completeness, we have written down the algorithm for sampling K eigenvectors from Kulesza and Taskar (2012) and a sub-algorithm that it uses here in Algorithm 6 and 7, respectively.

Algorithm 5 Sampling from a K-DPP (based on Algorithm 1 from Kulesza and Taskar (2012))

```
$(v_n, \lambda_n)_{n=1}^N$ = Eigendecomposition of $L^b$
*Phase 1 begins*
$V \leftarrow$ Sample $K$ eigenvectors of $L^b$ (Algorithm 6)
*Phase 2 begins*
$J \leftarrow \emptyset$
while $|V| > 0$ do
    Select $i$ from $[N]$ with $Pr(i) = \frac{1}{|V|} \sum_{v \in V} (v^T e_i)^2$    ▷ $e_i$ = i-th standard basis vector
    $J \leftarrow J \cup \{i\}$
    $V \leftarrow V_\perp$, an orthonormal basis for the subspace of $V$ orthogonal to $e_i$
end while
Output: $J$
```

Algorithm 6 Sampling K eigenvectors (Algorithm 8 in Kulesza and Taskar (2012))

```
Input: $K$, $(v_n, \lambda_n)_{n=1}^N$ = Eigendecomposition of $L^b$
Compute $e_l^n$ for $l = 0, 1, ..., K$ and $n = 0, 1, ..., N$ (Algorithm 7)
$J \leftarrow \emptyset$
$l \leftarrow K$
for $n = N, ..., 2, 1$ do
    if $l = 0$ then
        break
    end if
    if $u \sim U[0,1] < \lambda_n \frac{e_{l-1}^{n-1}}{e_l^n}$ then
        $J \leftarrow J \cup \{v_n\}$
        $l \leftarrow l - 1$
    end if
end for
Output: $J$
```

Cost for computing the marginal probability of one arm If the K-Metaplayer decides to update arm i at the end of block b, she needs to compute the marginal probability $\sum_{i \in I \in \mathcal{M}} p^b(I)$. We can rewrite this as follows:

$$\begin{aligned}
\sum_{i \in I \in \mathcal{M}} p^b(I) &= \sum_{i \in I \in \mathcal{M}} \frac{e^{-\eta \widetilde{\mathbf{L}_\mathbf{I}^\mathbf{b}}}}{Z_K^N} && \text{(Normalizer } Z_K^N := \textstyle\sum_{J \in \mathcal{M}} e^{-\eta \widetilde{\mathbf{L}_\mathbf{J}^\mathbf{b}}}\text{)} \\
&= \frac{1}{Z_K^N} \sum_{i \in I \in \mathcal{M}} e^{-\eta \sum_{j \in I} \widetilde{L_j^b}} && \text{(by definition of } \widetilde{\mathbf{L}_\mathbf{I}^\mathbf{b}}\text{)} \\
&= \frac{e^{-\eta \widetilde{L_i^b}}}{Z_K^N} \cdot \underbrace{\sum_{i \notin \{i_1, \ldots, i_{K-1}\} \subseteq \mathcal{Y}} e^{-\eta \sum_{k=1}^{K-1} \widetilde{L_{i_k}^b}}}_{=:(*)}
\end{aligned}$$

Algorithm 7 Sub-algorithm for Alg. 6: Computing elementary symmetric polynomials (Algorithm 7 in Kulesza and Taskar (2012))

```
Input: $K$, eigenvalues $\lambda_1, \lambda_2, ..., \lambda_N$
$e_0^n \leftarrow 1 \; \forall n \in \{0, 1, 2, ..., N\}$
$e_l^0 \leftarrow 0 \; \forall l \in \{1, 2, ..., K\}$
for $l = 1, 2, ..., K$ do
    for $n = 1, 2, ..., N$ do
        $e_l^n \leftarrow e_l^{n-1} + \lambda_n e_{l-1}^{n-1}$
    end for
end for
Output: Values $e_i^j, \forall i \in \{0, ..., K\}, \forall j \in \{0, ..., N\}$
```

