Attacking Multi-Player Bandits and How to Robustify Them

Shivakumar Mahesh
University of Oxford
United Kingdom
shivakumar.mahesh@spc.ox.ac.uk

Anshuka Rangi
Amazon
USA
arangi@eng.ucsd.edu

Haifeng Xu
University of Chicago
USA
haifengxu@uchicago.edu

Long Tran-Thanh
University of Warwick
United Kingdom
long.tran-thanh@warwick.ac.uk

ABSTRACT

Motivated by cognitive radios, stochastic Multi-Player Multi-Armed Bandits has been extensively studied in recent years. In this setting, each player pulls an arm, and receives a reward corresponding to the arm if there is no collision, namely the arm was selected by one single player. Otherwise, the player receives no reward if collision occurs. In this paper, we consider the presence of malicious players (or attackers) who obstruct the cooperative players (or defenders) from maximizing their rewards, by deliberately colliding with them. We provide the first decentralized and robust algorithm RESYNC for defenders whose performance deteriorates gracefully as $\tilde{O}(C)$ as the number of collisions C from the attackers increases. We show that this algorithm is order-optimal by proving a lower bound which scales as $\Omega(C)$. This algorithm is agnostic to the algorithm used by the attackers and agnostic to the number of collisions C faced from attackers.

KEYWORDS

Multi-player bandits, Collision attack, Robustification

ACM Reference Format:
Shivakumar Mahesh, Anshuka Rangi, Haifeng Xu, and Long Tran-Thanh. 2024. Attacking Multi-Player Bandits and How to Robustify Them. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 9 pages.

1 INTRODUCTION

Multi-Player Multi-Armed Bandits (MP-MAB) algorithms have found applications in distributed computing, social recommendation systems, federated learning, sensor networks, Internet of Things, web services and crowdsourcing systems. Typically, these variants involve a large number of players playing a bandit instance, and may or may not be communicating with each other. The objective of the players, together as a group, is to maximize the collective reward. The distributed nature of these applications makes the learning algorithms prone to attacks from malicious players (or attackers).

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

In this paper, we focus an important and widely studied decentralized MP-MAB setting motivated by cognitive radios ([2, 12]). In this setting, an arm corresponds to a channel frequency, and the player selects a channel to transmit information, where the reward corresponding to this arm is given by the transmission quality of the channel. A key feature of this MP-MAB setting is that if multiple players choose the same arm (or channel) in a round, then *collision occurs* and all these players *receive zero reward for that round.* Additionally, the players *receive feedback whether or not a collision occurred* on the arm they pulled. Finally, in the decentralized case, players are independent and *cannot communicate with each other* through dedicated communication channels.

Decentralized MP-MAB setting has been widely studied in the absence of attackers ([1, 3, 14, 15, 20]). The optimal algorithms in the absence of attackers critically relies on the assumption that all the players are cooperative and execute the same algorithm. However, this assumption critically impairs the application of these algorithms. For example, in cognitive radio, a channel can be accessed by any player with a transmitter, and access is not restricted to cooperative players alone. Therefore, malicious players (or attackers) can obstruct cooperative players (or defenders) from maximizing reward by deliberately colliding with them. Further, the algorithms used by such attackers is unknown to the defenders. There has been limited focus on addressing this key issue and designing robust MP-MAB algorithms for the defenders that are agnostic to the algorithm used by attackers.

While the goal of the defenders is to minimize their collective regret, the goal of the attackers is to force the defenders to incur linear regret while minimizing the number of adversarial collisions induced by them. To this end, at each round, each attacker may pull an arm, observe the reward and the feedback if the collision occurred corresponding to the arm. If the attacker chooses to not to pull any arm, then no reward and collision information is observed, also termed as "staying quiet" by [1]. This work focuses on proposing algorithms for the defenders which are robust to these attackers. As a motivating example, one may consider the defenders as licensed spectrum users and the attackers as unlicensed ones. In this case, the licensed users wish to find the optimal allocation of arms, in the presence of interference from unlicensed users.

1.1 Related work

The concern of designing MP-MAB algorithms robust to malicious players (or attackers) has been raised in multiple works ([13, 20]) but has only been studied under the assumption that the attacker

Feedback Model	*Algorithm/ Reference*	*Prior Knowledge*	*Regret under our Attack model*
Non-Distinguishable Collision Sensing	MC, Proposition 1 [20]	T, Δ	$\mathbb{E}R_T = \Omega(T)$ under $C = O\left(\frac{K}{\Delta^2}\log(K^2T)\right)$
Non-Distinguishable Collision Sensing	SIC-MMAB, Proposition 2 [6]	T	$\mathbb{E}R_T = \Omega\left(\frac{K-N}{K}T\right)$ under $C = O\left(K^2\log T\right)$
Non-Distinguishable Collision Sensing	SIC-GT, Proposition 3 [7]	T	$\mathbb{E}R_T = \Omega(T)$ under $C = O\left(K^2\log T\right)$
Non-Distinguishable Collision Sensing	CNJ,CUJ, extended version of this paper [17] [21]	T, Δ	$\mathbb{E}R_T = \Omega(T)$ under $C = \tilde{O}(\log T)$
Distinguishable Collision Sensing	CDJ, extended version of this paper [17] [21]	T, Δ	$\mathbb{E}R_T = \Omega(T)$ under $C = \tilde{O}(\log T)$
Non-Distinguishable Collision Sensing	RESYNC, Theorem 1 This work	T, Δ, n, j	$\mathbb{E}R_T = O\left(CN^3 + KCN\frac{\log(K^2T)}{\Delta^2}\right)$
Distinguishable Collision Sensing	RESYNC2, Theorem 2 This work	T, Δ	$\mathbb{E}R_T = O\left(KCN + KN\frac{\log(K^2T)}{\Delta^2}\right)$
Non-Distinguishable + Distinguishable Collision Sensing	LOWER BOUND, Theorem 3 This work	N/A	$\mathbb{E}R_T = \Omega\left(\frac{N}{K}C\right)$

Table 1: Summary of Contributions: Regret Bounds of Decentralized MP-MAB Algorithms under C number of collisions from attackers. Here, N denote the number of defenders, and $\Delta = \mu_{(N)} - \mu_{(N+1)}$ is the gap between the expected rewards of N-th and $(N+1)$-th best arms. We use j to denote the prior knowledge where, each defender has access to a integer $j \in [N]$ distinct from every other defender (for more details refer Assumption 1).

behaviour is known in advance to the defenders. For instance, [21, 27] consider attackers that follow a specific algorithm of learning and pulling the optimal set of arms, and construct robust defence algorithms (CNJ, CUJ and CDJ) against these specific attackers. In this work, we show that these algorithms are not robust to an attacker with a different attack strategy. Namely, we show that the existing algorithms including these robust algorithms incur linear regret $\Omega(T)$ with only small $O(\log T)$ number of adversarial collisions from an attacker with a different strategy.

