Dominate or Delete: Decentralized Competing Bandits with Uniform Valuation

*Abishek Sankararaman† *Soumya Basu‡ and Karthik Abinav Sankararaman§

Abstract

We study regret minimization problems in a two-sided matching market where uniformly valued demand side agents (a.k.a. agents) continuously compete for getting matched with supply side agents (a.k.a. arms) with unknown and heterogeneous valuations. Such markets abstract online matching platforms (for e.g. UpWork, TaskRabbit) and falls within the purview of matching bandit models introduced in Liu et al. [24]. The uniform valuation in the demand side admits a unique stable matching equilibrium in the system. We design the first decentralized algorithm - UCB with Decentralized Dominant-arm Deletion (UCB-D3), for matching bandits under uniform valuation that does not require any knowledge of reward gaps or time horizon, and thus partially resolves an open question in [24]. UCB-D3 works in phases of exponentially increasing length. In each phase i, an agent first deletes dominated arms – the arms preferred by agents ranked higher than itself. Deletion follows dynamic explore-exploit using UCB algorithm on the remaining arms for 2^i rounds. Finally, the preferred arm is broadcast in a decentralized fashion to other agents through *pure exploitation* in $(N-1)K$ rounds with N agents and K arms. Comparing the obtained reward with respect to the unique stable matching, we show that UCB-D3 achieves $O(\log(T)/\Delta^2)$ regret in T rounds, where Δ is the minimum gap across all agents and arms. We provide a (orderwise) matching regret lower-bound.

1 Introduction

Online matching markets (e.g. UpWork and Mechanical Turk) are economic platforms that connect demand side, (e.g. businesses in Upwork or Mechanical Turk), to the supply side (e.g. freelancers in Upwork, or crowdworkers in Mechanical Turk). These platforms enable the demand side agents (a.k.a. agents) to make repeated decisions to match with the supply side agent (a.k.a. arms) of their preference. On these platforms, supply side agents when faced with multiple offers, chooses the demand side agent of her choice. With resource limited supply side, the agents thus compete for arms, while navigating uncertainty on the quality of each arm. The multi-agent matching bandit model was introduced recently by Liu. et.al [24] to understand the learning dynamics of competing agents in a two-sided market. Uncertainty in matching markets have recently been studied in various disciplines (*e.g.,* [15] uses an empirical approach and [17] uses an economic approach), and [24] formalizes the problem from a learning perspective, which is the methodology adopted in this paper.

*AS and SB Equal Contribution
†EECS Department, University of California Berkeley. email: abishek@berkeley.edu
‡ECE Department, University of Texas at Austin. email: basusoumya@utexas.edu
§Facebook. email: karthikabinavs@gmail.com

Liu et.al [24] study two paradigms —centralized and decentralized. In the centralized paradigm, each agent (demand side) at each time, shares her preference *truthfully* over the arms (supply side) with a central arbiter. Two algorithms in the centralized setting was proposed —Explore then commit (ETC) and Upper confidence bound (UCB) based, that lead the system to an equilibrium efficiently (to be made precise shortly). In practice, UCB based algorithm is preferable, as unlike ETC, does not need any knowledge of either the time horizon (the time the market operates) or minimum sub-optimality gap, (the difference in average quality of best and second best arm for all agents).

In the decentralized paradigm, no central arbiter exists, and each agent relies only on her own observations to make arm choices. The decentralized paradigm is of utmost importance in multi-agent setting, as sharing observation with a central arbiter is prone to privacy breach, lacks transparency of the arbiter, and is susceptible to untruthful inputs from agents [22, 28]. Only the ETC based algorithm (that requires knowledge on time horizon and sub-optimally gaps) was extended to the decentralized paradigm in [24], leaving open the design of a practical decentralized algorithm. We resolve this for market settings when demand side agents are valued uniformly. Specifically, we design the first decentralized algorithm that neither requires knowledge of the time-horizon, nor of sub-optimality gaps; yet leads the system with uniformly valued agents to equilibrium efficiently.

Our setup consists of N agents and $K \geq N$ arms, with agents (arms) having a preference ordering over arms (agents). The agents are ranked homogeneously (with no two agents tied) across all arms (uniform valuation). The preference ordering of an agent over arms is given by the set of arm-means for that agent, higher the mean, higher the preference. The arm-means are different for different agents (heterogeneous system). Agents do not know the arm-means (and thus their preference) and have to learn them over time. All agents choose an arm each simultaneously in each round (a decentralized system). Thus, in every round, each arm is chosen by any number (0 to N) of agents. An arm matches only with the highest ranked proposing agent (if any), while blocking the remaining agents (if any). Therefore, each agent is either *matched* to her arm of choice and receives a corresponding stochastic reward, or is *blocked* by her arm of choice (which they are notified of) and receive a deterministic 0 reward.

