

Cooperative and Stochastic Multi-Player Multi-Armed Bandit: Optimal Regret With Neither Communication Nor Collisions

Sébastien Bubeck
Microsoft Research

Thomas Budzinski
UBC

Mark Sellke
Stanford University

November 6, 2020

Abstract

We consider the cooperative multi-player version of the stochastic multi-armed bandit problem. We study the regime where the players cannot communicate but have access to shared randomness. In prior work by the first two authors, a strategy for this regime was constructed for two players and three arms, with regret $\tilde{O}(\sqrt{T})$, and with no collisions at all between the players (with very high probability). In this paper we show that these properties (near-optimal regret and no collisions at all) are achievable for any number of players and arms. At a high level, the previous strategy heavily relied on a 2-dimensional geometric intuition that was difficult to generalize in higher dimensions, while here we take a more combinatorial route to build the new strategy.

1 Introduction

We consider the cooperative multi-player version of the classical stochastic multi-armed bandit problem. We denote by m the number of players and by $K \geq m$ the number of arms. The bandit instance is described by the mean rewards $\mathbf{p} = (p(1), \dots, p(K)) \in [0, 1]^K$, which is unknown to the players. Denote $(Y_t(i))_{1 \leq i \leq K, 1 \leq t \leq T}$ for a sequence of independent random variables such that $\mathbb{P}(Y_t(i) = 1) = p(i)$ and $\mathbb{P}(Y_t(i) = 0) = 1 - p(i)$. At each time step $t = 1, \dots, T$, each player $X \in [m]$ chooses an action $i_t^X \in [K]$, and observes the corresponding reward $Y_t(i_t^X)$. We define the regret by:

$$R_T = T \cdot \max_{\mathbf{a} \in \{0,1\}^K: \sum_{i=1}^K a(i)=m} \mathbf{a} \cdot \mathbf{p} - \sum_{t=1}^T \sum_{X=1}^m p(i_t^X).$$

We assume that once the game has started the players cannot communicate at all (but they can agree on a strategy prior to the game starting). Usually in these cooperative multi-player bandit problems the players are also penalized for *collisions*, i.e. time steps t such that $i_t^X = i_t^Y$ for some distinct players $X \neq Y$. Here instead, following the prior work [BB20], we consider a seemingly daunting constraint on the players that subsumes such penalty: we ask that with high probability (with respect to the reward generation process) they simply *do not collide at all*. Our main result reads as follows:

Theorem 1.1. *There exists a randomized strategy for the players based on shared randomness¹, such that for any $\mathbf{p} \in [0, 1]^K$, we have the regret bound*

$$\mathbb{E}[R_T] \leq O\left(mK^{11/2} \sqrt{T \log(T)}\right),$$

and furthermore with probability at least $1 - \frac{1}{T}$ (with respect to the reward generation process) the players never collide.

¹The shared randomness assumption only requires the players to have access to a public source of randomness (e.g., unpredictable weather patterns). Indeed, the “adversary” that selects the bandit instance \mathbf{p} can have access to this source too, but the point is that once \mathbf{p} is chosen it is fixed for the rest of the game, and so the choice cannot depend on future outcomes of the randomness source.

Related works. The multiplayer bandit problem with limited communication was first introduced roughly at the same time in [LJP08, LZ10, AMTS11], and has been extensively studied since then [AM14, RSS16, BBM⁺17, LM18, BP19, ALK19, BLPS20], with various assumptions on the communication/collisions. Yet it is only in [BB20] that it was realized that one could in fact obtain the optimal regret *without any collisions at all*. The latter result was however limited to 2 players and 3 actions. Indeed the strategy was based on a simple 2-dimensional construction, reproduced here in Figure 2. The extension even to 4 actions seemed very difficult. In the present paper we take a more combinatorial route, which lends itself better to the high-dimensional picture. The analogue of Figure 2 for our new strategy is Figure 2 (see below, in the high-level overview of our strategy, for more details on the meaning of the partitions depicted in these figures).

Paper organization. We start in Section 2 with our main construction, a certain “stable” partition of the cube $[0, 1]^K$. Following [BB20] we then put this partition to work in Section 3 on a simpler version of the problem, the so-called *full information case*, where the players observe at each round a reward on *all* the arms, but these rewards are independent realizations for each player. Finally, in Section 4, we show how to deal with the extra difficulty of exploration for the bandit setting, and we complete the proof of Theorem 1.1.

High level overview of our strategy. Our strategy is based on a (time-dependent) colored partition of $[0, 1]^K$, where the colors correspond to m -tuples of arms. At a given time step, players decide which arm to play based on the color of the partition element that contains their respective current empirical estimate of \mathbf{p} . We describe our partition using a decision tree, where each node is labeled with a set of currently undecided arms (i.e. for which we have not yet decided whether they belong to the top m actions or not) and the children of this node correspond to the possible subdivisions of this undecided set. This naturally gives a hierarchical decomposition of $[0, 1]^K$, and one of our key insights is to assign a partition element not only to the leaves of this decomposition, but also to any part which is *close* to a decision boundary at some inner node. The effect of the latter modification is a certain *stability* property of the partition: as long as the empirical estimates of the players remain close, the partition elements that decide the players’ actions will be neighbors in the decision tree (Lemma 2.2). Thus to avoid collision it suffices to find a coloring such that two players never collide if they play the arms recommended by two neighboring nodes in the decision tree (Lemma 2.4). What about the regret of such a strategy? If players play according to a leaf of the tree, then in fact they will act optimally (Lemma 2.3) or close to optimally, up to the deviations between \mathbf{p} and its estimates. The question is how to control the regret from inner nodes, which correspond to the interfaces on Figure 2, where the players make suboptimal choices. For this, one needs to have the geometric picture in mind: to stay at an inner node means that one was close to a decision boundary, which will only happen for a “small” fraction of the instances \mathbf{p} . To turn things around, as in [BB20], we choose a *random* decision boundary, so that for any fixed instance \mathbf{p} , the probability to be in the boundary will be small. It remains to carefully analyze how the “small” probability relates to the regret from playing suboptimally in this boundary (Lemma 3.1).

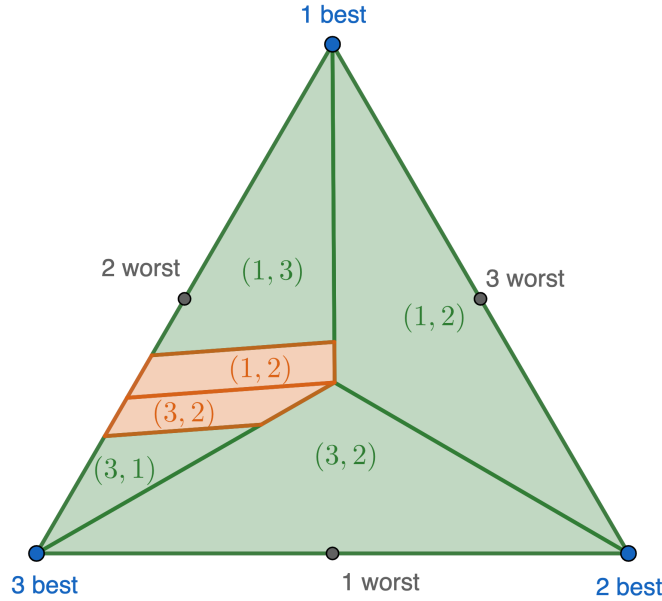
2 A partition of the hypercube

In this section, which comprises the main new contribution of this work, we construct our randomized partition of the hypercube $[0, 1]^K$ together with the corresponding piecewise constant strategy for the m players. This was the main barrier in extending [BB20] beyond the case of 2 players and 3 actions.