[7] consider non-cooperative players whose incentives are to maximize their own individual rewards. They provide the first algorithm SIC-GT, which is robust to selfish players i.e. following their algorithm is a ϵ-Nash equilibrium. For performance guarantees, they assume that selfish players will not deviate from this particular ϵ-Nash equilibrium where every player executes the same algorithm. This assumption may fail in practice since non-cooperative players may be unaware of the algorithm used by a specific group of cooperative players, and given that the Nash equilibrium is not unique, they may opt to use a single-player bandit algorithm to select arms instead. In this work, we also show that if even one player deviates from the common algorithm, the cooperative players are forced to suffer linear regret $\Omega(T)$.

Against this background, our work addresses the limitations of existing algorithms by proposing a robust algorithm whose performance is agnostic to the attack strategy.

1.2 Contributions

In this work, we consider two feedback models: distinguishable and non-distinguishable collision sensing. In distinguishable sensing ([21]), each defender receives feedback on whether a collision occurred and can also distinguish whether the collision occurred from an attacker or a defender. In non-distinguishable sensing ([5, 7, 24]), a defender receives feedback on whether a collision occurred and does not have the capability to distinguish whether a collision occurred from an attacker or a defender. Our main contributions are the following:

- First, we show that the representative existing algorithms in the literature, namely MC, SIC-MMAB, SIC-GT, CNJ, CUJ and CDJ, are not robust to adversarial collisions. More specifically, only $O(\log T)$ adversarial collisions from a single attacker are sufficient to ensure that the expected regret of these algorithms scales linearly as $\Omega(T)$, where T is the total number of rounds for which the players interacts in the MP-MAB setting. Table 1 summarizes the results showing "non-robustness" for these existing algorithms.
- In the non-distinguishable collision-sensing setting, we propose a novel algorithm RESYNC which exhibits robust behaviour to adversarial collisions. We show that the expected regret of this algorithm deteriorates gracefully as $\tilde{O}(C)$, where C is the total number of adversarial collisions from the attackers. We also shows that this scaling with C is order-optimal up to logarithmic factors in T by proving a corresponding lower bound in MP-MAB setting which scales as $\tilde{\Omega}(C)$.
- In the distinguishable collision-sensing setting, we propose another novel algorithm RESYNC2, and show that the expected regret of this algorithm also deteriorates linearly as $\tilde{O}(C)$. This scaling of the expected regret is order-optimal in C. Due to an additional feedback information available in

this setting, the regret bound of RESYNC2 is further improved by logarithmic factors in T in comparison to RESYNC.
- Finally, we present several experiments which validate our theoretical findings.

Due to space limitations, we defer the proofs and experiments to the extended version of this paper [17] .

2 PRELIMINARIES

We consider a multi-player variant of the standard stochastic MAB problem with K arms, denoted by set $[K] = \{1, \cdots, K\}$. For each arm $k \in [K]$ at time (or round) $t \leq T$, we denote its realized reward by $X_k(t) \in [0, 1]$ drawn i.i.d according to distribution ν_k with expectation μ_k. Additionally, we assume that the expected rewards of the arms are different, namely $\mu_{(1)} > \mu_{(2)} > \cdots > \mu_{(K)}$ where $\mu_{(i)}$ denotes the i-th largest expected reward. We denote the number of attackers by M and the number of defenders by N such that $K \geq N$ (as commonly assumed in [8, 18, 22, 24, 26]). We index the defenders using the set $\{1, \ldots, N\}$ and the attackers using the set $\{N + 1, \ldots N + M\}$.

In decentralized MP-MAB, at each round $t \leq T$, each player $j \in [N + M]$ pulls an arm $\pi^j(t) \in [K]$ and receives a reward

$$r^j(t) = X_{\pi^j(t)}(t)(1 - \eta_{\pi^j(t)}(t)), \tag{1}$$

where $\eta_k(t) := \mathbb{1}(|1 \leq j \leq N + M : \pi^j(t) = k\}| > 1)$ is the *collision indicator* (this value is 1 if more than one player pulls that arm, otherwise remains 0). At each round t, each player who pulled an arm observes their collision indicator $\eta_{\pi^j(t)}(t)$ and the corresponding reward $r^j(t)$. If a collision occurred, namely $\eta_{\pi^j(t)}(t) = 1$, then the defenders may or may not receive additional information to distinguish whether that collision occurred with defenders, attackers or both. Based on the availability of this additional information, we consider two feedback models: Non-distinguishable and Distinguishable collision sensing.

In *non-distinguishable collision sensing*, the feedback is limited to the corresponding reward $r^j(t)$ and the collision indicator $\eta_{\pi^j(t)}(t)$ at each round t for player j. No additional information is available to the defenders to distinguish between the players causing the collisions.

In *distinguishable collision sensing*, the defenders receive the information about the nature of the players who caused the collision. More specifically, at each round t, defender j observes the reward $r^j(t)$ and the collision indicators

$$\eta_k^D(t) := \mathbb{1}(|\{1 \leq j \leq N : \pi^j(t) = k\}| > 1) \tag{2}$$

$$\eta_k^A(t) := \mathbb{1}(|\{N < j \leq N + M : \pi^j(t) = k\}| \geq 1) \tag{3}$$

where $\eta_k^D(t)$ and $\eta_k^A(t)$ represent if the collision occurred due to a defender and an attacker, respectively. Note that $\eta_{\pi^j(t)}(t) = \eta^A_{\pi^j(t)}(t) \vee \eta^D_{\pi^j(t)}(t)$. These indicators enable the defenders to distinguish between collisions from an attacker and a defender. As an illustration of this feedback model from [21], in cognitive radio networks (CRNs) defenders would be able to distinguish between collisions between themselves and attackers through acknowledgments. At the end of each round, each defender receives an ACK/NACK feedback. If a collision happens due to other defenders but not the attackers, each defender receives a NACK signal. However, if the collision is due to the attackers alone, no NACK signal is received. Finally if collision occurs simultaneously, a corrupted NACK signal is received.