The system is said to be in *equilibrium*, when the matching between arms and the agents is *stable* [16]. A matching corresponds to an allocation of arms to agents, where each arm is allocated to at-most one agent, and every agent is allocated an arm. A matching is stable, if there exists no agent-arm pair, who would mutually prefer each other as opposed to their current partners in the matching. The stable matching is typically not unique. However, when the agents have an uniform valuation, the stable matching is unique [14].

The regret of an algorithm is defined as the difference of cumulative reward attained by the algorithm, and the cumulative reward in equilibrium when agents and arms always match according to the unique stable matching. An algorithm is *efficient* if the regret grows sub-linearly with time; lower the growth rate more the efficiency. Our main contribution, is the design of a provably efficient decentralized algorithm which is oblivious to the time horizon length and the sub-optimality gaps in heterogeneous uniform valuation two-sided markets.

Uniformly valued agents model online crowd-sourcing platforms such as Upwork and TaskRabbit, consisting of workers (arms) that wish to be matched with jobs/employers (agents) (see also [24]). Different workers have different efficacy for the various jobs (captured by the arm means), which are a-priori unknown to the employers. Workers on these platform can choose at-most one offer each time, from among the offers they receive. As different employers have different compensation structure, a ranking of the employers (agents) is induced and we model the worker to take up the more beneficial offer at each time (i.e., matched with the lowest ranked agent choosing that arm).

1.1 Main Contributions

1. Decentralised Phase Based Arm-Elimination Algorithm. We introduce UCB with Decentralized Dominant-arm Deletion (UCB-D3); a decentralized algorithm that does not need the agents to be aware of the arm-gaps or time-horizon. Thus, our algorithm partially[1] resolves the open question on design of decentralized algorithms for matching bandits posed in [24]. Moreover, we observe through simulations (Section 7), that UCB-D3 significantly outperforms ETC, the (only) prior decentralized algorithm and is comparable to the centralized UCB algorithm from [24].

Our algorithm proceeds in phases that grow exponentially, with each phase divided into an regret minimization block and a communication block that is constant. At the beginning of each phase, every agent will *delete* arms that are estimated to be the stable match partners of higher ranked arms and only play from among the arms not deleted according to the UCB algorithm [2]. The intuition for this is as follows; if an agent deletes all the true stable match partners of higher ranked agents, then the best of the remaining arms, is its stable match partner. In UCB-D3 algorithm, agents set the *arm with which it was matched the most in the current phase* as its estimated stable match partner. This we show enables all agents to quickly identify their stable match partner. In the communication block, every agent, communicates its estimated stable match partner to all lower ranked agents.

The two important features in our algorithm are the following.
(i) It is *non-monotone*; namely, if an agent (erroneously) deletes an arm in a given phase, it can still be played in the future. This ensures that no agent deletes its stable match partner forever, and thus incur large regret.
(ii) The UCB in the regret minimization block is computed using all the samples collected by an agent thus far and not just on the sample in the current phase. This ensures we do not loose valuable information, which is key to obtaining low ($O(\log(T))$) regret.

2. Regret upper bound. We show that the regret of any agent ranked $j \geq 2$ over a time horizon T, scales as $O\left(\log(T)/\Delta^2\right)$, where Δ is the smallest arm-gap across all agents.[2] In our system, the errors in an agent leaks to the other agents through the communication block, when an agent signals its estimated stable matched arm. Furthermore, as based on these signals each agent deletes its perceived dominated arms its own error gets amplified. Thus any error in estimating a stable matched arm can create a domino effect that propagates across agents and phases. It is known that a good estimator of the best-arm in a (single agent) standard MAB is the most played arm [10].

[1] Our results hold under Uniform Valuation among the agents.

[2] A precise regret bound involving all arm means is given in Theorem 1.

In our multi-agent setup, we prove that the same estimator localized in each phase reduces these cascades–stopping them in finite expected time, even in the presence of possible erroneous arm deletions and collisions. Our proof relies on an inductive approach, where we show agents stop providing wrong signals in the communication block, sequentially in the order of their rank (see also Figure 1).

3. Regret lower bound. We show through a matching lower bound, that a regret scaling $\Omega\left(\log(T)/\Delta^2\right)$ is unavoidable. In our decentralized and heterogeneous setting, agents can not collaboratively learn the arm-means of other agents. This necessitates $\Omega(\log(T)/\Delta^2)$ exploration for any agent with a sub-optimality gap Δ, which in turn may lead to $\Omega(\log(T)/\Delta^2)$ collisions with the optimal arm for some lower ranked agent. Formalizing this intuition, we prove our instance dependent regret lower bound.