By inspecting the expression inside sum $(*)$ more closely, we observe that it looks very similar to the K-DPP that we defined before. In fact, that expression can be seen as a (K-1)-DPP over ground set $[N] \setminus \{i\}$ with Kernel matrix L_{-i}^b consisting of L^b without the i-th row and column. Therefore, the sum $(*)$ is actually just the normalization constant, let's call it Z_{K-1}^{N-i}, of that (K-1)-DPP. Hence, the marginal probability for arm i can be written as:

$$\sum_{i \in I \in \mathcal{M}} p^b(I) = \frac{e^{-\eta \widetilde{L_i^b}}}{Z_K^N} \cdot Z_{K-1}^{N-i}$$

From proposition 5.1 in Kulesza and Taskar (2012), we know that both Z_K^N and Z_{K-1}^{N-i} can be computed in $O(NK)$ each. We conclude that calculating the marginal probability for one arm in any block can be done in $O(NK)$.

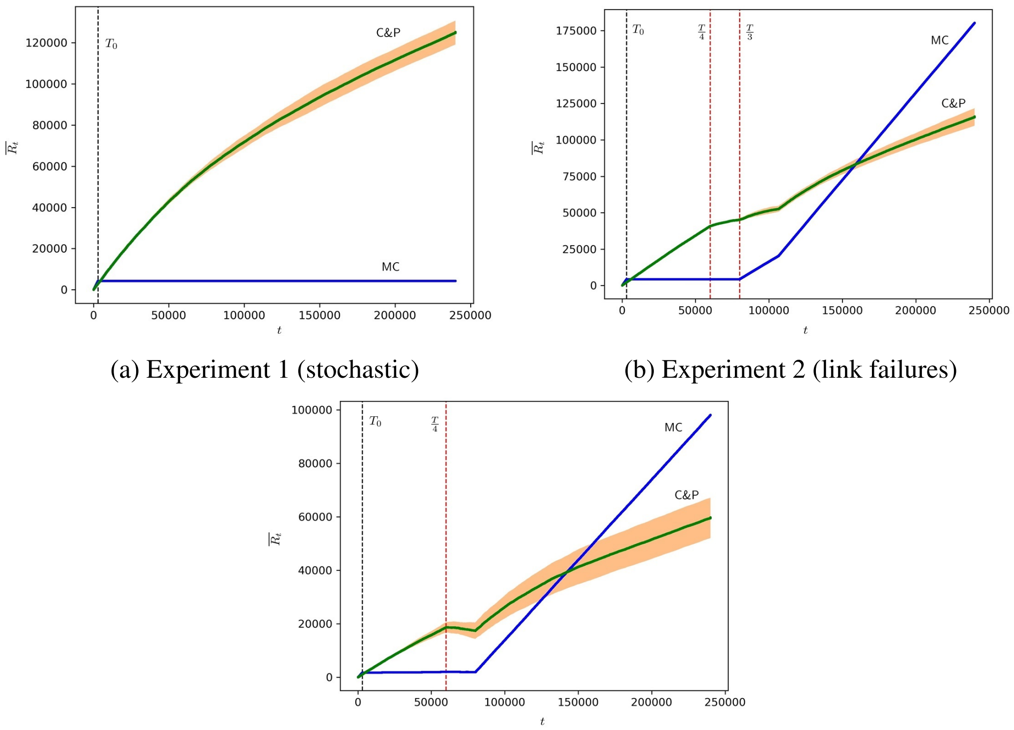
5. Experiments

We run experiments with three different setups and compare the performance of C&P to the Musical Chairs algorithm (MC) from Rosenski et al. (2016).