The performance of an algorithm is measured in terms of expected regret which is defined as the difference between the maximal expected reward and the algorithm cumulative reward over T steps, namely

$$R_T := T \sum_{k=1}^{N} \mu_{(k)} - \sum_{t=1}^{T} \sum_{j=1}^{N} \mu_{\pi^j(t)} \cdot (1 - \eta_{\pi^j(t)}(t)). \tag{4}$$

The maximal expected reward corresponds to the top N arms, also referred to as the *optimal* set of arms.

The goal of the attackers is to force the defenders to incur linear regret while keeping the number of collisions they induce to be as small as possible to remain stealthy. The motivation for stealthy attacks is best illustrated using an example. As noted by [27], in CRNs, licensed users are protected by law. Therefore users of the network are motivated to reduce their interference with licensed users, since there will be heavy penalties if the licensed users detect prolonged interference from them ([25]). Therefore from an attackers perspective, collisions (or interference in the example) against defenders (or licensed users in the example) are inherently expensive. Hence, we evaluate the defenders' regret in terms of the number of times the attacker collides with the defender. Thus, the attack cost C is the number of adversarial collisions encountered by the defenders, namely

$$C = \sum_{t=1}^{T} \mathbb{1}\Big(\exists d, \exists a : 1 \leq d \leq N, N + 1 \leq a \leq N + M, \pi^d(t) = \pi^a(t)\Big) \tag{5}$$

We exclusively consider the case where the attack cost C is unknown to the defenders. In other words, the defenders are *agnostic* to C.

3 LIMITS OF EXISTING ALGORITHMS

This section shows the limitations of existing algorithms against a single attacker who has no prior information about the defenders and expected rewards of arms.

Attack on MC: In [20], the "Musical Chairs" subroutine in the MC algorithm is used to allocate players to optimal arms. This subroutine has inspired many algorithms to follow the same (or slightly modified) procedure ([4, 14, 24, 26]). By successfully attacking this subroutine in MC algorithm, we show that there is a critical threat to any follow-up algorithms that use this subroutine. The following proposition proves that the MC algorithm is not robust to a single attacker.

PROPOSITION 1. *Assuming that the defenders use MC, there exists an attack strategy with expected attack cost of*

$$O\left(\max\left(K \log(K^2 T)/\Delta^2_{min}, K^2 \log T\right)\right)$$

which ensures that the expected regret of the defenders is $\Omega(T)$*, where* $\Delta_{min} = \min_i \mu_{(i)} - \mu_{(i+1)}$.

Attack on SIC-MMAB: The seminal work of [6] introducing the SIC-MMAB algorithm with implicit communication through forced collisions has inspired many algorithms to follow suit ([9, 11, 16, 18,

23, 26]). We present an attack on SIC-MMAB algorithm which can be slightly modified to show the non-robustness of these follow up works. The following proposition proves that the SIC-MMAB algorithm is not robust to a single attacker.

PROPOSITION 2. *Assuming that the defenders use SIC-MMAB, there exists an attack strategy with expected attack cost* $O\left(K^2 \log T\right)$ *which ensures that the expected regret of the defenders is* $\Omega(T)$.

Attack on SIC-GT: The SIC-GT algorithm by [7] is robust to selfish players, namely the attacker is a reward maximizing player who gains reward from pulling arms. Playing SIC-GT is an ϵ-Nash equilibrium, and this equilibrium is achieved through "Grim Trigger" ([10]) punitive strategies. However, the following proposition proves that the SIC-GT algorithm is not robust to a single attacker.

PROPOSITION 3. *Assuming that the defenders use SIC-GT, there exists an attack strategy with expected attack cost* $O(K^2 \log T)$ *which ensures that the expected regret of the defenders is* $\Omega(T)$.

The attack strategies are presented in the extended version of this paper [17].

4 THE RESYNC ALGORITHM

In this section we consider the non-distinguishable collision sensing model, where defenders cannot determine whether a collision is from an attacker or a defender. We propose the algorithm RESYNC which is robust to adversarial collisions. We make the following assumption on the prior knowledge of defenders:

ASSUMPTION 1. *All defenders know the number of defenders* N. *Additionally, each defender has a unique identification number* $j \in [N]$, *also referred as internal rank, and has knowledge of their own internal rank.*

The *internal rank* allows the defenders to coordinate in the bandit game. In Section 5 where we consider the distinguishable collision case, we remove this assumption, and use the additional feedback to devise a sub-routine to estimate N and j.

4.1 Description of RESYNC

RESYNC consists of an exploration phase and an exploitation phase, where a defender may alternate between phases based on the collision feedback. The algorithm divides the time-horizon T into successive epochs of size $T_B = T_0 + 2N^2 + N$ rounds where $T_0 = 8K\lceil \log(2K^2T)/\Delta^2 \rceil$. Both exploration and exploitation phases run for T_B rounds. At any epoch some defenders may be in the exploration phase while other defenders may be in the exploitation phase. In both of these phases, each defender will choose whether or not to "restart", and enter the exploration phase in the next epoch. Whether or not a defender chooses to restart depends on the collisions sensed by the defender in each phase. We say that the defenders are *synchronized* over an epoch if all of them are in the same phase (exploration/exploitation) in that epoch. Otherwise we say that the defenders are *desynchronized* over that epoch.

RESYNC relies on three key ideas to solve the limitations of the existing approaches in the literature. First, the algorithm obtain sufficient number of uncorrupted observations to determine the optimal set of arms with high probability in a decentralized manner.