2 Problem Setting

Agents and arms. We consider N agents, and $K \geq N$ arms. The agents are ranked where the rank of any agent $j \in [N]$ is j (which is unknown to the agents). This is without loss of generality, as we can relabel the agents to obtain this. At each time, all agents choose one of the K arms simultaneously, to play and collect a reward. An agent is *matched* to the arm of its choice in a given round, if and only if it is the highest ranked agent playing that arm at that time.[3] Otherwise, it is *blocked*. If agent $j \in [N]$ is matched with arm $k \in [K]$ at any time, then, agent j is receives a stochastic $[0, 1]$ valued reward with mean $\mu_{jk} \in (0, 1)$ independent of all other rewards.[4] If an agent is blocked at any time, then it is notified and receives a deterministic reward of 0.

The arm means $(\mu_{jk})_{j\in[N],k\in[K]}$ are heterogeneous across the agents, and are not known to the agents apriori. Furthermore, for every agent j, the set of K arm-means $(\mu_{jk})_{k\in[K]}$ are all distinct. In the sequel, we denote by $I^{(j)}(t) \in [K]$, to be the arm played by agent j in round t. For each arm $k \in [K]$ and time $t \geq 1$, denote by $M_k(t) = \min\{j : I^{(j)}(t) = k\}$, to be the agent with which it is matched, where the minimum of an empty set is defined to be infinity.

Decentralized algorithms. We consider decentralized algorithms, namely, at each time t, the choice of arm to choose by any agent, must only depend on the events (past arm choices, rewards and blocking) observed by the agent. The agents however are allowed to form a protocol to follow at the beginning before playing arms.

Performance metric - Unique Stable Matching. In our setting, each agent *prefers* arms in the increasing order of arm-means, and each arm *prefers* agents according to the uniform agent ranking. A matching of agents to arms is stable, if there is no agent-arm pair, unmatched in the current matching, that mutually prefer each other than their respective current matches [16]. In our system, as the agents have uniform rank across all arms there exists a unique *stable matching* (which is not true when agents are ranked non-uniformly across arms.) Indeed, in any stable match agent j must match with it's most preferred arm which is not matched with any agent with rank $(j-1)$ or higher[5].

[3] We use the notion that rank l' is higher than rank l if and only if $l < l'$, throughout the paper.

[4] This is done for convenience. Our analysis can be easily adapted to any sub-gaussian reward.

[5] Agents ranked 1 through $j-1$

We compare the performance of any decentralized online learning strategy to an oracle, in which the agents and arms are matched according to this unique *stable matching*. Stable matching is the appropriate benchmark, as it captures the equilibrium of the allocation of arms to agents, when agents are myopic and want to maximize individual rewards as in our setup. (See also in [24]).

We now describe the unique stable matching formally. We note that in what follows due to our assumption that all arm means are unique there is no tie. Agent ranked 1 prefers the arm $k_*^{(1)} = \arg\max_{k\in[K]} \mu_{1k}$ the most, therefore $(1, k_1^*)$ forms a stable match as arm $k_*^{(1)}$ also prefers agent 1 the most. Now, for any agent ranked $2 \leq j \leq N$, the stable match denoted by $k_*^{(j)}$, is defined inductively as $k_*^{(j)} = \arg\max_{k\in[K]\setminus\{k_*^{(1)},\cdots,k_*^{(j-1)}\}} \mu_{jk}$. In words, the stable match for agent ranked j is the best arm from among all arms that do not form stable match to agents ranked 1 through to $j-1$. Recall that $I^{(j)}(t) \in [K]$ denotes the arm chosen by agent j in time t. The regret of any agent $j \in [N]$, after T time steps is $R_T^{(j)} = \sum_{t=1}^{T} \mathbb{E}[\mu_{jk_*^{(j)}} - \mu_{jI^{(j)}(t)} \mathbf{1}_{M_{I^{(j)}(t)}(t)=j}]$.

3 UCB with Decentralized Dominant-arm Deletion (UCB-D3)

3.1 Key Ideas and intuition.

The main idea behind our algorithm is that agents at all times, will *delete* certain set of arms with which it estimates it may collide a lot and thus lose on reward. An arm is *dominated for an agent*, if it is a stable-match partner of some higher ranked agent. Thus, if an agent deletes all the dominated arms, then the best arm among those remaining is its stable-match partner. Moreover, these arms that are not deleted are all necessarily sub-optimal for all higher ranked agents. Thus, deleting the dominated arms also implies that an agent will incur less collisions. The goal of our algorithm is thus, to achieve this "equilibrium" in a decentralized way, whereby all agents, eventually, delete their dominated arms and play only from among the rest.