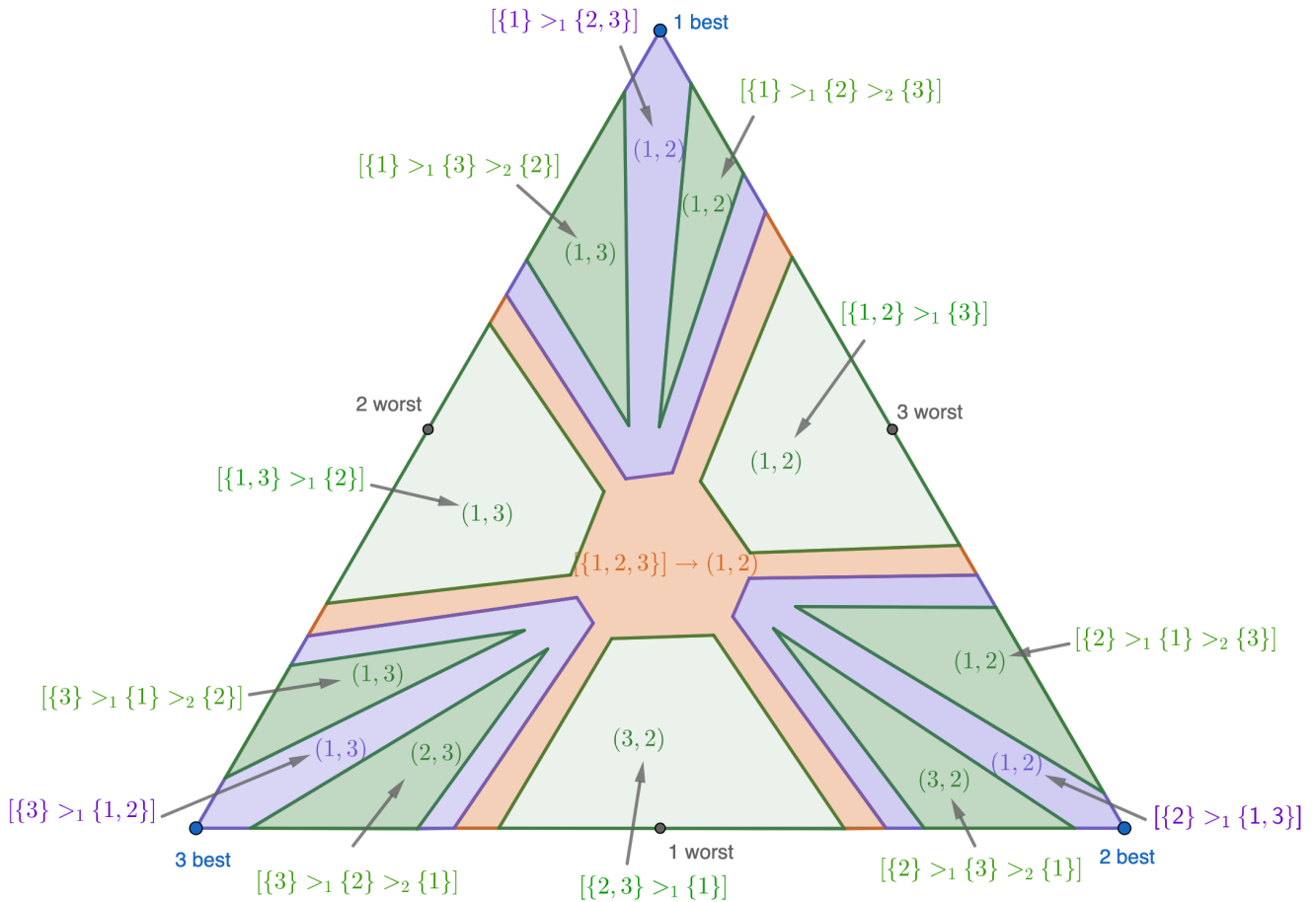
2.1 The tree $\mathcal{T}_{K,m}$

The most basic goal of our construction is to partition $[0, 1]^K$ based on the order of the coordinates. We begin by giving the combinatorial setup which underlies this construction. We first consider the class of *ordered partitions* of $[K]$. These are set partitions of $[K]$ where the elements within a part are not ordered, but the parts are ordered. That is, an ordered set partition of $[K]$ has the form

$$P = [S_1 > S_2 > \cdots > S_j],$$



The basic partition in [BB20], restricted to a plane where $p(1) + p(2) + p(3)$ is constant. The parts are labelled by the actions each player plays based on their estimate of \mathbf{p} . Crucially, adjacent regions never result in collisions. Except for the small orange region, the players always play the top 2 actions.



Our new partition for 2 players and 3 actions, also restricted to a plane where $p(1) + p(2) + p(3)$ is constant. The parts are labelled by pairs of actions as before, as well as vertices of the tree $\mathcal{T}_{K,m}$ defined in subsection 2.1. There are now two levels of “skeleton” shown in orange and purple, separating the main parts in which the top 2 actions are always played. Again adjacent regions never result in a collision. In contrast with the partition above, our new partition extends to general (K, m) .

where $(S_i)_{i=1}^j$ partition $[K]$. For example $[\{1, 3, 5\} > \{2, 6, 7\} > \{4\}]$ is an ordered set partition of $\{1, 2, 3, 4, 5, 6, 7\}$ and is identical to $[\{5, 1, 3\} > \{6, 2, 7\} > \{4\}]$. We will construct these inequalities “one at a time” and it turns out we need to keep track of the order in which they are introduced. Therefore we define a *doubly ordered partition* (henceforth DOP) to be an ordered set partition in which the inequality signs are themselves ordered. Thus a DOP of $[K]$ has the form

$$P = [S_1 >_{\sigma(1)} S_2 >_{\sigma(2)} \cdots >_{\sigma(j-1)} S_j]$$

for some permutation $\sigma \in \mathfrak{S}_{j-1}$. For example

$$[\{1, 3, 5\} >_1 \{2, 6, 7\} >_2 \{4\}] \quad \text{and} \quad [\{1, 3, 5\} >_2 \{2, 6, 7\} >_1 \{4\}]$$

are the two DOPs with underlying ordered partition $[\{1, 3, 5\} > \{2, 6, 7\} > \{4\}]$. The set of DOPs that we have just defined has a natural tree structure, and we denote this tree by \mathcal{T}_K . More precisely, the root of \mathcal{T}_K is the trivial DOP $\text{ROOT} := [\{1, 2, \dots, K\}]$ and, for every DOP

$$P_1 = [S_1 >_{\sigma(1)} S_2 >_{\sigma(2)} \cdots >_{\sigma(i-1)} S_i >_{j-1} S_{i+1} >_{\sigma(i+1)} \cdots >_{\sigma(j-1)} S_j]$$

that is ROOT (i.e. with $j \geq 2$), the parent of P_1 is

$$P = [S_1 >_{\sigma(1)} S_2 >_{\sigma(2)} \cdots >_{\sigma(i-1)} S_i \cup S_{i+1} >_{\sigma(i+1)} \cdots >_{\sigma(j-1)} S_j].$$

In other words, descending the tree \mathcal{T}_k amounts to adding inequalities $>_1, >_2, \dots$ in this order.

The tree we will work with throughout this paper is a subtree $\mathcal{T}_{K,m} \subseteq \mathcal{T}_K$. The reason is that we are only concerned with identifying the set of the top m actions, potentially without knowing the relative order within these top m . To focus on this, we introduce the following definitions. Let $i(P)$ be the largest integer $i \geq 0$ such that $\sum_{j=1}^i |S_j| \leq m$ (for example $i(\text{ROOT}) = 0$). We define the set $A(P) := S_1 \cup \dots \cup S_{i(P)}$ (with the convention $A(P) = \emptyset$ if $i(P) = 0$), which corresponds to the set of actions that the DOP P has already identified as being in the top m actions. We now define the set $B(P)$ of actions that needs to be partitioned further to fully identify the top m actions: if $|A(P)| = m$ then $B(P) := \emptyset$, and otherwise $B(P) := S_{i(P)+1}$. We can now define $\mathcal{T}_{K,m} \subseteq \mathcal{T}_K$ as the subtree formed by paths from the root where only inequalities involving $B(P)$ may be added to a DOP P at any time. In other words, we define $\mathcal{T}_{K,m}$ recursively as follows: let

$$P_1 = [S_1 >_{\sigma(1)} S_2 >_{\sigma(2)} \cdots >_{\sigma(i-1)} S_i >_{j-1} S_{i+1} >_{\sigma(i+1)} \cdots >_{\sigma(j-1)} S_j]$$

be a DOP and let

$$P = [S_1 >_{\sigma(1)} S_2 >_{\sigma(2)} \cdots >_{\sigma(i-1)} S_i \cup S_{i+1} >_{\sigma(i+1)} \cdots >_{\sigma(j-1)} S_j]$$

be its parent. If $P \in \mathcal{T}_{K,m}$, then $P_1 \in \mathcal{T}_{K,m}$ if and only if $B(P) = S_i \cup S_{i+1}$. See Figure 3 for an example. We also denote by $\mathcal{L}(\mathcal{T}_{K,m})$ the set of leaves of the tree $\mathcal{T}_{K,m}$. Note that the leaves of $\mathcal{T}_{K,m}$ are DOPs which determine the top m actions. However not all DOPs determining the top m actions are leaves of $\mathcal{T}_{K,m}$. For example we have

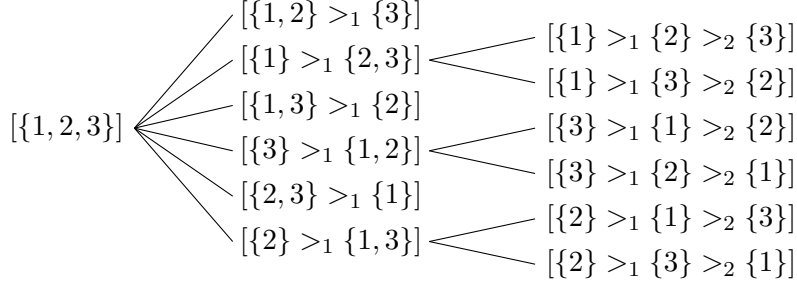
$$[\{4, 8\} >_2 \{2, 6, 7\} >_1 \{1, 3, 5\}] \in \mathcal{T}_{8,2} \quad \text{but} \quad [\{4, 8\} >_1 \{2, 6, 7\} >_2 \{1, 3, 5\}] \notin \mathcal{T}_{8,2}.$$

The latter holds because the parent DOP $[\{4, 8\} >_1 \{1, 3, 5, 2, 6, 7\}]$ determines the top 2 actions hence is already a leaf of $\mathcal{T}_{8,2}$.