Algorithm 1: RESYNC

Input: T (horizon), N (number of defenders), j (internal rank), T_B (length of an epoch)

Initialize: Restart $\leftarrow$ True, Opt $\leftarrow \emptyset$, $\forall i : \tilde{\mu}_i \leftarrow 0, o_i \leftarrow 0, s_i \leftarrow 0$

for $\lfloor \frac{T}{T_B} \rfloor$ epochs **do**

  if Restart **then**

    (Restart, Opt) $\leftarrow$ Exploration(Restart, j, $\forall i\ o_i, s_i$) ;

    `// Restart ← True ⟺ collision occurs during sequential hopping or verdict of (intra/inter)-communication phase is to Restart`

  else

    Restart $\leftarrow$ Exploitation(Restart, j, Opt) ; `// Restart ← True ⟺ collision occurs during inter-communication phase`

Subroutine 2: Exploitation

Input: Restart, j, Opt; **Output** Restart

Initialize Restart $\leftarrow$ False, and $T_0 \leftarrow 8K\lceil \log(2K^2T)/\Delta^2 \rceil$

for $T_0 + 2N^2$ *times steps* **do**

  Set $k = t + j \pmod N$; Pull Opt$[k]$

for N *rounds* **do** `// Inter-communication phase`

  Set $k = t + j \pmod N$

  Pull Opt$[k]$ and receive $\eta_{\text{Opt}[k]}$

  if $\eta_{Opt[k]} = 1$ **then**

    Restart $\leftarrow$ True

return (Restart)

Second, the algorithm maintains synchronization over the defenders against attackers that challenge to desynchronize the system. Third, the algorithm ensures that the defenders pull the optimal set of arms in an orthogonal fashion (with no collisions amongst themselves), for the majority of the time-horizon.

The outline of RESYNC (**R**estart **Syn**chronously under Adversarial **C**ollisions) is provided in Algorithm 1, with the exploration and exploitation protocols provided in Subroutines 3 and 2 respectively. We use $t \in [T]$ to denote the round of the bandit game, and assume every defender knows about t at all times.

Exploration. (Subroutine 3) Within the exploration phase, there are four sub-phases, which are as follows: sequential hopping that runs for T_0 rounds, sensing that runs for N^2 rounds, intra-communication that runs for N^2 rounds and inter-communication that runs for N rounds.

Sequential Hopping. (Subroutine 3 lines 4-16) This sub-phase lasts for $T_0 = 8K\lceil \log(2K^2T)/\Delta^2 \rceil$ rounds. At each round, each defender pulls an arm, namely $j+t \pmod K) \in [K]$, based on its internal rank j and round of the bandit game $t \in [T]$. If all the defenders are in exploration phase, then this strategy ensures that there is no collision between the defenders. At each round t, the defender will also observe the reward and collision indicator. For the rounds when the the collision indicator is 0 indicating no collision, the defender maintains the cumulative sum of observed rewards and number of observations for each arm $k \in [K]$. If there is a collision, then the *Restart* variable is set to True and the observation received

Subroutine 3: Exploration

Input Restart, j, and for all $i \in [K]$, o_i, s_i;
Output: Restart, Opt
Initialize: Restart $\leftarrow$ False, SufficientObservations $\leftarrow$ False, and $T_0 \leftarrow 8K\lceil \log(2K^2T)/\Delta^2 \rceil$
for T_0 *rounds* **do** // Sequential hopping phase
  Pull $k = t + j \pmod K$ and receive η_k and $r_k(t)$
  if $\eta_k = 0$ **then**
    $o_k \leftarrow o_k + 1$
    $s_k \leftarrow s_k + r_k(t)$
  else
    Restart $\leftarrow$ True
if $\forall i : o_i \geq T_0/K$ **then**
  SufficientObservations $\leftarrow$ True
  For all $i \in [K]$, we have $\tilde{\mu}_i = s_i/o_i$
  Opt $\leftarrow$ List of N best empirically performing arms sorted according to arm index $i \in [K]$.
else
  Restart $\leftarrow$ True
for $(i, k) \in [N] \times [N]$ **do** // Sensing phase
  if $j = i$ **then**
    if SufficientObservations **then**
      Pull Opt[1] and receive $\eta_{Opt[1]}$ // attempt to sense the presence of defenders that are in exploitation phase
      if $\eta_{Opt[1]} = 1$ **then**
        Restart $\leftarrow$ True
    else
      Pull 1
  else
    if SufficientObservations **then**
      Pull Opt[1] + 1 $\pmod K$
    else
      Pull 1
for $(i, k) \in [N] \times [N]$ **do** // Intra-communication phase
  if $j = i$ **then** // send
    if Restart **then**
      Pull k
    else
      Pull j
  else // receive
    Pull j and receive η_j
    if $\eta_j = 1$ **then**
      Restart $\leftarrow$ True
for N *rounds* **do** // Inter-communication phase
  if SufficientObservations **then**
    Pull Opt[1] // to notify any defenders in exploitation phase to rejoin exploration in the next epoch
  else
    Pull 1
return (Restart, Opt)

is ignored. If there are sufficient reliable observations for all arms, then the defender sets the flag *SufficientObservations* to be True.

Sensing. (Subroutine 3 lines 17-29) This sub-phase lasts for N^2 rounds. In this sub-phase, a defender attempts to detect whether there is at least one other defender in the exploitation phase using the collision feedback she receives. This sub-phase is used to help defenders synchronize in the next epoch if they are desynchronized in the current epoch. In the sensing sub-phase, if a defender has the flag *SufficientObservations* to be True, then she pulls the optimal arm with least index (which we refer to as Opt[1]) in the jN-th, ..., $(jN + N - 1)$ - -th rounds of this sub-phase, and pulls arm Opt[1] + 1 (mod K) in the other rounds of the sub-phase. Note that this defender pulls Opt[1] for N consecutive rounds in this sub-phase, and sets *Restart* to True if a collision occurs in those N rounds.

Intra-Communication. (Subroutine 3 lines 30-39) In this sub-phase each defender in exploration phase communicates with every other defender in the same phase, whether or not to re-enter the exploration phase in the next epoch based on whether every defender has *Restart* to be True. Each defender has her own communicating arm, corresponding to her internal rank. When the defender i is communicating, she sends a bit at a round to the defender k by deciding which arm to pull: a 1 bit is sent by pulling the communicating arm of defender k (a collision occurs and collision feedback is received by defender k) and a 0 bit is sent by pulling her own arm.