In order to do so, UCB-D3; proceeds in phases, with the set of active arms chosen at the start of a phase and fixed in the duration of a phase. Each phase $i \in \mathbb{N}$ is split into two blocks, *(i)* a regret minimization block lasting 2^{i-1} rounds and a *(ii)* communication block lasting a constant $(N-1)K$ rounds. During regret minimization, all agents play from among their active set using the standard UCB algorithm [2] and ignoring collisions. During the communication block agents communicate their estimated stable match partners through collisions. The active arms for an agent in the next phase is all the arms except those that were estimated to be stable-match partner to other higher ranked agents *in that phase*. Our algorithm is *non-monotone*; even if an agent deletes an arm in a given phase, it can potentially be active in a future phase, if in the future phase, no higher ranked agent estimates this arm to be their stable match partner.

A key feature in the algorithm, is that agents' estimate of their stable match partner is the *most matched arm in the regret minimization block of the current phase* (not the arm with highest estimated arm or the highest UCB index). In a single agent MAB where the agent plays according to UCB, one can show that the best arm will be the most played for all but random finite times. Using this observation, we design our algorithm so that the highest ranked agent first settles into an equilibrium (its most played arm coincides with its best arm) as it plays according to UCB(except for the communication rounds) without any collisions. The lower ranked agents follow suit one after the other (see also Figure 1).

3.2 Algorithm.

Notation. For an agent with rank $j \in [N]$, arm $k \in [K]$ and time slot $t \in \mathbb{N}$, denote by $N_k^{(j)}(t) \in \mathbb{N} \cup \{0\}$, to be the number of times agent with rank j was matched to arm k, upto and including time t. If $N_k^{(j)}(t) > 0$, denote by $\widehat{\mu}_k^{(j)}(t)$ to be the empirical observed mean of arm k by agent j using all the samples upto and including time t.

The first $(N-1)$ time slots are used to estimate agent's ranks and subsequently UCB-D3 proceeds in phases. From time slots N on wards, the algorithm proceeds in phases, numbered $i \in \mathbb{N}$, with an agents phase being non-decreasing with time. The algorithm is synchronous, i.e., at each time t, all agents are in the same phase. The pseudocode is given in Algorithm 1 and also described below.

• **Rank estimation.** Rank estimation occurs for the first $(N-1)$ time slots as follows. In the first time slot, all agents will pull arm 1. In subsequent time slots $t \in [2, N-1]$, all those agents that have never been matched in time slots $[1, t-1]$ will pull arm indexed t. Agents that were matched to some $t' \in [1, t-1]$, will play arm t' (the index of the first match) at time t. The estimated rank of an agent is the first time slot when it was matched. If an agent is unmatched in the first $N-1$ time slots, its rank is N. One can observe using an inductive reasoning that, the estimated rank of an agent is equal to its true rank. Thus, all agents are aware of their rank after this phase. The pseudo-code `RANK-ESTIMATION( )` is given in Algorithm 2 in Appendix A.

• **Phase i.** We now describe the algorithm by fixing a particular phase $i \in \mathbb{N}$ and a an agent with rank j. The phase i starts at time $S_i := (2^{i-1} + (i-1)(N-1)K + N - 1)$. It is divided into two blocks: the *first block* $[S_i, S_i + 2^{i-1} - 1]$ is the regret-minimization block lasting 2^{i-1} time-slots and the *second block* is $[S_i + 2^{i-1} - 1, S_{i+1} - 1]$ is the communication block lasting $(N-1)K$ time slots. At the beginning of phase $i \in \mathbb{N}$, associated with rank $j \in [N]$ is an active set of arms $\mathcal{A}_i^{(j)} \subseteq [K]$ with cardinality $|\mathcal{A}_i^{(j)}| \geq K + 1 - j$. (Observe that $\mathcal{A}_i^{(1)} = [K]$). In the beginning of phase 1, we initialize $\mathcal{A}_1^{(j)} = [K]$ for all $j \in [N]$.

⋄ **Regret-Minimization (RM) block.** In the RM block of phase $i \in \mathbb{N}$, an agent with rank j, will play from among the arms in $\mathcal{A}_i^{(j)}$ according to the standard UCB-α rule [2], where $\alpha \geq 2$ is a hyper-parameter. Thus, the arm played (but not necessarily matched) at time t in the RM block of phase i is $I^{(j)}(t) \in \arg\max_{k \in \mathcal{A}_i^{(j)}} \left(\widehat{\mu}_k^{(j)}(t-1) + \sqrt{\frac{2\alpha \ln(t)}{N_k^{(j)}(t)}} \right)$. Ties are broken arbitrarily.