MC achieves constant regret with high probability in a stochastic setting by assuming a fixed gap between the K-th and $(K+1)$-th best arms. It starts with a *learning* phase of $T_0 \in O(1)$ rounds, during which players explore arms uniformly at random and observe losses. Based on these observations, the players compute a mean loss estimate for each arm. In the second phase, the players play a *musical chairs* game, where each player keeps choosing u.a.r. among the K best arms according to her own estimates. As soon as a player chooses an arm without colliding for the first time, she becomes the owner of that arm and stays there for the rest of the game.

For all experiments, we set $N = 8$, $K = 4$, $T = 240000$, $T_R = 20$ and $T_0 = 3000$. This value for T_0 was also used for the experiments by Rosenski et al. (2016). We repeat this for 10 runs for each setup and measure the online regret R_t, i.e., the difference between the cumulative player loss at time $t \in [T]$ and the cumulative loss of the K arms that are the K best in the time period $[t]$.

For each setup, we create a plot that shows the average regret and the standard deviation (as a colored region around the average). In the plots, the blue curves show the results of MC and the green curves show the results of C&P. The black dashed line indicates the end of MC's learning phase ($t = T_0$).



(a) Experiment 1 (stochastic)

(b) Experiment 2 (link failures)

(c) Experiment 3 (link improves)

Figure 2: Results: The red dashed lines indicate when a link failed or came up.

For all of the following three setups, we also run experiments to measure the accumulated regret R_T after T rounds. These can be found in Section A.4 of Appendix A.

Experiment 1 (Stochastic) We choose N mean losses in [0,1] u.a.r. with a gap of at least 0.05 between the Kth and $(K+1)$-th best arms. For each arm, the losses are sampled i.i.d. from a Bernoulli distribution with the selected means. This is a similar setup as the one from Rosenski et al. (2016). The results are shown in Figure 2a. As we can see, MC (blue curve) accumulates regret up to time T_0 but the regret stays constant afterwards. For our algorithm (green curve), we can see that it keeps accumulating regret until the end of the game.

Experiment 2 (Non-stochastic) We model a network scenario in which good links (i.e. small losses) fail suddenly. Concretely, we initially set the mean loss μ_i for each arm i as follows: $\mu_1 = \mu_2 = \mu_3 = \mu_4 = 0.1$ and $\mu_5 = \mu_6 = \mu_7 = \mu_8 = 0.3$. Each arm i's losses are sampled i.i.d. from Bernoulli distribution $Ber(\mu_i)$. At time $\frac{T}{4}$, “link" (arm) 1 fails and its remaining losses are sampled i.i.d. from $Ber(0.9)$. After a while, at time $\frac{T}{3}$, link 3 also fails and from then on its losses are also chosen from $Ber(0.9)$. Figure 2b shows the results of this experiment. The red dashed lines represent the two link failures.

Experiment 3 (Non-stochastic) We model another network scenario, in which a bad link improves suddenly (or a link that was down comes up). We set the initial mean losses as follows: $\mu_1 = 0.9$ and $\mu_2 = \mu_3 = \mu_4 = \mu_5 = \mu_6 = \mu_7 = \mu_8 = 0.7$. As before, the losses are sampled i.i.d. from a Bernoulli distribution with the corresponding means. At time $\frac{T}{4}$, link 1 improves and its losses are from then on chosen from $Ber(0.1)$. Figure 2c shows the results of this experiment. The red dashed line shows when link 1 improves.

In Experiments 2 and 3, the link failures and improvements, respectively, happen *after* the learning phase T_0 in MC. Due to this, MC (blue curve) cannot react to them and its regret starts to increase. For C&P (green curve), the regret is initially larger than MC, yet it is able to react to the link changes.

6. Discussion and Conclusions

We have presented the first efficient algorithm for the multiplayer “no communication" adversarial setting. Our method obtains a regret bound of $\widetilde{O}(K^{4/3}N^{1/3}T^{2/3})$, and it is interesting to understand whether one can devise an efficient method that obtains a rate of $O(\text{poly(K,N)}\sqrt{T})$.