During this phase if a defender has *Restart* to be True, then she sends a 1 bit to each defender in order to signal a restart. If a defender has *Restart* to be False, then she sends a 0 bit to every defender. Crucially, an attacker can change this 0 bit to a 1 bit by inducing adversarial collisions, however the attacker cannot change a 1 bit to a 0 bit, since collisions from one defender to another cannot be reversed by an attacker. Therefore if some defender in the exploration phase has insufficient observations on some arm, every defender restarts, and enters an exploration phase in the next epoch.

Inter-Communication. (Subroutine 3 lines 40-44) This sub-phase is used for defenders in exploration phase to communicate with the defenders in the exploitation phase within the same epoch so that they can synchronize in the next epoch. In the inter-communication sub-phase, if *SufficientObservations* is True, then the defender pulls Opt[1] for N consecutive rounds to signal to any possible defenders in the exploitation phase, that they should enter the exploration phase in the next epoch.

Exploitation. (Subroutine 2) In the exploitation phase, the defender sequentially hops over the optimal set of arms throughout the entire epoch. The last N rounds of this protocol is the inter-communication sub-phase within the exploitation phase (Subroutine 2 lines 5-9) used to receive communication from defenders in exploration phase. If a defender in exploitation phase experiences a collision during the inter-communication phase, then this is interpreted as a signal from at least one defender in the exploration phase to enter the exploration in the next epoch.

4.2 Analysis of the RESYNC Algorithm

Theorem 1 bounds the expected regret incurred by RESYNC given the attack cost C.

Theorem 1. *Assume N defenders run* RESYNC *against M attackers, with $\Delta = \mu_{(N)} - \mu_{(N+1)}$. Given the attack cost is C, the expected regret of* RESYNC *is bounded by* $O\left(CN^3 + CKN\frac{\log(K^2T)}{\Delta^2}\right)$.

The proof of Theorem 1 is composed of three key arguments whose main ideas are presented below. Formal proof appears in the extended version of this paper [17].

We begin by upper bounding the number of rounds required for all defenders to collect sufficient observations for each arm, to determine the optimal set of arms with high probability. The following Lemma can be easily derived from [20].

Lemma 1. *If all defenders have collected at least $8\lceil\log(2K^2T)/\Delta^2\rceil$ observations for each arm, then all defenders have determined the optimal set of arms with probability at least $1 - 1/T$.*

The concept of restarting under insufficient observations, and the design of the intra-communication phase where all defenders can communicate whether or not they have sufficient observations for each arm (in the presence of attackers) yields the following two "synchronization" Lemmas:

Lemma 2. *Suppose all defenders are in exploration phase in a certain epoch. If there exists a defender who does not have at least $8\lceil\log(2K^2T)/\Delta^2\rceil$ observations for each arm, then all defenders re-enter the exploration phase in the next epoch.*

Lemma 3. *All defenders have collected at least $8\lceil\log(2K^2T)/\Delta^2\rceil$ reliable observations for each arm after at most $T_B + C$ rounds since the start of the game*

We then state the third synchronization lemma that completely describes the behaviour of defenders over epochs in the absence of adversarial collisions.

Lemma 4. *Suppose all defenders are in exploration phase or all defenders are in exploitation phase in a certain epoch. If no attacker collides with any defender during this epoch then all defenders enter exploitation phase in the next epoch.*

The following is the situation where desynchronization occurs under adversarial collisions. Suppose all defenders are in exploration phase or all defenders are in exploitation phase in a certain epoch. If the attackers cause N_1 defenders trigger restart and N_2 to not, then the N_1 defenders re-enter exploration phase while the N_2 enter exploitation phase. What is crucial is that all defenders are able to synchronize in the next epoch, which the final synchronization lemma proves is true.

Lemma 5. *Suppose $N_1 \geq 1$ defenders are in exploration phase and N_2 defenders are in exploitation phase in a certain epoch. Then independent of the arms pulled by attackers during the epoch, all defenders enter exploration phase in the next epoch.*

The above Lemma holds due to the design of the sensing sub-phase (where defenders in exploration phase can sense the presence of defenders in the exploitation phase in the presence of attackers) and inter-communication sub-phase (where defenders in exploration phase communicate with defenders in exploitation phase and force them to rejoin exploration in the next epoch).

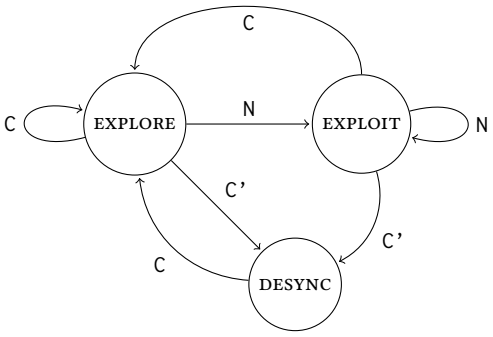

Figure 1: Transition Dynamics of the System of Defenders over epochs

Finally using the synchronization lemmas, we can upper bound the number of epochs in which not all defenders are in exploitation phase, which is the main argument to bound the regret. To achieve this we consider the transition dynamics of the system of defenders over epochs (see Figure 1). For the formal description refer to the extended version of this paper [17].

The states EXPLORE, DESYNC and EXPLOIT in Figure 1 correspond to the state of the bandit game in a certain epoch where all defenders are in exploration phase, at least one defender is in exploration phase and at least one defender is in exploitation phase, or all defenders are in exploitation phase respectively. Similarly the action space {N, C, C'} corresponds to the actions the attackers can take over epochs. Formally, N corresponds to arm pulls during an epoch that cause no collisions with defenders, C to arm pulls during an epoch that cause all defenders to restart exploration on the next epoch, and C' to arm pulls during an epoch that cause $N_1 : 1 < N_1 < N$ defenders to restart respectively. Here we upper bound the number of epochs the state is not EXPLOIT, given that the number of epochs in which actions C and C' are played by the attackers are at most C.

Lemma 6. *(Informal) After $\lceil 1 + C/T_B\rceil$ epochs since the start of the game, the number of epochs in which, not all defenders are in exploitation phase, is at most $O(C)$.*

When all defenders are in exploitation phase, they pull the top N arms with no collisions amongst themselves, conditioned on the event that all of them have determined the optimal set of arms. These arguments show that the leading term in the expected regret is $O(CT_B) = O(CN^3 + KCN\log(K^2T)/\Delta^2)$.