At the end of the RM block of phase i, agent with rank j, creates an estimate of its best arm denoted by $\mathcal{O}_i^{(j)} \in \mathcal{A}_i^{(j)}$, as the arm that it matched with the most number of times in the RM block of phase i. Formally, $\mathcal{O}_i^{(j)} \in \arg\max_{k \in \mathcal{A}_i^{(j)}} (N_k^{(j)}[i] - N_k^{(j)}[i-1])$, where, for any agent $j \in [N]$, phase $i \in \mathbb{N}$ and arm $k \in [K]$, $N_k^{(j)}[i]$ is the number of times arm k was matched to agent j in all the RM blocks up to and including phase i, with the convention $N_k^{(j)}[-1] = 0$. Ties are broken arbitrarily.

$\diamond$ **Communication block.** In the communication block of phase $i \in \mathbb{N}$, agent with rank $j \in [N]$, communicates its estimated best arm $\mathcal{O}_i^{(j)}$ to every other agent with ranks $(j+1)$ through N. This is achieved by arranging arm collisions in a specific way. The communication block, which is of duration $(N-1)K$ time slots, is further sub-divided into $N-1$ sub-blocks, each of K contiguous time slots. In the sub-block $l \in [N-1]$ of the communication block, *all agents with ranks* $(l+1)$ *to rank* N, will pull all the K arms, exactly once in a round robin fashion starting from arm 1. Whereas, in the l-th sub-block of the communication block in phase i, every agent with rank $l' \in [1, l]$, will play arm $\mathcal{O}_i^{(l')}$ in all the K time slots.

For agent ranked $j \geq 2$, denote by $\mathcal{D}_i^{(j)} \subseteq [K]$, the set of arms with which agent j collides in the $j-1$th sub-block of the communication block in phase i. Observe from the communication protocol that $\mathcal{D}_i^{(j)} := \{\mathcal{O}_i^{(1)}, \cdots, \mathcal{O}_i^{(j-1)}\}$. The pseudo-code of the communication block, namely `DOMINATED-ARM-DETECTION`(), is provided in Algorithm 3 in Appendix A.

$\diamond$ **Update active arms.** At the end of the communication block of a phase $i \in \mathbb{N}$, every agent $j \geq 2$ will update its active set of arms for phase $i+1$, by deleting the arms $\mathcal{D}_i^{(j)}$, i.e. $\mathcal{A}_{i+1}^{(j)} := [K] \setminus \mathcal{D}_i^{(j)}$. Agent ranked 1 will have all the K arms active, ($\mathcal{A}_{i+1}^{(1)} = [K]$).

Algorithm 1 UCB with Decentralized Dominant-arm Deletion (UCB-D3) (at Agent $j \in [N]$)

$\texttt{Rank}^{(j)} \leftarrow$ `RANK-ESTIMATION()` $\triangleright$ First $N-1$ arm pulls to estimate rank using Algorithm 2
$\mathcal{C}_0 = \emptyset$ $\triangleright$ Set of arms blocked in the beginning of phase 1
for $i \in \{1, 2, \cdots\}$ **do** $\triangleright$ For each Phase i
  $\mathcal{A}_i^{(j)} \leftarrow [K] \setminus \mathcal{C}_{i-1}$ $\triangleright$ Set of Active arms in phase i
  for $S_i \leq t \leq S_i + 2^{i-1}$ **do** $\triangleright$ The first 2^{i-1} times (RM block) of phase i
    Play an arm $I^{(j)}(t) \in \arg\max_{k \in \mathcal{A}_i^{(j)}} \left(\widehat{\mu}_k^{(j)}(t-1) + \sqrt{\frac{2\alpha \log(t)}{N_k^{(j)}(t-1)}} \right)$
  end for
  $\mathcal{O}_i^{(j)} \leftarrow$ the most matched arm from $\mathcal{A}_i^{(j)}$ in the first 2^{i-1} time slots of phase i.
  $\mathcal{C}_i \leftarrow$ `DOMINATED-ARM-DETECTION(`$\mathcal{O}_i^{(j)}, i,$ $\texttt{Rank}^{(j)}$`)` $\triangleright$ The last $(N-1)K$ times of phase i given in Algorithm 3
end for

4 Regret Upper Bound for UCB-D3

We first set some necessary notations for stating the regret bounds.