In our algorithm, there is a single learner (coordinator) while all others just accept the coordinator’s decisions and ignore the loss feedback that they receive. This poses a single point of failure. One possible way to remedy this might be to switch coordinators after each block in a round-robin fashion: Player 1 would be the coordinator in block 1, player 2 would be the coordinator in block 2 and so on.

Acknowledgments

We would like to thank Johannes Kirschner and Mojmír Mutný for their valuable feedback on the manuscript. We would also like to thank the anonymous reviewers for their reviews and suggestions. This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme grant agreement No. 815943, as well as from the ETH Zurich Postdoctoral Fellowship and Marie Curie Actions for People COFUND program.

Appendix A.

A.1. Regret analysis for Lemma 1 (K-Metaplayer)

In this section, we will prove that using $\eta = \sqrt{\frac{\log\binom{N}{K}}{KTN}}$, the metaplayer's regret with Alg. 1 is bounded by,

$$\mathbb{E}[R_T^{\text{meta}}] \leq 2\sqrt{KTN \log\binom{N}{K}} \leq 2K\sqrt{TN\log N}\ .$$

Proof Using the view of meta-arms (see Section 3.1), Alg. 1 is very similar to playing EXP3 on $|\mathcal{M}| = \binom{N}{K}$ meta-arms. In order to apply regret guarantees from the EXP3 analysis, we need to show that:

1. A meta-arm $I \in \mathcal{M}$ is chosen proportional to $\exp(-\eta\widetilde{\mathbf{L}^{\mathbf{t}}_{\mathbf{I}}})$ at time t, where $\widetilde{\mathbf{L}^{\mathbf{t}}_{\mathbf{I}}} = \sum_{\tau=1}^{t-1}\sum_{i\in I}\widetilde{l_i^\tau}$ is the cumulative loss estimate of I at time t. This can be seen directly in Alg. 1.

2. $\widetilde{\mathbf{l}^{\mathbf{t}}_{\mathbf{I}}} = \sum_{i\in I}\widetilde{l_i^t}$ is an unbiased estimate of the true meta-arm's loss $\mathbf{l}^{\mathbf{t}}_{\mathbf{I}} = \sum_{i\in I} l_i^t$ at time t, for any $I \in \mathcal{M}$ and any t. For this, we will first show that for any arm $i \in [N]$, $\widetilde{l_i^t}$ is an unbiased estimate of l_i^t:

$$\begin{aligned}
\mathbb{E}[\widetilde{l_i^t}|p^t] &= K \cdot \frac{l_i^t}{\sum_{i\in Z\in\mathcal{M}} p^t(Z)} \cdot Pr[J^t = i] \\
&= K \cdot \frac{l_i^t}{\sum_{i\in Z\in\mathcal{M}} p^t(Z)} \cdot \underbrace{Pr[i\in I^t]}_{=\sum_{i\in Z\in\mathcal{M}} p^t(Z)} \underbrace{\cdot\, Pr[J^t = i | i \in I^t]}_{=\frac{1}{K}} \\
&= l_i^t
\end{aligned}$$

From the law of total expectation, we can derive that $\mathbb{E}[\widetilde{l_i^t}] = \mathbb{E}[\mathbb{E}[\widetilde{l_i^t}|p^t]] = l_i^t$. Finally, by linearity of expectation (as $\widetilde{\mathbf{l}^{\mathbf{t}}_{\mathbf{I}}} = \sum_{i\in I}\widetilde{l_i^t}$, we conclude that $\widetilde{\mathbf{l}^{\mathbf{t}}_{\mathbf{I}}}$ is an unbiased estimate of $\mathbf{l}^{\mathbf{t}}_{\mathbf{I}}$.