For $P \in \mathcal{T}_{K,m}$ we use $\text{CHILDREN}(P)$ to denote its set of children, and $\text{PARENT}(P)$ to denote its (unique) parent. For example we have

$$\text{PARENT}([\{1, 3, 5\} >_1 \{2, 6, 7\} >_2 \{4\}]) = [\{1, 3, 5\} >_1 \{2, 4, 6, 7\}].$$

For convenience we will treat $\mathcal{T}_{K,m}$ as a partial order, so that $Q \preceq P$ means Q is an ancestor of P . In particular, the root satisfies $\text{ROOT} \preceq P$ for any $P \in \mathcal{T}_{K,m}$. Finally we denote by $d_{\mathcal{T}_{K,m}}$ the graph distance in the tree $\mathcal{T}_{K,m}$.



The tree $\mathcal{T}_{3,2}$, with 9 leaves and 4 inner nodes.

2.2 Inequalities to define a region

Our goal is now to connect DOPs to regions inside $[0, 1]^K$. To do so we will use certain types of inequalities involving the values of the coordinates of $\mathbf{x} = (x(i)) \in [0, 1]^K$. We first need a few definitions.

Definition 1. Let $\mathbf{x} \in [0, 1]^K$ and $P \in \mathcal{T}_{K,m} \setminus \mathcal{L}(\mathcal{T}_{K,m})$. We define

$$\text{range}_P(\mathbf{x}) := \max_{k \in B(P)} x(k) - \min_{\ell \in B(P)} x(\ell).$$

Definition 2. Let $\mathbf{x} \in [0, 1]^K$ and P be a DOP of the form:

$$P = [S_1 >_{\sigma(1)} S_2 >_{\sigma(2)} \cdots >_{\sigma(i-1)} S_i >_{\sigma(i)} S_{i+1} >_{\sigma(i+1)} \cdots >_{\sigma(j-1)} S_j]. \quad (1)$$

We define

$$\text{gap}_P(\mathbf{x}) = \min_{k \in S_i} x(k) - \max_{\ell \in S_{i+1}} x(\ell).$$

In words, $\text{range}_P(\mathbf{x})$ represents the range of values in the set of coordinates for which the DOP P has not yet identified whether they are in the top m actions or not. On the other hand, $\text{gap}_P(\mathbf{x})$ represents how large was the “cut” made by the DOP P when we added its last inequality. The next easy lemma says that there always exists a “large cut”.

Lemma 2.1. Let $\mathbf{x} \in [0, 1]^K$ and $P \in \mathcal{T}_{K,m} \setminus \mathcal{L}(\mathcal{T}_{K,m})$. There exists a DOP $Q \in \text{CHILDREN}(P)$ such that

$$\text{gap}_Q(\mathbf{x}) \geq \frac{1}{K} \cdot \text{range}_P(\mathbf{x}) \geq 0.$$

Proof. Write $B(P) = \{a_1, \dots, a_\ell\}$, with $x(a_1) \geq x(a_2) \geq \dots \geq x(a_\ell)$ (note that $\ell \geq 2$ since P is not a leaf). The pigeonhole principle implies that some adjacent pair of values $x(a_j), x(a_{j+1})$ will differ by at least $\frac{\text{range}_P(\mathbf{x})}{K}$, so one can simply set Q to be the child of P which separates $B(P)$ into $\{x(a_1), \dots, x(a_j)\}$ and $\{x(a_{j+1}), \dots, x(a_\ell)\}$. \square

2.3 Constructing the partition

We now finally construct the partition of $[0, 1]^K$, depending in a deterministic way on a function $c : \mathcal{T}_{K,m} \rightarrow [0, \frac{1}{K}]$ as well as a small constant $\varepsilon > 0$. The partition elements will be indexed by vertices of the tree $\mathcal{T}_{K,m}$, or in other words the partition is defined by a mapping $\mathcal{P}_{c,\varepsilon} : [0, 1]^K \rightarrow \mathcal{T}_{K,m}$. This mapping is easiest to describe algorithmically, which we do in Algorithm 1.

Let us now comment on what Algorithm 1 is doing (the reader might also find useful to look at Figure 2 at the same time). First, lines 15 and 16 are the recursive step, where we decide which is the next inequality that we add to our DOP. This decision depends on the parameters $c(P)$. Moreover, line 8 for $Q = P$ means that if one of these decisions is 6ε -close to make, we simply output P . This corresponds to the interfaces of width 6ε between large cells on Figure 2. Note that the positions of

Algorithm 1: Definition of the mapping $\mathcal{P}_{c,\varepsilon} : [0, 1]^K \rightarrow \mathcal{T}_{K,m}$.

```

1 parameters:  $c : \mathcal{T}_{K,m} \rightarrow [0, \frac{1}{K}]$ ,  $\varepsilon > 0$ , input:  $x \in [0, 1]^K$ , returns: vertex  $P \in \mathcal{T}_{K,m}$ .
2 Initialize  $P = \text{ROOT}$ .
3 while  $P \notin \mathcal{L}(\mathcal{T}_{K,m})$  do
4   for  $Q \preceq P$  do
5     Write  $B(Q) = \{a_1, \dots, a_\ell\}$ , with  $x(a_1) \geq x(a_2) \geq \dots \geq x(a_\ell)$ .
6     for  $j = 1, 2, \dots, \ell - 1$  do
7       Define the child  $Q_j$  of  $Q$  by splitting  $B(Q)$  into  $\{a_1, \dots, a_j\} > \{a_{j+1}, \dots, a_\ell\}$ .
8       if  $|\text{gap}_{Q_j}(\mathbf{x}) - c(Q) \cdot \text{range}_Q(\mathbf{x})| \leq (d_{\mathcal{T}_{K,m}}(P, Q) + 1) \cdot 6\varepsilon$  then
9         return  $P$ 
10      end
11   end
12   Write  $B(P) = \{a_1, \dots, a_\ell\}$ , with  $x(a_1) \geq x(a_2) \geq \dots \geq x(a_\ell)$ .
13   for  $j = 1, 2, \dots, \ell - 1$  do
14     Define the child  $P_j$  of  $P$  by splitting  $B(P)$  into  $\{a_1, \dots, a_j\} > \{a_{j+1}, \dots, a_\ell\}$ .
15     if  $\text{gap}_{P_j}(\mathbf{x}) \geq c(P) \cdot \text{range}_P(\mathbf{x})$  then
16       // By Lemma 2.1 and  $c(P) \leq \frac{1}{K}$ , this occurs for at least one  $j \in [\ell - 1]$ .
17        $P \leftarrow P_j$ 
18       break (go back to line 3)
19   end
20 return  $P$ 

```

these interfaces depend on the $c(P)$. Moreover, for a given Q , the condition of line 15 becomes weaker as the algorithm progresses and P gets deeper in the tree $\mathcal{T}_{K,m}$. Therefore, a point that was very close to being assigned to Q will be assigned to its child. This results in “coating” of the interfaces with layers of width 6ε . For example, on Figure 2, this corresponds to the region $[\{1\} >_1 \{2, 3\}]$ separating the regions $[\{1, 2, 3\}]$ and $[\{1\} >_1 \{2\} >_2 \{3\}]$. In general, there can be up to K such interface layers.

A crucial property of the partition is the following stability property. The first item will be useful to ensure the absence of collisions. The goal of the second is to state a “consistency” property for different values of ε , which will be needed later in the bandit analysis.