5 THE RESYNC2 ALGORITHM

This section proposes RESYNC2 for the distinguishable collision sensing setting, where defenders can determine whether a collision is from an attacker or a defender. We remove Assumption 1 and show that the additional feedback allows RESYNC2 to have better performance than RESYNC.

The algorithm RESYNC2, consists of three phases that run sequentially, namely, initialization, exploration and exploitation. Note that unlike RESYNC, the exploration and exploitation phases are not

intertwined, thereby removing the necessity of the sensing and inter-communication sub-phases in RESYNC.

Initialization. The purpose of the initialization phase is to estimate the total number of defenders, and assign distinct internal ranks between the defenders. The initialization phase is similar to the one from SIC-MMAB ([6]) although adapted to the distinguishable collision sensing setting, so number of defenders and internal ranks can be determined in the presence of attackers.

Exploration. The exploration phase progresses in epochs of size $2K$, with two sub-phases, sequential hopping (from RESYNC) but which lasts only for K rounds and a modified intra-communication phase (also from RESYNC) that also lasts only for K rounds. The interplay between sequential hopping and inter-communication is similar to that of RESYNC in the sense that, the outcome of sequential hopping (whether one reliable observation was received for each arm during sequential hopping) is communicated through forced collisions in the intra-communication phase. We say an epoch is successful if each defender received one reliable observation for each arm during sequential hopping. The defenders explore until there are $8\lceil\log(2K^2T)/\Delta^2\rceil$ successful epochs.

Exploitation In the exploitation phase, the defenders sequentially hop over the optimal set of arms until the end of the time-horizon.

The complete pseudocode of RESYNC2 is given in the extended version of this paper [17].

The performance of RESYNC2 improves from that of RESYNC due to the additional feedback available in distinguishable collision sensing. This feedback allows defenders to communicate robustly using collisions and collision feedback even in the presence of attackers. That is, they can determine with certainty whether a 1 bit or a 0 bit was sent by a defender during an inter-communication phase, by reading the collision indicator η_k^D. In the previous feedback setting, only η_k is available and the bits sent in this manner between defenders can be modified by an attacker through adversarial collisions. This robust communication allows us to disentangle the exploration and exploitation phases from RESYNC, leading to the size of the epochs to reduce from $O\left(N^2 + K\log(K^2T)/\Delta^2\right)$ to just $O(K)$, thereby improving the regret guarantee.

5.1 Analysis of the RESYNC2 Algorithm

Theorem 2 bounds the expected regret incurred by RESYNC2 given the attack cost C.

THEOREM 2. *Assume N defenders run RESYNC2 against M attackers, with $\Delta = \mu_{(N)} - \mu_{(N+1)}$. Conditioned on the number of attacks being C, the expected regret of RESYNC2 is bounded by*

$$O\left(KCN + KN\frac{\log(K^2T)}{\Delta^2}\right)$$

The proof of Theorem 2 is composed of two key arguments whose main idea is presented below.

We begin by showing that after initialization phase is complete, all defenders have estimated the number of defenders N correctly, and have distinct internal ranks in $[N]$ with high probability. It is worth noting that the initialization phase is successful with high probability, irrespective of the arms pulled by the attackers, whereas if the same initialization phase from [6] was used, this would not be the case.

Then similar to the analysis for Theorem 1, we upper bound the number of rounds required for all defenders to collect sufficient observations for each arm, in order to determine the optimal set of arms with high probability using Lemma 1 and the following Lemma.

LEMMA 7. *After at most $C + 8\log(2K^2T)/\Delta^2$ epochs of exploration, all defenders have collected at least $8\lceil\log(2K^2T)/\Delta^2\rceil$ observations for each arm.*

After each defender has sufficient observations on each arm, all defenders will enter the exploitation phase and never leave this phase, where they will pull the top N arms orthogonally until the end of the time-horizon. These arguments show that the leading term in the expected regret is $O\left(KCN + KN\log(K^2T)/\Delta^2\right)$.

6 REMOVING THE ASSUMPTION THAT Δ IS KNOWN

In this section we highlight how the techniques from RESYNC2 can be combined with SIC-MMAB in order to remove all prior knowledge except T in the distinguishable collision sensing setting. The resulting algorithm will have the structure of SIC-MMAB along with the robust phases presented in RESYNC2. The resulting algorithm will have an initialization phase along with exploration-communication phases. Each exploration phase will contain a sequential hopping and intra-communication phase. There will also be a separate communication phase outside the exploration phase used to communicate arm-statistics with other defenders. For the remainder of this section, we use N_p, K_p to denote the active number of defenders in phase p and active number of arms in phase p respectively.

Initialization. The initialization phase will require the following substitutions of subroutines in SIC-MMAB. The Musical Chairs subroutine from SIC-MMAB must be substituted with the Orthogonalization subroutine from RESYNC2 and the Estimate_M subroutine from SIC-MMAB must be substituted with the Estimate-Defenders subroutine from RESYNC2. This must be done to ensure the initialization phase is robust to adversarial collisions.

Communication. Next as pointed out in Section 5, communication can be made robust (i.e. defenders can exchange bits in the presence of attackers) by relying on the collision indicator η_k^D instead of η_k. Therefore the communication phase from SIC-MMAB can be preserved by substituting each occurrence of η_k in the algorithm with η_k^D.

Exploration. Finally the exploration phase will need to inherit the intra-communication sub phase from RESYNC2 in order to ensure robust exploration. That is, the outcome of sequential hopping (whether one reliable observation was received for each arm during sequential hopping) must be communicated through forced collisions in the intra-communication phase. In the p-th exploration-communication phase, the exploration phase will progress in epochs of size $2K_p$, and until there are 2^p successful epochs. By successful epoch we mean that each defender recieves one reliable observation for each active arm in that epoch. During the first K_p rounds of an epoch the defenders sequentially hop the active set of arms and the next K_p rounds of exploration is an intra-communication phase as

Subroutine 4: Exploration Phase

```
π ← j-th active arm, epochNumber ← 0
while epochNumber ≤ 2^p do
    for K_p rounds do
        Restart ← False
        π ← π + 1 (mod [K_p])
        Pull π and receive η_π^D, η_π^A, r_π(t)
        η_π ← η_π^D ∨ η_π^A
        if η_π = 0 then
            s[π] ← s[π] + r_π(t)
        else
            Restart ← True
    for K_p rounds do
        if Restart then
            Pull K_p[1]
        else
            π ← π + 1 (mod [K_p])
            Pull π and receive η_π^D
            if η_π^D = 1 then
                Restart ← True
    if not Restart then
        epochNumber ← epochNumber +1
```

in RESYNC2. The following exploration phase must be substituted in SIC-MMAB.