Given 1. and 2., we can apply standard EXP3 regret guarantees to obtain the following bound on the metaplayer's regret:

$$\mathbb{E}[R_T^{\text{meta}}] \leq \eta \sum_{t=1}^{T}\sum_{I\in\mathcal{M}} \mathbb{E}[p^t(I) \cdot \underbrace{\mathbb{E}[(\widetilde{\mathbf{l}^{\mathbf{t}}_{\mathbf{I}}})^2 | p^t]}_{=:(*)}] + \frac{\log\binom{N}{K}}{\eta}$$

(e.g. see Lecture 9, McMahan and Dekel (2014). Also, we used that $|\mathcal{M}| = \binom{N}{K}$.)

The variance term $(*)$ can be simplified as follows:

$$\begin{aligned}
\mathbb{E}[(\widetilde{\mathbf{l}^{\mathbf{t}}_{\mathbf{I}}})^2|p^t] &= \mathbb{E}[(\sum_{i\in I}\widetilde{l^t_i})^2|p^t] && \text{(by definition)}\\
&= \sum_{j,k\in I}\mathbb{E}[\widetilde{l^t_j}\cdot\widetilde{l^t_k}|p^t] && \text{(Linearity of expectation)}\\
&= \sum_{i\in I}\mathbb{E}[(\widetilde{l^t_i})^2|p^t]
\end{aligned}$$

(The loss estimate at time t is non-zero for at most one arm, thus all terms for $j \neq k$ cancel)

$$\begin{aligned}
&= \sum_{i\in I}\left(\frac{K\cdot l^t_i}{\sum_{i\in Z\in\mathcal{M}}p^t(Z)}\right)^2\cdot Pr[J^t=i]\\
&= \sum_{i\in I}\left(\frac{K\cdot l^t_i}{\sum_{i\in Z\in\mathcal{M}}p^t(Z)}\right)^2\cdot \underbrace{Pr[i\in I^t]}_{=\sum_{i\in Z\in\mathcal{M}}p^t(Z)}\cdot\underbrace{Pr[J^t=i|i\in I^t]}_{=\frac{1}{K}}\\
&= K\sum_{i\in I}\frac{(l^t_i)^2}{\sum_{i\in Z\in\mathcal{M}}p^t(Z)}
\end{aligned}$$

Plugging this back into our expression for the regret, we obtain:

$$\begin{aligned}
\mathbb{E}[R^{\text{meta}}_T] &\leq \eta\sum_{t=1}^{T}\sum_{I\in\mathcal{M}}\mathbb{E}[p^t(I)\cdot\mathbb{E}[(\widetilde{\mathbf{l}^{\mathbf{t}}_{\mathbf{I}}})^2|p^t]] + \frac{\log\binom{N}{K}}{\eta}\\
&= \eta\sum_{t=1}^{T}\sum_{I\in\mathcal{M}}\mathbb{E}[p^t(I)\cdot K\sum_{i\in I}\frac{(l^t_i)^2}{\sum_{i\in Z\in\mathcal{M}}p^t(Z)}] + \frac{\log\binom{N}{K}}{\eta}\\
&= K\eta\sum_{t=1}^{T}\mathbb{E}[\underbrace{\sum_{I\in\mathcal{M}}p^t(I)\sum_{i\in I}\frac{(l^t_i)^2}{\sum_{i\in Z\in\mathcal{M}}p^t(Z)}}_{(\star)}] + \frac{\log\binom{N}{K}}{\eta} && \text{(Linearity of expectation)}
\end{aligned}$$

In $(\star)$, we first sum over all meta-arms I and then over all arms i that are in I. We can instead sum over all arms i first and then over all meta-arms I that contain i. Hence, we can rewrite $(\star)$ as follows:

$$\begin{aligned}
\sum_{I\in\mathcal{M}}p^t(I)\sum_{i\in I}\frac{(l^t_i)^2}{\sum_{i\in Z\in\mathcal{M}}p^t(Z)} &= \sum_{i=1}^{N}\frac{(l^t_i)^2}{\sum_{i\in Z\in\mathcal{M}}p^t(Z)}\sum_{i\in I\in\mathcal{M}}p^t(I)\\
&= \sum_{i=1}^{N}(l^t_i)^2
\end{aligned}$$