Lemma 2.2. *We fix $\varepsilon > 0$, $c : \mathcal{T}_{K,m} \rightarrow [0, \frac{1}{K}]$, and $\mathbf{x}, \mathbf{y} \in [0, 1]^K$.*

1. *If $\|\mathbf{x} - \mathbf{y}\|_{\ell^\infty} \leq \varepsilon$, then $d_{\mathcal{T}_{K,m}}(\mathcal{P}_{c,\varepsilon}(\mathbf{x}), \mathcal{P}_{c,\varepsilon}(\mathbf{y})) \leq 1$.*
2. *Let $P \in \mathcal{T}_{K,m}$ and assume that $|x(i) - y(i)| \leq \varepsilon$ for all $i \in A(P) \cup B(P)$. Let also $\varepsilon' \in (0, \varepsilon]$. Then it is not possible that $\mathcal{P}_{c,\varepsilon}(\mathbf{x})$ and $\mathcal{P}_{c,\varepsilon'}(\mathbf{y})$ are descendants of two distinct children of P .*

Proof. We start with the second point, since it will be useful for the first one. Assume that we run Algorithm 1 on both \mathbf{x} with (c, ε) and on \mathbf{y} with (c, ε') in parallel and that both instances are currently both on P (in the while loop). Then we need to prove that the two instances do not branch into two distinct children of P . If P is the final output for either \mathbf{x} or \mathbf{y} then the conclusion is immediate. If not, we denote by $P_j(\mathbf{x})$ (resp. $P_j(\mathbf{y})$) the j -th child of P considered by Algorithm 1 run on \mathbf{x} (resp. on \mathbf{y}). Note that the coordinates may be ordered differently in \mathbf{x} and in \mathbf{y} , so it not clear that the children of P considered for \mathbf{x} and \mathbf{y} should be the same. Assume that an instruction $P \leftarrow P_j$ occurs for \mathbf{x} at $j = j_{\mathbf{x}}$ (resp. at $j = j_{\mathbf{y}}$ for \mathbf{y}). If $P \leftarrow P_{j_{\mathbf{x}}}(\mathbf{x})$ occurs for \mathbf{x} in line 16 of the algorithm, we have

$$\text{gap}_{P_{j_{\mathbf{x}}}(\mathbf{x})}(\mathbf{x}) \geq c(P) \cdot \text{range}_P(\mathbf{x}). \quad (2)$$

On the other hand, since P was not the output for \mathbf{x} , the inequality of line 8 is not satisfied, which means that the distance between the left and the right-hand side of the last display is at least 6ε , i.e.

$$\text{gap}_{P_{j_{\mathbf{x}}}(\mathbf{x})}(\mathbf{x}) \geq c(P) \cdot \text{range}_P(\mathbf{x}) + 6\varepsilon. \quad (3)$$

Hence, since $|\mathbf{x} - \mathbf{y}|_{\ell^\infty} \leq \varepsilon$, we also have

$$\text{gap}_{P_{j_{\mathbf{x}}(\mathbf{x})}}(\mathbf{y}) > c(P) \cdot \text{range}_P(\mathbf{y}).$$

In particular, the fact that the left-hand side is positive means that $P_{j_{\mathbf{x}}(\mathbf{x})}$ is also one of the children of P considered when the algorithm is run on \mathbf{y} , i.e. $P_{j_{\mathbf{x}}(\mathbf{y})} = P_{j_{\mathbf{x}}(\mathbf{x})}$. Moreover, the last display means that the inequality in line 15 is satisfied when the algorithm considers $P_{j_{\mathbf{x}}(\mathbf{y})} = P_{j_{\mathbf{x}}(\mathbf{x})}$ for \mathbf{y} . This proves that $j_{\mathbf{y}} \leq j_{\mathbf{x}}$.

We now assume $j_{\mathbf{y}} < j_{\mathbf{x}}$ and will reach a contradiction (note that the argument is not symmetric in \mathbf{x} and \mathbf{y} since we assume $|\mathbf{x} - \mathbf{y}| \leq \varepsilon$ and not $|\mathbf{x} - \mathbf{y}| \leq \varepsilon'$). Similarly to Equation (3), using the fact that P is not the output for \mathbf{x} and that $P \leftarrow P_{j_{\mathbf{y}}}$ does not occur when we run the algorithm on \mathbf{x} , we have

$$\text{gap}_{P_{j_{\mathbf{y}}(\mathbf{x})}}(\mathbf{x}) \leq c(P) \cdot \text{range}_P(\mathbf{x}) - 6\varepsilon.$$

Using $|\mathbf{x} - \mathbf{y}|_{\ell^\infty} \leq \varepsilon$, we deduce

$$\text{gap}_{P_{j_{\mathbf{y}}(\mathbf{x})}}(\mathbf{y}) \leq c(P) \cdot \text{range}_P(\mathbf{y}) - 4\varepsilon.$$

On the other hand, for the exact same reason as in Equation (3), we have

$$\text{gap}_{P_{j_{\mathbf{y}}(\mathbf{y})}}(\mathbf{y}) \geq c(P) \cdot \text{range}_P(\mathbf{y}) + 6\varepsilon' > c(P) \cdot \text{range}_P(\mathbf{y}).$$

From the last two displays, we obtain

$$\left| \text{gap}_{P_{j_{\mathbf{y}}(\mathbf{x})}}(\mathbf{y}) - \text{gap}_{P_{j_{\mathbf{y}}(\mathbf{y})}}(\mathbf{y}) \right| > 4\varepsilon.$$

On the other hand, we write $B(P) = \{a_1, \dots, a_\ell\} = \{b_1, \dots, b_\ell\}$, where $x(a_1) \geq \dots \geq x(a_\ell)$ and $y(b_1) \geq \dots \geq y(b_\ell)$. Then by definition of gap, the last display becomes

$$\left| (y(b_{j_{\mathbf{y}}}) - y(b_{j_{\mathbf{y}}+1})) - (y(a_{j_{\mathbf{y}}}) - y(a_{j_{\mathbf{y}}+1})) \right| > 2\varepsilon. \quad (4)$$

On the other hand, using the assumption $|\mathbf{x} - \mathbf{y}|_{\ell^\infty}$, we have

$$\left| y(b_{j_{\mathbf{y}}}) - y(a_{j_{\mathbf{y}}}) \right| \leq \left| y(b_{j_{\mathbf{y}}}) - x(a_{j_{\mathbf{y}}}) \right| + \left| x(a_{j_{\mathbf{y}}}) - y(a_{j_{\mathbf{y}}}) \right| \leq 2\varepsilon$$

and similarly for $j_{\mathbf{y}} + 1$. This contradicts Equation (4), so we obtain $j_{\mathbf{x}} = j_{\mathbf{y}}$. As explained in the first part of the proof, we have $P_{j_{\mathbf{x}}(\mathbf{x})} = P_{j_{\mathbf{x}}(\mathbf{y})}$, so both instances of the algorithm branch in this child of P , which proves the second item of the Lemma.

We now prove the first item. We first use the second item with $\varepsilon' = \varepsilon$. Note that the assumption $|\mathbf{x} - \mathbf{y}|_{\ell^\infty} \leq \varepsilon$ is stronger than the assumption needed on the second item. Therefore, we know that when we run the algorithm on \mathbf{x} and \mathbf{y} in parallel, the runs agree as long as both instances are still running. Now assume without loss of generality that \mathbf{x} stops first. If \mathbf{x} stops at a leaf $P \in \mathcal{L}(\mathcal{T}_{K,m})$ then we already know by the second item that \mathbf{y} also stops at P . So let us assume that \mathbf{x} stops at an inner node P , and that \mathbf{y} continues to a child P' of P (if \mathbf{y} continues it must be at a child of P by the second item). We argue now that \mathbf{y} must in fact stop at P' , which concludes the proof. First note that since \mathbf{x} stops at P , it means that line 8 occurred for some child Q_j of $Q \preceq P$ obtained by splitting $B(Q) = \{a_1, \dots, a_\ell\}$ into $\{a_1, \dots, a_j\} > \{a_{j+1}, \dots, a_\ell\}$. In other words one has:

$$\left| \text{gap}_{Q_j}(\mathbf{x}) - c(Q) \cdot \text{range}_Q(\mathbf{x}) \right| \leq (d_{\mathcal{T}_{K,m}}(P, Q) + 1) \cdot 6\varepsilon.$$

Now observe that, since $|\mathbf{x} - \mathbf{y}|_{\ell^\infty} \leq \varepsilon$, one has $|\text{range}_Q(\mathbf{x}) - \text{range}_Q(\mathbf{y})| \leq 2\varepsilon$. Next denote b_1, \dots, b_ℓ a permutation of a_1, \dots, a_ℓ such that $y(b_1) \geq \dots \geq y(b_\ell)$ and let j' be the index j of the largest $y(j)$ which is ε -close to $x(a_j)$. Now consider the child Q' of Q obtained by splitting $B(Q)$ into $\{b_1, \dots, b_{j'}\} > \{b_{j'+1}, \dots, b_\ell\}$. One has $|\text{gap}_{Q'}(\mathbf{y}) - \text{gap}_{Q_j}(\mathbf{x})| \leq 2\varepsilon$. In particular we obtain by the triangle inequality:

$$\left| \text{gap}_{Q'}(\mathbf{y}) - c(Q) \cdot \text{range}_Q(\mathbf{y}) \right| \leq 4\varepsilon + (d_{\mathcal{T}_{K,m}}(P, Q) + 1) \cdot 6\varepsilon = (d_{\mathcal{T}_{K,m}}(P', Q) + 1) \cdot 6\varepsilon.$$

In particular, in the run on \mathbf{y} , when the while loop is at P' , line 8 will occur when invoked on Q' , which shows that the run stops at P' . \square

The next lemma establishes three basic guarantees for Algorithm 1. In particular, it ensures that the DOP associated to \mathbf{x} describes the order of the coordinates of \mathbf{x} , e.g. if $\mathcal{P}_{c,\varepsilon}(\mathbf{x})$ is of the form $\{\{1\} > \{2\} > \dots > \{K\}\}$, then the coordinates of \mathbf{x} are in nonincreasing order.