Dominated arms. For each agent $j \in [N]$, denote by the set of dominated arms for agent j as $\mathcal{D}_*^{(j)} := \{k_*^{(1)}, \cdots, k_*^{(j-1)}\}$ and the non-dominated arm as $\mathcal{A}_*^{(j)} := [K] \setminus \{k_*^{(1)}, \cdots, k_*^{(j-1)}\}$, where $k_*^{(j)}$ is the stable matched arm for agent j.

Gaps. For any agent $j \in [N]$ and arm $k \neq k_*^{(j)}$, denote by $\Delta_k^{(j)} = (\mu_{jk_*^{(j)}} - \mu_{jk})$ the gap in reward of the k-th arm and the stable matching arm $k_*^{(j)}$. Thus, by definition of $k_*^{(j)}$ and by the uniqueness assumption of arm-means, for all non dominated arms $k \in \mathcal{A}_*^{(j)} \setminus \{k_*^{(j)}\}$, we have $\Delta_k^{(j)} > 0$.

Minimum gap. Let us denote by $\Delta := \min_{j\in[N]} \min_{k\in\mathcal{A}_*^{(j+1)}} \Delta_k^{(j)}$ the smallest gap of non-dominated arms across all agents. Since arm-means are unique for all agents, $\Delta > 0$.

Threshold phase. We define $i^* := \min\left\{i \in \mathbb{N} : \frac{20NK\alpha i}{\Delta^2} \leq 2^{i-1}\right\}$. It is easy to verify that $2^{i^*} := O\left(\frac{NK}{\Delta^2}\log\left(\frac{NK}{\Delta^2}\right)\right)$. This is the phase where the minimum gap in the system can be learned reliably.

Theorem 1. *Suppose every agent runs the distributed matching bandit algorithm given in Algorithm 1 with UCB parameter $\alpha \geq 2$, then the regret of any agent with rank j satisfies*

$$\mathbb{E}[R_T^{(j)}] \leq \underbrace{\sum_{i=1}^{j-1}\sum_{k\in\mathcal{A}_*^{(j)}}\left(\frac{4\alpha\log(T)}{(\Delta_k^{(i)})^2}+1\right)}_{\text{Regret due to Collisions}} + \underbrace{\sum_{k\in\mathcal{A}_*^{(j)}\setminus k_*^{(j)}}\left(\frac{4\alpha\log(T)}{\Delta_k^{(j)}}\right)}_{\text{Regret due to sub-optimal non-dominated arm matches}}$$
$$+ \underbrace{(N-1)K\log_2(T)}_{\text{Regret due to communications}} + N + j(2^{i^*} + i^* + 4(NK)^2 + 3NK) + \frac{2NK}{T^{2\alpha-4}}. \quad (1)$$

The following corollary highlights the dependencies of the regret on different model parameters.

Corollary 2. *Suppose every agent runs the distributed matching bandit algorithm with parameter $\alpha \geq 2$, then the regret of any agent with rank j after T time steps satisfies*

$$\mathbb{E}[R_T^{(j)}] \leq 4\alpha\log(T)\left(\frac{(j-1)(K+1-j)}{\Delta^2} + \frac{[K-1-j]_+}{\Delta}\right) + O\left((NK)^2 + \frac{NK}{\Delta^2}\log\left(\frac{NK}{\Delta^2}\right)\right),$$

where for any $x \in \mathbb{R}$, $[x]_+ := \max(x, 0)$.

Thus, we see from the above corollary, that agent 1, the best ranked agent, will experience no collisions, and thus has a regret scaling as $O(\log(T)/\Delta)$. For any other agent ranked $2 \leq j \leq N$, the regret scales as $O\left((K+1-j)(j-1)/\Delta^2\log(T)\right) + O\left([K-1-j]_+/\Delta\log(T)\right)$. However, this is only an *upper bound* on regret. In particular, under certain circumstances the regret under our algorithm can be negative —if an agent is matched to an arm with higher mean compared to the mean of its stable match partner arm. Moreover, we observe in simulations (Figure 2 and Appendix 7) that, UCB-D3 outperforms the only prior decentralized algorithm, the ETC from [24] and is comparable to a centralized UCB algorithm of [24].

Incentive Compatability - We further discuss in Appendix C in Proposition 16, that our algorithm satisfies certain *incentive compatible properties.* Roughly speaking, this proposition guarantees that even if an agent unilaterally deviates from playing the UCB-D3 algorithm, it must still incur $O(\log(T)/\Delta^2)$ regret.