By plugging this back into our regret expression, we get:

$$\begin{aligned}
\mathbb{E}[R_T^{\text{meta}}] &\leq K\eta \sum_{t=1}^{T} \mathbb{E}[\sum_{I\in\mathcal{M}} p^t(I) \sum_{i\in I} \frac{(l_i^t)^2}{\sum_{i\in Z\in\mathcal{M}} p^t(Z)}] + \frac{\log\binom{N}{K}}{\eta} \\
&= K\eta \sum_{t=1}^{T} \mathbb{E}[\sum_{i=1}^{N} (l_i^t)^2] + \frac{\log\binom{N}{K}}{\eta} \\
&= K\eta \sum_{t=1}^{T} \sum_{i=1}^{N} \mathbb{E}(\underbrace{l_i^t}_{\in[0,1]})^2 + \frac{\log\binom{N}{K}}{\eta} \\
&\leq KTN\eta + \frac{\log\binom{N}{K}}{\eta} \\
&= 2\sqrt{KTN\log\binom{N}{K}} && (\text{for } \eta = \sqrt{\tfrac{\log\binom{N}{K}}{KTN}}) \\
&\leq 2K\sqrt{TN\log N}
\end{aligned}$$

This concludes the regret analysis for Lemma 1. ■

A.2. Success analysis of the ranking algorithm 2

In this section, we will show that the players will successfully compute a ranking using algorithm 2 within $T_R = K \cdot e \cdot \log T$ rounds with probability at least $1 - \frac{K}{T}$. The analysis uses ideas from the proof of Lemma 3 in Rosenski et al. (2016).

For a fixed player, let q^t be the probability that she gets a rank assigned in step t. q^t can be bounded as:

$$q^t = \sum_{i\in\text{Free}} \frac{1}{K} \cdot (1 - \frac{1}{K})^{\text{Unranked}-1}$$

(Free = set of available arms at time t, Unranked = number of players who don't have a rank yet)

$$\begin{aligned}
&\geq \sum_{i\in\text{Free}} \frac{1}{K} \cdot (1 - \frac{1}{K})^{K-1} && (\text{Unranked is at most } K) \\
&\geq \frac{1}{K\cdot e} && (|\text{Free}| \geq 1,\ (1-\tfrac{1}{K})^{K-1} \geq e^{-1} \text{ for } K \geq 1)
\end{aligned}$$

The probability that she doesn't have a rank after step t is thus at most:

$$\begin{aligned}
&(1 - \frac{1}{K\cdot e})^t \\
&\leq e^{-\frac{t}{K\cdot e}} && (\text{Using the inequality } 1 - x \leq e^{-x})
\end{aligned}$$

By union bound, the probability that there's at least one player who is not fixed after $t = T_R$ rounds, is at most

$$K \cdot e^{-\frac{T_R}{K\cdot e}}$$

By setting $T_R = K \cdot e \cdot \log T$, we conclude that after T_R rounds, the probability that all players have a rank, is at least

$$1 - K \cdot e^{-K \cdot e \cdot \frac{\log T}{K \cdot e}}$$
$$= 1 - \frac{K}{T}$$

A.3. Staying Quiet

So far, we assumed that players need to stay quiet during the Coordinate phase of our C&P algorithm presented in Section 3. I.e., during sub-block k, all players except the coordinator and player k, don't pick any arms. This assumption can however be relaxed using a simple modification to our protocol:

During sub-block $k \in \{2, ..., K\}$, all followers except player k *stay* on arm 1. Player k explores all arms in a round-robin fashion for at most N steps, until she collides on an arm $i \neq 1$. If she manages to do so, i is the arm that the coordinator has chosen for her. If player k doesn't collide on any other arm except on 1, she can conclude that the coordinator has picked arm 1 for her.