Lemma 2.3. *Suppose Algorithm 1 is run on $\mathbf{x} \in [0, 1]^K$. Suppose that the operation $P \leftarrow \widehat{P}$ occurs during Algorithm 1 (in Line 16). Then:*

1. \mathbf{x} obeys all the inequalities of \widehat{P} .
2. If for any ancestor $Q \preceq P$ and any child Q_j of Q the inequality

$$|\text{gap}_{Q_j}(\mathbf{x}) - c(Q) \cdot \text{range}_Q(\mathbf{x})| \leq 10K\varepsilon. \quad (5)$$

never holds, and if $\mathbf{y} \in [0, 1]^K$ satisfies $|\mathbf{x} - \mathbf{y}|_{\ell^\infty} \leq \varepsilon$, then $P \leftarrow \widehat{P}$ also occurs when Algorithm 1 is run on \mathbf{y} .

3. If $\widehat{P} \notin \mathcal{L}(\mathcal{T}_{K,m})$ is not a leaf and is the final output of Algorithm 1, then there exists some ancestor $Q \preceq \widehat{P}$ with a child $Q_j \in \text{CHILDREN}(Q)$ such that inequality (5) holds.

Proof. To show the first assertion, we simply observe that Algorithm 1 sorts the actions of $B(P)$ in line 5 according to their values at the point \mathbf{x} , hence every inequality sign added is true for \mathbf{x} .

For the second assertion, we observe that the factor $10K$ in Equation 5 is so large that, by the triangle inequality, Line 8 of the algorithm applies to neither \mathbf{x} nor \mathbf{y} for any Q_j a child of $Q \preceq P$. Moreover for these Q_j and Q , the signs of $(\text{gap}_{Q_j}(\mathbf{x}) - c(Q) \cdot \text{range}_Q(\mathbf{x}))$ and $(\text{gap}_{Q_j}(\mathbf{y}) - c(Q) \cdot \text{range}_Q(\mathbf{y}))$ always agree, so x, y behave identically in Line 15. Since x, y do not terminate before \widehat{P} in Line 8 and behave identically in Line 15 before $P \leftarrow \widehat{P}$, we conclude that $\widehat{P} \leftarrow P$ occurs when Algorithm 1 is run on \mathbf{y} .

Finally for the last assertion, because \widehat{P} is not a leaf, Line 8 of Algorithm 1 must have been the reason to terminate the algorithm on \mathbf{x} . As $\mathcal{T}_{K,m}$ has depth at most K , inequality 5 must hold just before termination. \square

2.4 Coloring the partition

We now want to turn our partition into a full strategy. For this, we need a rule specifying, for each DOP P and each player X , which arm player X should play in the partition element P . We construct here a “robust” rule, such that when eventually combined with Lemma 2.2 it will give a collision-free strategy for the players. We start with two definitions.

Definition 3. For a DOP P , define $\text{Feas}_P \subseteq \binom{[K]}{m}$ to consist of all m -subsets of $[K]$ which comprise the top m actions in some total ordering extending P .

Note that this is more stringent than only requiring that each element might individually be in the top m . In particular, sequences in Feas_P contain all elements of $A(P)$ and a fixed size subset of $B(P)$.

Definition 4. An m -coloring of $\mathcal{T}_{K,m}$ is a function $F : \mathcal{T}_{K,m} \rightarrow [K]^m$. An m -coloring $F = (f_1, \dots, f_m)$ is called *collision-robust* if for any $P, Q \in \mathcal{T}_{K,m}$ with $d_{\mathcal{T}_{K,m}}(P, Q) \leq 1$ and any $i, j \in [m]$ with $f_i(P) = f_j(Q)$, one must necessarily have $i = j$.

Now, fix c, ε and suppose we are given an arbitrary function $G : \mathcal{T}_{K,m} \rightarrow \binom{[K]}{m}$ such that $G(P) \in \text{Feas}_P$ for all P . Then we claim that there is a robust m -coloring F such that $F(P)$ is a permutation of $G(P)$ for all P . To see this, we simply proceed recursively down the tree. We choose $F(\text{ROOT})$ to be the lexicographically first (i.e. sorted) permutation of $G(\text{ROOT})$, and then for $P \in \mathcal{T}_{K,m}$ choose $F(P)$ to be a permutation of $G(P)$ with maximum possible overlap with $F(\text{PARENT}(P))$ (if there are several such permutations, pick the first lexicographically). That is, if $f_i(\text{PARENT}(P)) \in G(P) \cap G(\text{PARENT}(P))$, we have $f_i(P) = f_i(\text{PARENT}(P))$. It is easy to see that this produces a robust m -coloring. Hence, we have shown the following.

Lemma 2.4. *For any*

$$G : \mathcal{T}_{K,m} \rightarrow \binom{[K]}{m}$$

such that $G(P) \in \text{Feas}_P$ for all P , there is a collision-robust coloring

$$F : \mathcal{T}_{K,m} \rightarrow [K]^m,$$

where $F(P)$ is always a permutation of $G(P)$.

3 The full feedback scenario

We consider here the *full information* version of the problem described in the introduction, where the players observe at each round a reward on *all* the arms (not only the one played as in the bandit case), but these rewards are independent realizations for each player. We propose to use the construction from the previous section to build a strategy for this full information game. We fix a robust m -coloring F of $\mathcal{T}_{K,m}$ given by applying Lemma 2.4 to the function G which sets $G(P)$ to be the lexicographically first element of Feas_P . We take $c : \mathcal{T}_{K,m} \rightarrow [0, \frac{1}{K}]$ to be an i.i.d. uniform function of the distance to the root in $\mathcal{T}_{K,m}$. That is, we set

$$c(P) = C(d(P, \text{ROOT})),$$

where $C(0), C(1), \dots, C(K-1)$ are i.i.d. uniform variables on $[0, \frac{1}{K}]$. For each time $t \in [T]$, we write

$$\varepsilon_t := 10 \sqrt{\frac{\log(mKT)}{t}}.$$

The strategy followed by player X at time t is then the following:

1. Denote by $\mathbf{q}_t^X = (q_t^X(i)) \in [0, 1]^K$ the vector of empirical mean rewards of the arms observed by X from time 1 to time $t-1$.
2. Apply Algorithm 1 to find $\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^X) \in \mathcal{T}_{K,m}$.
3. Play the action $i_t^X = f_X(\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^X))$, where $F = (f_1, \dots, f_m)$.

The key observation is the following lemma that will be used to bound the probability that Algorithm 1 stops at an inner node. This is important to estimate, since this event may result in suboptimal arm choices.

Lemma 3.1. *Let $\delta > 0$ and $\mathbf{x} \in [0, 1]^K$. For any $P \in \mathcal{T}_{K,m}$ of depth h and any child Q of P , we have*

$$\mathbb{P}(|\text{gap}_Q(\mathbf{x}) - C(h) \cdot \text{range}_P(\mathbf{x})| \leq \delta \mid C(0), \dots, C(h-1)) \leq \frac{2K\delta}{\text{range}_P(\mathbf{x})}.$$

Proof. By construction $C(h)$ is uniform in $[0, \frac{1}{K}]$ even under the conditioning. Moreover, the set of values $C(h)$ for which the event on the left occurs is an interval of length $\frac{2\delta}{\text{range}_P(\mathbf{x})}$, which implies the claim. \square

Our main result for this section is as follows:

Theorem 3.2. *The strategy described above satisfies for any $\mathbf{p} \in [0, 1]^K$:*

$$\mathbb{E}[R_T] \leq O(mK^4 \sqrt{T \log(T)}).$$

Furthermore with probability at least $1 - \frac{1}{T}$ the players never collide.²

Proof. Let us first prove the non-collision property. By the Hoeffding inequality and a crude union bound over times, players and arms, with probability at least $1 - \frac{1}{T}$, it holds that for every t and each player X , we have

$$|\mathbf{q}_t^X - \mathbf{p}|_{\ell^\infty} \leq \frac{1}{3} \varepsilon_t. \tag{6}$$

Moreover, if this occurs, for any two players X and Y , we have $|\mathbf{q}_t^X - \mathbf{q}_t^Y| < \varepsilon_t$, so by Lemma 2.2 the DOPs $\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^X)$ and $\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^Y)$ are neighbour vertices in $\mathcal{T}_{K,m}$. By our choice of F using Lemma 2.4, this ensures that there is no collision, i.e. that X and Y do not play the same arm.