Proof sketch of Theorem 1 Using the rank estimation protocol each agent learns its own rank correctly in a decentralized way. We may assume agent j knows it is ranked j henceforth. As agent 1 experiences no collisions, we show that for all phases after a random finite number of phases

Figure 1: This setup consists of 5 agents and 5 arms with arm-means chosen i.i.d. uniform in $[0, 1]$. Over 100 parallel runs with 13 phases each, this heatmap counts the arm communicated by each agent at the end of each phase across runs. The color intensity increases with the number of runs where the arm is communicated. This figure demonstrates the freezing of dominated-arms empirically; all agents communicate its stable match (depicted as a contiguous black strip) partner arm after a certain phase number.

$\tau^{(1)}$, agent 1 identifies the best arm in the stable match (see also Figure 1). Thus, from phase $\tau^{(1)}$ on-wards, agent 2 will eliminate the correct arm, arm $k_*^{(1)}$. Moreover, as agent 1 is playing UCB algorithm, the number of times it plays a sub-optimal arm is small and thus, the total number of collisions experienced by agent 2 is small. Subsequently, we show that for all phases after a random phase $\tau^{(2)} \geq \tau^{(1)}$, agent 2 always identifies its correct arm $k_*^{(2)}$ as the best arm. Similarly, we show that for all $j \in [2, N]$, for all phases after a random phase $\tau^{(j)}$, all agents $j' \leq j$ always identifies the set of dominated arms $\mathcal{D}_*^{(j)}$ (hence the non-dominated arms $\mathcal{A}_*^{(j)}$), correctly by eliminating the most played arm of the higher ranked agents in the previous phase. Furthermore, from $\mathcal{A}_*^{(j)}$ it identifies its stable match $k_*^{(j')}$ as its best arm due to UCB dynamics, and thus, regret of all agents ranked j or lower is well-behaved (see also Figure 1). The regret bound is proved by establishing upper bounds on the expectations of $\tau^{(j)}$, $\forall j \in [K]$. We do so by establishing high probability upper bound on the number of times a sub-optimal arm is played by the UCB algorithm, even though the set of active arms varies across phases for each agent (except agent 1) (Lemma 10 in appendix).

5 Instance Dependent Regret Lower Bound

We provide a regret lower bound by adapting approach ([2, 23]) to our multi-agent setup. Let $R_T(\boldsymbol{\nu}, \pi)$ denote the cumulative (sum over all agents and time horizon) expected regret of a policy π on the instance with arm distributions $\nu = \{\nu_{jk} : j \in [N], k \in [K]\}$ for a horizon of length T. Also, denote by $\mathcal{P}$ the set of all probability distributions with bounded support $[0, 1]$.[6] We define $D_{\inf}(\nu, x, \mathcal{P}) = \inf_{\nu' \in \mathcal{P}}\{D(\nu, \nu') : \mu(\nu') > x\}$ for any distribution $\nu \in \mathcal{P}$. Here, $\mu : \mathcal{P} \to \mathbb{R}$ is the operator mapping a distribution in $\mathcal{P}$ to its mean and $D(\cdot, \cdot)$ is the KL divergence.

Definition 3 (Uniformly Consistent Policies)**.** *A policy* π *is* uniformly consistent *if and only if for all* $\boldsymbol{\nu} \in \mathcal{P}$*, all* $\alpha \in (0, 1)$*, the regret* $\limsup_{T\to\infty} \frac{R_T(\boldsymbol{\nu},\pi)}{T^\alpha} = 0$.

This notion is used for stochastic bandit lower bounds to eliminate tuning a policy to the current instance while admitting large regret in other instances (c.f. [2, 23]).

Optimally stable bandits. Our regret lower bounds hold over a sub-class of bandits where the stable matching is optimal. Let us consider the class of bandit instances where dominated arms

[6] Please refer to the supplementary material for a formal definition of the policy and the environment.

are bad, i.e. for any instance $\boldsymbol{\nu}$ in this class, for all agents $j \in [N]$, $\mu_{jk} < \mu_{jk_*^{(j)}}$ for all arms $k \in [K] \setminus \{k_*^{(j)}\}$. We call this class of instances Optimally Stable Bandits (OSB), as each agent is matched with its optimal arm in the stable matching. Let for all $j \in [N]$, $\Delta_{\min}^{(j)} = \min_{k \neq k_*^{(j)}} \Delta_k^{(j)}$, which is always non-negative for an OSB instance.