A.4. Experiments (Measuring the accumulated regret)

For the three setups that we described in Section 5, we run experiments to measure the accumulated regret R_T of both MC and our algorithm. We visualize the outcome in a loglog plot to compare the experimental results with our theoretical bound (Theorem 2).

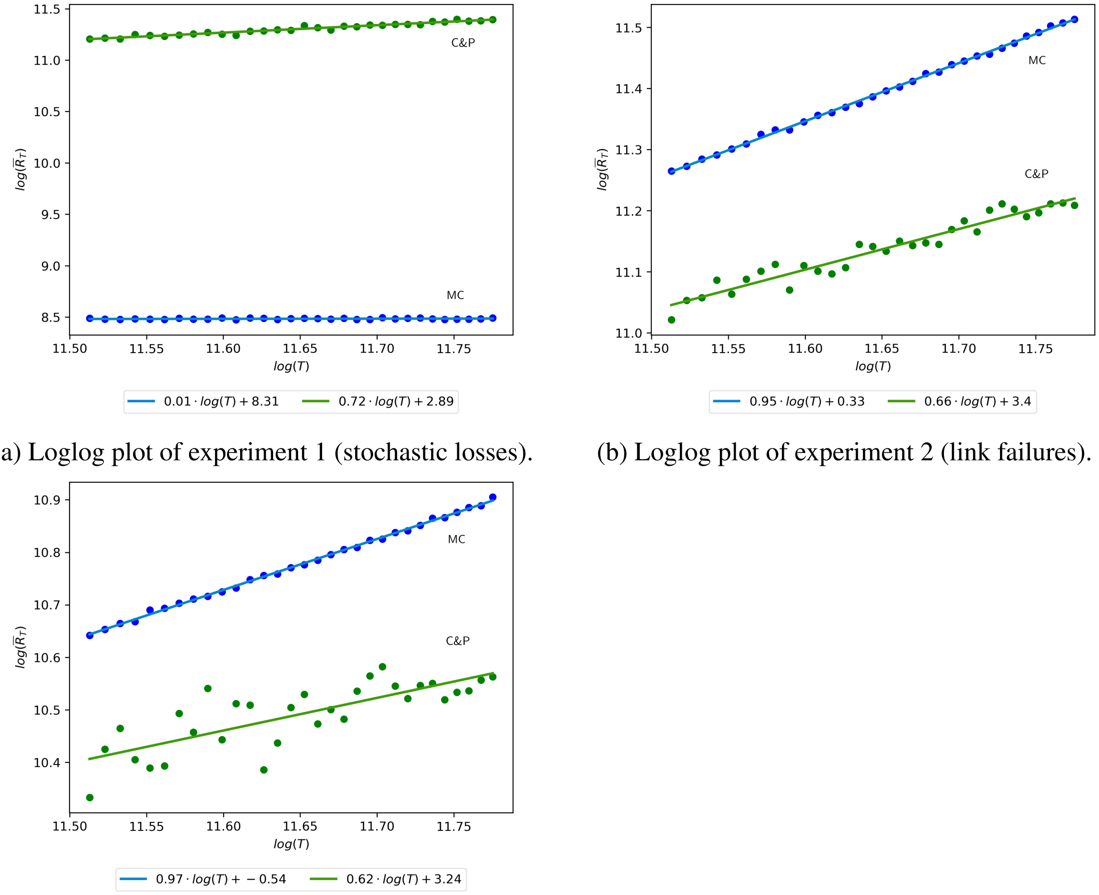
In all three experiments, we set $N = 8$, $K = 4$, $T_R = 25$ and $T_0 = 3000$ (length of MC's learning phase). For T, we choose $T = 100000 + i \cdot 1000$, where $i \in \{0, ..., 1300\}$. Per value of T, we do 10 runs and measure the regrets.