Finally the statistics are updated according to the description in SIC-MMAB. The technical innovations required to save SIC-MMAB from attackers are to incorporate our robust initialization, exploration and communication phases that we have described here which were inherited from our algorithm RESYNC2. For the expected regret an additional factor of $O(CK)$ is incurred to the regret bound in Theorem 1 from [6] due to the robust exploration phase detailed above (for details of the bound $O(CK)$ refer Proof of Theorem 2). The final bound on expected regret of the resulting algorithm is $O\left(CK + \sum_{k>N} \frac{\log T}{\mu_{(N)}-\mu_{(k)}} + MK \log T\right)$ where C is the number of collisions from attackers. For contrast, the expected regret of SIC-MMAB in the presence of no attackers is $O\left(\sum_{k>N} \frac{\log T}{\mu_{(N)}-\mu_{(k)}} + MK \log T\right)$.

7 LOWER BOUNDS

In this section, we consider lower bounds on expected regret of the defenders, when the attack cost is C, and will show that the expected regret of the defenders has a linear dependence on the attack cost.

The following theorem establishes a lower bound on the expected regret of any MP-MAB algorithm in terms of C.

THEOREM 3. *There exists an attacker with expected number of attacks at most C, for which any MP-MAB algorithm suffers expected regret* $\Omega(NC/K)$.

PROOF. The attacker samples an arm $k \sim \mathcal{U}(K)$, pulls k for C rounds, and pulls no arm for the remaining rounds. Clearly the number of collisions any defender will face from the attacker is at most C. The sampled arm is in the optimal set of arms with probability $\frac{N}{K}$. Under this event, during the first C rounds, any algorithm has per-round regret at least $\mu_{(N)} - \mu_{(N+1)}$. So, the expected regret over T rounds under this attack is at least $C \cdot (\mu_{(N)} - \mu_{(N+1)}) \cdot \frac{N}{K} = \Omega\left(\frac{NC}{K}\right)$. □

Hence, this lower bound establishes that both RESYNC and RESYNC2 exhibit order-optimal behaviour in terms of C.

8 DISCUSSION

We studied a setting in which N defenders collaborate to minimize regret from a multi-armed bandit where several players simultaneously pull arms and M attackers disrupt collaboration between defenders. We showed that even when $M = 1$, existing algorithms, including algorithms robust to jammers and selfish players, incur linear regret with only logarithmic number of collisions from the attacker. We thus proposed the algorithm RESYNC and RESYNC2 based on restarting synchronously under adversarial collisions in which the performance deteriorates gracefully as the number of collisions from multiple attackers increases. We then provided lower bound that proves that the regret scales linearly with the number of collisions from attackers. In conclusion, we establish that our proposed algorithms are order-optimal in terms of the attack cost.

This work leaves several questions open. Firstly, although the assumption that a lower bound on Δ is known can be removed in the distinguishable collision sensing setting (refer Section 6), by combining our synchronization mechanism with a generalization of the Successive-Eliminations algorithm ([19]) as in SIC-MMAB [6], it is unclear what algorithmic mechanism can be used in the non-distinguishable collision sensing setting when Δ is unknown and attackers exist in the game. Next, it would be interesting to remove Assumption 1 in the non-distinguishable collision sensing setting, either with algorithms that robustly estimate the value N online or by using an approach that does not require knowledge of N (as in [5]). Further, one may look at robustness to adversarial collisions in the no-sensing setting where no collision information is observed and the heterogeneous setting where the arm means vary among players.

REFERENCES

[1] Pragnya Alatur, Kfir Y. Levy, and Andreas Krause. 2022. Multi-Player Bandits: The Adversarial Case. *J. Mach. Learn. Res.* 21, 1, Article 77 (jun 2022), 23 pages.

[2] Animashree Anandkumar, Nithin Michael, Ao Kevin Tang, and Ananthram Swami. 2011. Distributed Algorithms for Learning and Cognitive Medium Access with Logarithmic Regret. *IEEE Journal on Selected Areas in Communications* 29, 4 (2011), 731–745. https://doi.org/10.1109/JSAC.2011.110406

[3] Orly Avner and Shie Mannor. 2022. Concurrent Bandits and Cognitive Radio Networks. In *Machine Learning and Knowledge Discovery in Databases* (Nancy France). Springer-Verlag, Berlin, Heidelberg, 66–81. https://doi.org/10.1007/978-3-662-44848-9_5

[4] Lilian Besson and Emilie Kaufmann. 2018. Multi-Player Bandits Revisited. In *Proceedings of Algorithmic Learning Theory (Proceedings of Machine Learning Research, Vol. 83)*, Firdaus Janoos, Mehryar Mohri, and Karthik Sridharan (Eds.). PMLR, 56–92. https://proceedings.mlr.press/v83/besson18a.html

[5] Ilai Bistritz and Amir Leshem. 2018. Distributed Multi-Player Bandits - a Game of Thrones Approach. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.), Vol. 31. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2018/file/c2964caac096f26db222cb325aa267cb-Paper.pdf

[6] Etienne Boursier and Vianney Perchet. 2019. SIC-MMAB: Synchronisation Involves Communication in Multiplayer Multi-Armed Bandits. In *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.), Vol. 32. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2019/file/c4127b9194fe8562c64dc0f5bf2c93bc-Paper.pdf

[7] Etienne Boursier and Vianney Perchet. 2020. Selfish Robustness and Equilibria in Multi-Player Bandits. In *Proceedings of Thirty Third Conference on Learning Theory (Proceedings of Machine Learning Research, Vol. 125)*, Jacob Abernethy and Shivani Agarwal (Eds.). PMLR, 530–581. https://proceedings.mlr.press/v125/boursier20a.html