From now on, all the information that we will use in the proof will be the description of the mappings $\mathcal{P}_{c, \varepsilon_t}$ together with the ‘‘stability’’ result of (6). We highlight right now that the exact same argument will be needed in the end of the proof in the bandit setting (Section 4).

We now control the regret under the event (6) for a fixed time step t . For $\mathbf{x} \in [0, 1]^K$, we denote by $P^0(\mathbf{x}) = \text{ROOT}$, $P^1(\mathbf{x})$, $P^2(\mathbf{x}), \dots$, the random (because of c) path in $\mathcal{T}_{K,m}$ visited during the while loop of Algorithm 1 (line 3) when we run it on \mathbf{x} with parameters c and ε_t . If the output of the algorithm

² m can be replaced easily by $\min(m, K-m)$. The same holds in Theorem 1.1.

is at depth k in $\mathcal{T}_{K,m}$, then we denote $P^h(\mathbf{x}) = P^k(\mathbf{x})$ for any $h \geq k$. Note that $P^1(\mathbf{x}), \dots, P^h(\mathbf{x})$ only depends on $C(0), \dots, C(h-1)$. We also write $P_j^h(\mathbf{x})$ for the child of $P^h(\mathbf{x})$ obtained by splitting $B(P^h(\mathbf{x})) = \{a_1, \dots, a_\ell\}$, where $x(a_1) \geq x(a_2) \geq \dots \geq x(a_\ell)$, into $\{a_1, \dots, a_j\} > \{a_{j+1}, \dots, a_\ell\}$. For $h \geq 0$ and $j \geq 1$, we consider the events:

$$E_{h,j} = \left\{ \left| \text{gap}_{P_j^h(\mathbf{p})}(\mathbf{p}) - C(h) \cdot \text{range}_{P^h(\mathbf{p})}(\mathbf{p}) \right| \leq 10K\varepsilon_t \right. \\ \left. \text{and } \left| \text{gap}_{P_i^k(\mathbf{p})}(\mathbf{p}) - C(k) \cdot \text{range}_{P^k(\mathbf{p})}(\mathbf{p}) \right| > 10K\varepsilon_t \text{ for any } k < h \text{ and any } i \right\}.$$

The first key observation is that on the event $E_{h,j}$, it must be that $P^h(\mathbf{p})$ is a node of depth h , and furthermore for any \mathbf{x} such that $\|\mathbf{x} - \mathbf{p}\|_{\ell^\infty} \leq \varepsilon_t$ one must also have $P^h(\mathbf{x}) = P^h(\mathbf{p})$. These follow from items 3, 2 of Lemma 2.3 - part 3 shows $P^h(\mathbf{p})$ has depth h and part 2 shows $P^h(\mathbf{x})$ takes the same value. Recalling item 1 of the same lemma, on the event $E_{h,j}$, we know that the total regret of the m players is upper bounded by $m \cdot \text{range}_{P^h(\mathbf{p})}(\mathbf{p})$. Indeed, the top m actions of \mathbf{p} can only differ from the players' actions in the choice of subset of $B(P^h(\mathbf{x})) = B(P^h(\mathbf{p}))$. The second key observation is that by Lemma 3.1 one has

$$\mathbb{P}(E_{h,j} | C(0), \dots, C(h-1)) \leq \frac{20K^2\varepsilon_t}{\text{range}_{P^h(\mathbf{p})}(\mathbf{p})}.$$

Thus we obtain that the contribution to the regret from $E_{h,j}$ is bounded by $20mK^2\varepsilon_t$. If none of the events $E_{h,j}$ occurs, then the players play exactly the top m actions according to \mathbf{p} , hence incur zero regret. Indeed, this follows from the same discussion as above using Lemma 2.3, but if $P^h(\mathbf{p})$ is a leaf of $\mathcal{T}_{K,m}$, then $\text{range}_{P^h(\mathbf{p})}(\mathbf{p}) = 0$. Finally, summing over h, j and t yields the regret estimate

$$\mathbb{E}[R_T] \leq O\left(mK^4 \sum_{t=1}^T \varepsilon_t\right). \quad (7)$$

The theorem follows. \square

4 The bandit scenario

For the bandit version, we again use mappings of the form $\mathcal{P}_{c,\varepsilon_t}$, with the exact same c as in the full information case but a larger ε_t . The important difference will be that, in order to avoid neglecting the exploration of some of the arms, we will use a different, randomized coloring of $\mathcal{T}_{K,m}$. Specifically, at each time t we apply an uniformly random permutation $\pi_t : [K] \rightarrow [K]$ to the actions in defining the lexicographic ordering used in Section 2.4, where the π_t are independent. This defines a π_t -random coloring of the vertices of $\mathcal{T}_{K,m}$ and preserves the collision-robustness of Lemma 2.4. Moreover, by symmetry, the randomness of π_t causes each $F(P)$ to contain a uniformly random subset of $B(P)$ of the appropriate size $m - |A(P)|$, and in particular to contain any arm $i \in B(P)$ with probability at least $\frac{1}{K}$.

We can now describe the strategy. We let $n_t^X(i)$ be the number of times player $X \in [m]$ sampled arm $i \in [K]$ in the first $t-1$ time steps, and let $r_t^X(i) \leq n_t^X(i)$ be the amount of reward observed so far. We let $q_t^X(i)$ be the empirical estimate of $p(i)$ by player X at the start of time t , defined by

$$q_t^X(i) = \frac{r_t^X(i)}{n_t^X(i)} \in [0, 1].$$

For the first $T_0 = 10^9 K \log(KT)$ time-steps, we simply have player X play arm $X + t \pmod{K}$ at time t . After that, at time $t > T_0$ the players as before play via the mapping $\mathcal{P}_{c,\varepsilon_t}$, i.e. player X plays arm $f_X(\mathcal{P}_{c,\varepsilon_t}(\mathbf{q}_t^X))$, where $F = (f_1, \dots, f_m)$ is our π_t -random coloring, c is a uniform function of the distance to the root as in Section 3, and

$$\varepsilon_t = 10000 \sqrt{\frac{K^3 \log(KT)}{t}}.$$

We now begin the analysis. We define the events:

$$\begin{aligned}\Omega_1 &= \left\{ \forall t \geq T_0, i \in [K], X \in [m], \text{ we have } |q_t^X(i) - p(i)| < \frac{\varepsilon_{n_t^X(i)}}{100K^{3/2}} \right\}, \\ \Omega_2 &= \left\{ \forall t \geq T_0, X \in [m], \forall P \in \mathcal{T}_{K,m}, \text{ if } \forall s \leq t \text{ we have } \mathcal{P}_{c,\varepsilon_s}(\mathbf{q}_s^X) \preceq P, \right. \\ &\quad \left. \text{then } \forall i \in A(P) \cup B(P), n_t^X(i) \geq \left\lfloor \frac{t}{2K} \right\rfloor \right\}, \\ \Omega &= \{\Omega_1 \text{ and } \Omega_2\}.\end{aligned}$$

We observe that

$$\frac{\varepsilon_{t/(2K)}}{100K^{3/2}} \leq \frac{\varepsilon_t}{10K}.$$

This means that when the conclusions of Ω hold, we have

$$|q_t^X(i) - p(i)| < \frac{\varepsilon_t}{10K}.$$

Lemma 4.1. *Using the above strategy, for any choice of $(C(h))$ and any $\mathbf{p} \in [0, 1]^K$, we have:*

$$\mathbb{P}[\Omega] \geq 1 - \frac{1}{T}.$$

Proof. We show that each of Ω_1, Ω_2 have probability at least $1 - \frac{1}{2T}$. For Ω_1 this follows immediately from Hoeffding's inequality.

We now show $\mathbb{P}[\Omega_2] \geq 1 - \frac{1}{2T}$. This is where we will use the randomization of the coloring using the π_t to explore evenly. For t small, we use the initial sampling phase (this is the only place we use the initial phase). Indeed, for $t \leq \frac{3T_0}{2}$ the inequality $n_t^X(i) \geq \frac{t}{2K}$ is immediate given our initial T_0 rounds of perfectly uniform sampling. Now, fix $t \geq \frac{3T_0}{2}$, $X \in [m]$, and $i \in [K]$. Let $E_{t,X}(i)$ denote the event that there is P for which $i \in A(P) \cup B(P)$ and, for all $s \leq t$, we have $\mathcal{P}_{c,\varepsilon_s}(\mathbf{q}_s^X) \preceq P$. Since we use a uniform random permutation π_t at each time $s \leq t$, conditionally on everything that happened before, the probability for X to play i is at least $\frac{1}{K}$. It follows that

$$\mathbb{P}\left(E_{t,X}(i) \text{ but } n_t^X(i) < \frac{t}{2K}\right) \leq \mathbb{P}\left(\text{Bin}\left(t - T_0, \frac{1}{K}\right) \leq \frac{t - T_0}{2K}\right).$$

As $t - T_0 \geq 1000K \log(KT)$, the right hand probability is at most $\frac{1}{2mKT^2}$ by applying the multiplicative Chernoff estimate $\mathbb{P}\left(\text{Bin}(N, p) \leq \frac{Np}{2}\right) \leq e^{-\frac{Np}{8}}$ in [MU17, Theorem 4.5]. Union bounding over all t, X, i concludes the proof. \square

From now on, our reasoning will be deterministic. As time increases and we get further in the tree $\mathcal{T}_{K,m}$, we need to have a reasonable estimate of \mathbf{p} to know which arms to keep exploring, but we also need to explore the right arms to have a good estimate (and avoid collisions). Therefore, we will use a ‘‘bootstrap’’ argument as in [BB20], proving a certain property Γ_t by induction on t .