In the loglog plots, the blue dots show the average regrets of MC and the green dots the average regrets of our algorithm. The standard deviations are shown as colored regions around the average regrets. Besides this, we fit a line on the log average regrets for each algorithm and plotted those as well. With these lines, we can compare whether the experimental results match what we expect from Theorem 2.

Experiment 1 We use the same setup as in experiment 1 from 5, i.e. arms with i.i.d. Bernoulli losses where the arms' means are sampled u.a.r. from [0,1] with a gap of at least 0.05 between the K-th and $(K + 1)$-th best arm. The results are shown in Figure 3a.

Experiment 2 In this experiment, we use the setup from experiment 2 in Section 5, i.e. we model a network in which two links go down at time $\frac{T}{4}$ and $\frac{T}{3}$, respectively. Figure 3b shows the results of this experiment.

Experiment 3 For this, we use the setup from experiment 3 in Section 5, in which a bad link suddenly improves or comes up at time $\frac{T}{4}$. The outcome of this experiment is shown in Figure 3c.



(a) Loglog plot of experiment 1 (stochastic losses).

(b) Loglog plot of experiment 2 (link failures).

(c) Loglog plot of experiment 3 (link improves).

Figure 3

References

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

Jean-Yves Audibert, Sébastien Bubeck, and Gábor Lugosi. Regret in online combinatorial optimization. *Mathematics of Operations Research*, 39(1):31–45, 2013.

Peter Auer, Nicolò Cesa-bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32:2002, 2002.

Orly Avner and Shie Mannor. Concurrent bandits and cognitive radio networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 66–81. Springer, 2014.

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

Orly Avner and Shie Mannor. Multi-user communication networks: A coordinated multi-armed bandit approach. *arXiv preprint arXiv:1808.04875*, 2018.

Baruch Awerbuch and Robert Kleinberg. Competitive collaborative learning. *Journal of Computer and System Sciences*, 74(8):1271–1288, 2008.

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

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

Sébastien Bubeck, Yuanzhi Li, Yuval Peres, and Mark Sellke. Non-stochastic multi-player multi-armed bandits: Optimal rate with collision information, sublinear without. *arXiv preprint arXiv:1904.12233*, 2019.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Nicolo Cesa-Bianchi, Claudio Gentile, Yishay Mansour, and Alberto Minora. Delay and cooperation in nonstochastic bandits. *JOURNAL OF MACHINE LEARNING RESEARCH*, 49:605–622, 2016.

Richard Combes, Mohammad Sadegh Talebi Mazraeh Shahi, Alexandre Proutiere, et al. Combinatorial bandits revisited. In *Advances in Neural Information Processing Systems*, pages 2116–2124, 2015.

Ofer Dekel, Ambuj Tewari, and Raman Arora. Online bandit learning against an adaptive adversary: from regret to policy regret. In *ICML*, 2012.

Alex Kulesza and Ben Taskar. *Determinantal Point Processes for Machine Learning*. Now Publishers Inc., Hanover, MA, USA, 2012. ISBN 1601986289, 9781601986283.

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

Tor Lattimore and Csaba Szepesvári. Bandit algorithms. *preprint*, 2018.

Haoyang Liu, Keqin Liu, Qing Zhao, et al. Learning in a changing world: Restless multi-armed bandit with unknown dynamics. *IEEE Trans. Information Theory*, 59(3):1902–1916, 2013.

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

Brendan McMahan and Ofer Dekel. Cse599s: Online learning, 2014. URL `https://courses.cs.washington.edu/courses/cse599s/14sp/scribes.html`.

Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

Jonathan Rosenski, Ohad Shamir, and Liran Szlak. Multi-player bandits: A musical chairs approach. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, 2016.

Taishi Uchiya, Atsuyoshi Nakamura, and Mineichi Kudo. Algorithms for adversarial bandit problems with multiple plays. In *International Conference on Algorithmic Learning Theory*, pages 375–389. Springer, 2010.

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