[8] Sébastien Bubeck, Yuanzhi Li, Yuval Peres, and Mark Sellke. 2020. Non-Stochastic Multi-Player Multi-Armed Bandits: Optimal Rate With Collision Information, Sublinear Without. In *Proceedings of Thirty Third Conference on Learning Theory (Proceedings of Machine Learning Research, Vol. 125)*, Jacob Abernethy and Shivani Agarwal (Eds.). PMLR, 961–987. https://proceedings.mlr.press/v125/bubeck20c.html

[9] Sumit J. Darak and Manjesh K. Hanawal. 2019. Multi-Player Multi-Armed Bandits for Stable Allocation in Heterogeneous Ad-Hoc Networks. *IEEE Journal on Selected Areas in Communications* 37, 10 (2019), 2350–2363. https://doi.org/10.1109/JSAC.2019.2934003

[10] James W. Friedman. 1971. A Non-cooperative Equilibrium for Supergames12. *The Review of Economic Studies* 38, 1 (01 1971), 1–12. https://doi.org/10.2307/2296617 arXiv:https://academic.oup.com/restud/article-pdf/38/1/1/4362169/38-1-1.pdf

[11] Wei Huang, Richard Combes, and Cindy Trinh. 2022. Towards Optimal Algorithms for Multi-Player Bandits without Collision Sensing Information. In *Proceedings of Thirty Fifth Conference on Learning Theory (Proceedings of Machine Learning Research, Vol. 178)*, Po-Ling Loh and Maxim Raginsky (Eds.). PMLR, 1990–2012. https://proceedings.mlr.press/v178/huang22a.html

[12] Wassim Jouini, Damien Ernst, Christophe Moy, and Jacques Palicot. 2009. Multi-armed bandit based policies for cognitive radio's decision making issues. 1 – 6. https://doi.org/10.1109/ICSCS.2009.5412697

[13] Jianwu Li, Zebing Feng, Zhiyong Feng, and Ping Zhang. 2015. A survey of security issues in Cognitive Radio Networks. *China Communications* 12, 3 (2015), 132–150. https://doi.org/10.1109/CC.2015.7084371

[14] Gábor Lugosi and Abbas Mehrabian. 2022. Multiplayer Bandits Without Observing Collision Information. *Math. Oper. Res.* 47, 2 (may 2022), 1247–1265. https://doi.org/10.1287/moor.2021.1168

[15] Akshayaa Magesh and Venugopal V. Veeravalli. 2019. Multi-player Multi-Armed Bandits with non-zero rewards on collisions for uncoordinated spectrum access. *CoRR* abs/1910.09089 (2019). arXiv:1910.09089 http://arxiv.org/abs/1910.09089

[16] Akshayaa Magesh and Venugopal V. Veeravalli. 2019. Multi-User MABs with User Dependent Rewards for Uncoordinated Spectrum Access. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*. 969–972. https://doi.org/10.1109/IEEECONF44664.2019.9048964

[17] Shivakumar Mahesh, Anshuka Rangi, Haifeng Xu, and Long Tran-Thanh. 2022. Multi-Player Bandits Robust to Adversarial Collisions. arXiv:2211.07817 [cs.LG]

[18] Abbas Mehrabian, Etienne Boursier, Emilie Kaufmann, and Vianney Perchet. 2020. A Practical Algorithm for Multiplayer Bandits when Arm Means Vary Among Players. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 108)*, Silvia Chiappa and Roberto Calandra (Eds.). PMLR, 1211–1221. https://proceedings.mlr.press/v108/mehrabian20a.html

[19] Vianney Perchet and Philippe Rigollet. 2013. The multi-armed bandit problem with covariates. *The Annals of Statistics* 41, 2 (apr 2013). https://doi.org/10.1214/13-aos1101

[20] Jonathan Rosenski, Ohad Shamir, and Liran Szlak. 2016. Multi-Player Bandits – a Musical Chairs Approach. In *Proceedings of The 33rd International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 48)*, Maria Florina Balcan and Kilian Q. Weinberger (Eds.). PMLR, New York, New York, USA, 155–163. https://proceedings.mlr.press/v48/rosenski16.html

[21] Suneet Sawant, Rohit Kumar, Manjesh K. Hanawal, and Sumit J. Darak. 2020. Learning to Coordinate in a Decentralized Cognitive Radio Network in Presence of Jammers. *IEEE Transactions on Mobile Computing* 19, 11 (2020), 2640–2655. https://doi.org/10.1109/TMC.2019.2927475

[22] Chengshuai Shi and Cong Shen. 2021. An Attackability Perspective on No-Sensing Adversarial Multi-player Multi-armed Bandits. In *2021 IEEE International Symposium on Information Theory (ISIT)*. 533–538. https://doi.org/10.1109/ISIT45174.2021.9517726

[23] Chengshuai Shi, Wei Xiong, Cong Shen, and Jing Yang. 2020. Decentralized Multi-player Multi-armed Bandits with No Collision Information. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 108)*, Silvia Chiappa and Roberto Calandra (Eds.). PMLR, 1519–1528. https://proceedings.mlr.press/v108/shi20a.html

[24] Chengshuai Shi, Wei Xiong, Cong Shen, and Jing Yang. 2021. Heterogeneous Multi-player Multi-armed Bandits: Closing the Gap and Generalization. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 22392–22404. https://proceedings.neurips.cc/paper/2021/file/bcb3303a96a92dc38c12992941de7627-Paper.pdf

[25] Beibei Wang, Yongle Wu, K.J. Ray Liu, and T. Charles Clancy. 2011. An anti-jamming stochastic game for cognitive radio networks. *IEEE Journal on Selected Areas in Communications* 29, 4 (2011), 877–889. https://doi.org/10.1109/JSAC.2011.110418

[26] PoAn Wang, Alexandre Proutiere, Kaito Ariu, Yassir Jedra, and Alessio Russo. 2020. Optimal Algorithms for Multiplayer Multi-Armed Bandits. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 108)*, Silvia Chiappa and Roberto Calandra (Eds.). PMLR, 4120–4129. https://proceedings.mlr.press/v108/wang20m.html

[27] Qian Wang, Kui Ren, Peng Ning, and Shengshan Hu. 2016. Jamming-Resistant Multiradio Multichannel Opportunistic Spectrum Access in Cognitive Radio Networks. *IEEE Transactions on Vehicular Technology* 65, 10 (2016), 8331–8344. https://doi.org/10.1109/TVT.2015.2511071