To define Γ_t , we first need a few definitions. For $k \geq 0$, let τ_k be the first time $t > T_0$ that $\mathcal{P}_{c,\varepsilon_t}(\mathbf{q}_t^X) \succeq P^k$ holds for some P^k of depth $k = d(\text{ROOT}, P^k)$ and some $X^k \in [m]$ (in particular $\tau_0 = T_0 + 1$). Throughout this section, we will keep denoting by P^k the corresponding DOP, and by $X^k \in [m]$ the corresponding player (choosing an arbitrary pair (P^k, X^k) if there are multiple instances at time τ_k). We let $k_t = \max\{k : \tau_k < t\}$. We now introduce the event Γ_t for each $t > T_0$.

Definition 5. For $t > T_0$, we denote by Γ_t the event

$$\Gamma_t = \{\Gamma_t^1 \text{ and } \Gamma_t^2 \text{ and } \Gamma_t^3\},$$

where $\Gamma_t^1, \Gamma_t^2, \Gamma_t^3$ are the three events:

1. For all $r \leq t$ and $X, Y \in [m]$ we have

$$d_{\mathcal{T}_{K,m}}(\mathcal{P}_{c,\varepsilon_r}(\mathbf{q}_r^X), \mathcal{P}_{c,\varepsilon_r}(\mathbf{q}_r^Y)) \leq 1.$$

In particular, there is no collision in the first t time-steps.

2. For any $k, s \geq 0$ satisfying $\tau_k \leq \min(s, t)$ and any $Q^k \neq P^k$ a sibling of P^k , we have

$$\mathcal{P}_{c, \varepsilon_s}(\mathbf{q}_s^Y) \not\subseteq Q^k.$$

3. For any $k \geq 1$ and $s \geq 0$ satisfying $\tau_k \leq \min(s, t)$ and for any $i \in A(P^{k-1}) \cup B(P^{k-1})$, we have

$$|q_s^Y(i) - p(i)| \leq \frac{\varepsilon_{\tau_k}}{10K}.$$

Our goal is to show that Ω implies Γ_T (which of course subsumes Γ_t for all $t \leq T$). We will prove that Γ_t holds inductively in t below in Proposition 4.3. We first state a simple consequence of Γ_t which is useful in proving Proposition 4.3.

Lemma 4.2. *Let $t > T_0$, and suppose that Γ_{t-1} holds. Then:*

1.

$$\text{ROOT} = P^0 \prec P^1 \prec \dots \prec P^{k_t}.$$

2. For any $r \leq t, X \in [m]$, there is $k_r^X \leq k_{t+1}$ such that $\mathcal{P}_{c, \varepsilon_r}(\mathbf{q}_r^X) = P^{k_r^X}$.

3. If $\tau_{k_{t+1}} = t$ holds, then $P^{k_{t+1}} \in \text{CHILDREN}(P^{k_t})$.

Proof. All three items follow immediately from Γ_{t-1}^2 . \square

Proposition 4.3. *If Ω holds, then Γ_T also holds.*

Proof. We prove Γ_t holds for all $t \geq T_0$ by induction on t . First note that $\Gamma_{T_0}^1$ and $\Gamma_{T_0}^2$ follow from the definition of the initial exploration phase, and $\Gamma_{T_0}^3$ is an empty statement, since $\tau_1 > T_0$ by definition.

Hence, we now assume Ω and Γ_{t-1} and prove $\Gamma_t^1, \Gamma_t^2, \Gamma_t^3$.

Proof of Γ_t^3 . We fix $k \geq 1$ and $s \geq 0$ satisfying $\tau_k \leq \min(s, t)$. Since the definition of Γ_{t-1}^3 does *not* require that $s \leq t-1$, there is nothing to prove unless $\tau_{k_{t+1}} = t$. We therefore assume $\tau_{k_{t+1}} = t$. By Lemma 4.2, we have

$$\text{ROOT} = P^0 \prec P^1 \prec \dots \prec P^{k_{t+1}}.$$

Moreover, for any $Y \in [m]$, there is $k_t^Y \leq k_{t+1}$ such that $\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^Y) = P^{k_t^Y}$. Using the event Ω_2 , we obtain for any $i \in A(P^{k-1}) \cup B(P^{k-1})$ the assertion of Γ_t^3 , namely:

$$|q_s^Y(i) - p(i)| \leq \frac{\varepsilon_{n_s^X(i)}}{100K^{3/2}} \leq \frac{\varepsilon_{\tau_k}}{10K}.$$

Proof of Γ_t^1 . Since Γ_{t-1}^1 holds, we only need to prove the statement for $r = t$. Invoking Lemma 4.2, we have:

$$\text{ROOT} = P^0 \prec P^1 \prec \dots \prec P^{k_t}.$$

Hence for each $k \leq k_t$, we have by the definition of Algorithm 1 (and more precisely the fact that we have not returned P^{k-1} according to line 8):

$$\left| \text{gap}_{P^k}(\mathbf{q}_{\tau_k}^{X^k}) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_{\tau_k}^{X^k}) \right| \geq 4\varepsilon_{\tau_k}. \quad (8)$$

Applying Γ_t^3 (proved just above, and whose hypothesis holds for $s \leq t$ as we assume Γ_{t-1}^1) implies that for $t \geq s \geq \tau_k$, $i \in A(P^{k-1}) \cup B(P^{k-1})$ and any $Y \in [m]$ we have

$$|q_s^Y(i) - p(i)| \leq \frac{\varepsilon_{\tau_k}}{10K}. \quad (9)$$

Combining Equation (8) with Equation (9) for (t, Y) and (τ_k, X^k) and using the triangle inequality, we deduce:

$$\left| \text{gap}_{P^k}(\mathbf{q}_t^Y) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_t^Y) \right| \geq 2\varepsilon_{\tau_k}. \quad (10)$$

Applying Equation (9) for (t, Y) and (t, Z) with $Y, Z \in [m]$ and the same k , we obtain:

$$\left| (\text{gap}_{P^k}(\mathbf{q}_t^Y) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_t^Y)) - (\text{gap}_{P^k}(\mathbf{q}_t^Z) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_t^Z)) \right| \leq \frac{\varepsilon_{\tau_k}}{K}. \quad (11)$$

Finally, combining Equations (10) and (11), we conclude:

$$\left| \frac{(\text{gap}_{P^k}(\mathbf{q}_t^Y) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_t^Y)) - (\text{gap}_{P^{k-1}}(\mathbf{q}_t^Z) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_t^Z))}{|\text{gap}_{P^k}(\mathbf{q}_t^Y) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_t^Y)|} \right| \leq \frac{1}{2K}. \quad (12)$$

Therefore, for any k, t with $\tau_k \leq t$ and $Y, Z \in [m]$, if

$$|\text{gap}_{P^k}(\mathbf{q}_s^Y) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_s^Y)| \leq 4d\varepsilon_s$$

holds for some $d \leq K + 1$, then it follows that:

$$|\text{gap}_{P^k}(\mathbf{q}_s^Z) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_s^Z)| \leq 4(d+1)\varepsilon_s.$$

We recall that $\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^Y) = P^{k_t^Y}$ for some $k_t^Y \leq k_{t+1}$. Using the description of Algorithm 1, the last equations mean that if $\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^Y) = P^{k_t^Y}$ is decided because the condition of line 8 is fulfilled for $P = P^{k_t^Y}$ and Q , then the same condition will be fulfilled for $P = P^{k_t^Y+1}$ and the same Q when we run the algorithm for \mathbf{q}_t^Z . Since $\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^Z) = P^{k_t^Z}$, this proves $k_t^Z \leq k_t^Y + 1$. The reasoning is symmetric in Y and Z , so we obtain $d_{\mathcal{T}_{K,m}}(\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^Z), \mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^Y)) \leq 1$. This concludes the inductive step for part 1.

Proof of Γ_t^2 . We fix (k, s, t) satisfying $\tau_k \leq \min(t, s)$. We apply the second item of Lemma 2.2 with $\mathbf{x} = \mathbf{q}_{\tau_k}^{X^k}$, $\mathbf{y} = \mathbf{q}_s^Y$, $\varepsilon = \varepsilon_{\tau_k}$, $\varepsilon' = \varepsilon_s$ and $P = P^{k-1}$. Note that the assumption $|q_{\tau_k}^{X^k}(i) - q_s^Y(i)| \leq \varepsilon_{\tau_k}$ for $i \in A(P^{k-1}) \cup B(P^{k-1})$ is satisfied by the event Γ_t^3 proved above. We obtain that $\mathcal{P}_{c, \varepsilon_{\tau_k}}(\mathbf{q}_{\tau_k}^{X^k}) = P^k$ and $\mathcal{P}_{c, \varepsilon_s}(\mathbf{q}_s^Y)$ cannot be descendants of two distinct children of P^{k-1} , which shows that Γ_t^2 holds. This completes our induction over t and concludes the proof of Proposition 4.3. \square

Having established the fundamental properties on the behavior of our algorithm, we now turn to the regret analysis. Unlike the full-feedback scenario, the estimation error of a coordinate at time t depends not on ε_t but the potentially much larger $\varepsilon_{n_t^X(i)}$. We circumvent this issue in the next lemma by showing that the high-error coordinates can be set to their exactly correct values without affecting the result of the space partition.

Lemma 4.4. *Let $X \in [m]$ be an arbitrary player. Let $t \geq T_0$, and let $k_* = k_{t/(4K^2)} = \max\{k : \tau_k < t/(4K^2)\}$. Define the modified probability estimate $\tilde{\mathbf{q}}_t^X$ by:*

$$\tilde{q}_t^X(i) = \begin{cases} q_t^X(i), & i \in A(P^{k_*}) \cup B(P^{k_*}), \\ p(i), & \text{else.} \end{cases}$$

Then assuming Ω holds, we have

$$\mathcal{P}_{c, \varepsilon_t}(\mathbf{q}_t^X) = \mathcal{P}_{c, \varepsilon_t}(\tilde{\mathbf{q}}_t^X).$$

Proof. Roughly speaking, the reason why this is true is that at each time-step, the arms that have not been explored a lot are not relevant to make further decisions. More precisely, let $k \leq k_*$. The idea will be to prove that

$$\begin{aligned} |\text{gap}_{P^k}(\mathbf{q}_t^X) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_t^X)| &\geq 6K\varepsilon_t, \\ |\text{gap}_{P^k}(\tilde{\mathbf{q}}_t^X) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\tilde{\mathbf{q}}_t^X)| &\geq 6K\varepsilon_t. \end{aligned}$$

Once we show this, it will follow that $\mathbf{q}_t^X, \tilde{\mathbf{q}}_t^X$ behave identically at all stages of Algorithm 1 which involve a coordinate they differ on. The idea is simply that ε_t is much smaller than $\varepsilon_{t/(4K^2)}$, so the interfaces shrank a lot between times $t/(4K^2)$ and t . Hence, if we were outside the interfaces at time $\tau_k < \frac{t}{4K^2}$, we are definitely still outside at time t . Indeed, by the definition of τ_k and Algorithm 1, we have for some $Y \in [m]$:

$$|\text{gap}_{P^k}(\mathbf{q}_{\tau_k}^Y) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_{\tau_k}^Y)| \geq 4\varepsilon_{\tau_k}.$$

From here, applying Γ_7^3 and the triangle inequality shows that

$$\begin{aligned} |\text{gap}_{P^k}(\mathbf{q}_t^X) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\mathbf{q}_t^X)| &> 3\varepsilon_{\tau_k} \geq 6K\varepsilon_t, \\ |\text{gap}_{P^k}(\tilde{\mathbf{q}}_t^X) - c(P^{k-1}) \cdot \text{range}_{P^{k-1}}(\tilde{\mathbf{q}}_t^X)| &> 3\varepsilon_{\tau_k} \geq 6K\varepsilon_t. \end{aligned}$$

Note that the reason why the second inequality holds is that by definition, the quantities $\text{gap}_{P^k}(\mathbf{x})$ and $\text{range}_{P^{k-1}}(\mathbf{x})$ only depend on the coordinates $x(i)$ such that $i \in A(P^{k-1}) \cup B(P^{k-1})$. These are the coordinates for which Γ_7^3 provides an estimate.

Finally, we know from Proposition 4.3 that $\mathcal{P}_{c,\varepsilon_t}(\mathbf{q}_t^X) = P^{k_t^X}$ for some k_t^X , and since the equations just above hold for all $k \leq k_*$ we conclude that $\mathcal{P}_{c,\varepsilon}(\mathbf{q}_t^X), \mathcal{P}_{c,\varepsilon}(\tilde{\mathbf{q}}_t^X) \succeq P^{k_*}$ and moreover that line 8 of Algorithm 1 never comes into effect for $Q = P^k$ with $k < k_*$. We finally observe that $\mathbf{q}_t^X, \tilde{\mathbf{q}}_t^X$ now exactly agree in all still-relevant coordinates $i \in A(P^{k_*}) \cup B(P^{k_*})$ and hence end up in the same region at the end of Algorithm 1. \square

Lemma 4.4 will now imply the final regret bound.

Theorem 4.5. *The expected regret in the bandit case is*

$$O\left(mK^{11/2}\sqrt{T\log(T)}\right).$$

Proof. We assume Ω holds throughout. We then have, for all i :

$$|\tilde{q}_t^X(i) - p_t(i)| \leq \frac{\varepsilon_{n_t^X/(4K^2)}^{(i)}}{100K^{3/2}} \leq \frac{\varepsilon_{t/(10K^3)}}{10K^{3/2}} < \frac{\varepsilon_t}{3}. \quad (13)$$

By Lemma 4.4 we have $\mathcal{P}_{c,\varepsilon_t}(\mathbf{q}_t^X) = \mathcal{P}_{c,\varepsilon_t}(\tilde{\mathbf{q}}_t^X)$. Therefore the regret of the strategy defined by $(\mathbf{q}_t^X)_{t \in [T], X \in [m]}$ is equal to the regret of the “cheating strategy” obtained by playing using the estimates $\tilde{\mathbf{q}}_t^X$ instead of \mathbf{q}_t^X . Now note that by Equation (13), the “cheating strategy” using $\tilde{\mathbf{q}}_t^X$ satisfies Equation (6). Therefore, as argued in the proof of Theorem 3.2, we obtained the regret bound

$$\mathbb{E}[R_T] \leq O\left(mK^4 \sum_{t=1}^T \varepsilon_t\right).$$

The theorem follows by using the definition of ε_t . \square

References

- [ALK19] P. Alatur, K. Y. Levy, and A. Krause. Multi-player bandits: The adversarial case. *Journal of Machine Learning Research (JMLR)*, 2019.
- [AM14] O. Avner and S. Mannor. Concurrent bandits and cognitive radio networks. In *ECML/PKDD*, 2014.
- [AMTS11] A. Anandkumar, N. Michael, A. K. Tang, and A. Swami. Distributed algorithms for learning and cognitive medium access with logarithmic regret. *IEEE Journal on Selected Areas in Communications*, 29(4):731–745, 2011.
- [BB20] Sébastien Bubeck and Thomas Budzinski. Coordination without communication: optimal regret in two players multi-armed bandits. In *COLT*, 2020.
- [BBM⁺17] R. Bonnefoi, L. Besson, C. Moy, E. Kaufmann, and J. Palicot. Multi-armed bandit learning in iot networks: Learning helps even in non-stationary settings. In *International Conference on Cognitive Radio Oriented Wireless Networks*, pages 173–185. Springer, 2017.
- [BLPS20] S. Bubeck, Y. Li, Y. Peres, and M. Sellke. Non-stochastic multi-player multi-armed bandits: Optimal rate with collision information, sublinear without. In *COLT*, 2020.
- [BP19] Etienne Boursier and Vianney Perchet. Sic-mmab: synchronisation involves communication in multiplayer multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 12071–12080, 2019.

- [LJP08] L. Lai, H. Jiang, and H. V. Poor. Medium access in cognitive radio networks: A competitive multi-armed bandit framework. In *2008 42nd Asilomar Conference on Signals, Systems and Computers*, pages 98–102, 2008.
- [LM18] G. Lugosi and A. Mehrabian. Multiplayer bandits without observing collision information. *arXiv preprint arXiv:1808.08416*, 2018.
- [LZ10] K. Liu and Q. Zhao. Distributed learning in multi-armed bandit with multiple players. *IEEE Transactions on Signal Processing*, 58(11):5667–5681, 2010.
- [MU17] Michael Mitzenmacher and Eli Upfal. *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press, 2017.
- [RSS16] J. Rosenski, O. Shamir, and L. Szlak. Multi-player bandits - a musical chairs approach. In *ICML*, 2016.