Lemma 4 (Regret Decomposition for OSB)**.** *For a OSB instance $\boldsymbol{\nu} = \{\nu_{jk} : j \in [N], k \in [K]\}$, and any uniformly consistent policy π, agent $j \in [N]$ the following holds*

$$R_T^{(j)}(\boldsymbol{\nu}, \pi) \geq \max\left\{\sum_{j'=1}^{j-1} \Delta_{\min}^{(j)} \mathbb{E}_{\nu,\pi}[N_{k_*^{(j)}}^{(j')}(T)], \sum_{k \notin \mathcal{A}_*^{(j)} \setminus k_*^{(j)}} \Delta_k^{(j)} \mathbb{E}_{\nu,\pi}[N_k^{(j)}(T)]\right\}.$$

In a OSB instance, if an agent j's stable match partner $k_*^{(j)}$ gets matched to an agent ranked higher than j, then the next bext case is agent j gets matched to its second best (among all K arms). Even though this may happen rarely, it suffices to bound the first term in the above expression. Similar observations were utilized in providing a minimax (not instance dependent) lower bounds in [24].

Theorem 5. *For any agent $j \in [N]$, under any decentralized universally consistent algorithm π on a OSB instance $\boldsymbol{\nu}$ satisfies*

$$\liminf_{T \to \infty} \frac{R_T^{(j)}(\boldsymbol{\nu}, \pi)}{\log T} \geq \max\left\{\sum_{j'=1}^{j-1} \frac{\Delta_{\min}^{(j)}}{D_{\inf}(\nu_{j'k_*^{(j)}}, \mu_{j'k_*^{(j')}}, \mathcal{P})}, \sum_{k \notin \mathcal{A}_j \setminus k_*^{(j)}} \frac{\Delta_k^{(j)}}{D_{\inf}(\nu_{jk}, \mu_{jk_*^{(j)}}, \mathcal{P})}\right\}$$

In the following corollary, we show that the dependence of $\log T/\Delta^2$ in the regret upper bound (Theorem 1) is correct up to an $O(K)$. A detailed discussion on arm-gap dependence is provided in Appendix D. For all agents other than the best ranked agent, the regret scales as $\Theta\left(\log(T)/\Delta^2\right)$ in our model.

Corollary 6. *There exists a OSB bandit instance with Bernoulli rewards, where the regert of agent j is lower bounded as $\Omega\left(\max\left\{\frac{(j-1)\log T}{\Delta^2}, \frac{K \log T}{\Delta}\right\}\right)$.*

As the reward is heterogeneous the highest ranked agents with sub-optimality gap Δ is forced to explore $\Omega(\log(T)/\Delta^2)$ time each. Whereas, all other lower ranked agents are forced to compromise a lot whenever it's sub-optimality gap is large ($\Delta_{\min}^{(j)} = \Omega(1)$). The proof (in Appendix D) is built on the above observation.

6 Comparison with Regret Bounds for Related Models

The regret bound in our problem is in contrast with the performance of a related and a widely studied model known as the multi-agent *colliding bandits model* [7],[31] where, if two or more agents pull the same arm, then all agents are blocked. The regret in this case is measured with respect to an optimal allocation of agents to arms by an oracle that knows all the arm means. The per-agent regret for the colliding bandit model, for both the centralized and decentralized algorithms scale as $O\left(\frac{\log(T)}{\Delta}\right)$. The gap in performance between our model and the colliding bandits model arises due to the *asymmetric collisions*; when multiple agents choose the same arm, not all of them experience a collision in our model. Thus, agents cannot infer if its actions cause a collision to other higher

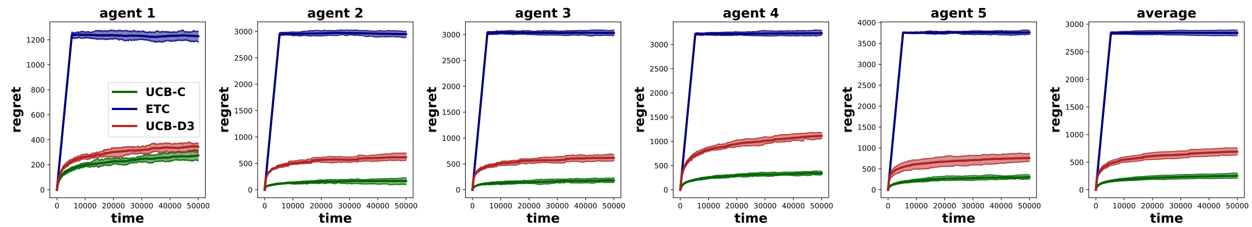


Figure 2: Simulations with 5 agents, and 5 arms. The arm-means for sub-optimal arms for each agent are chosen i.i.d. uniformly over $[0, 0.8]$, while the arm-mean of $i \in [5]$ for agent i was set to 0.9. The rewards are binary. The plots are averaged over 100 runs with 95% confidence intervals. Our algorithm outperforms ETC in [24] (with $H = 1056$), and is qualitatively similar to the centralized UCB in [24]. We provide additional simulation results in Appendix 7.

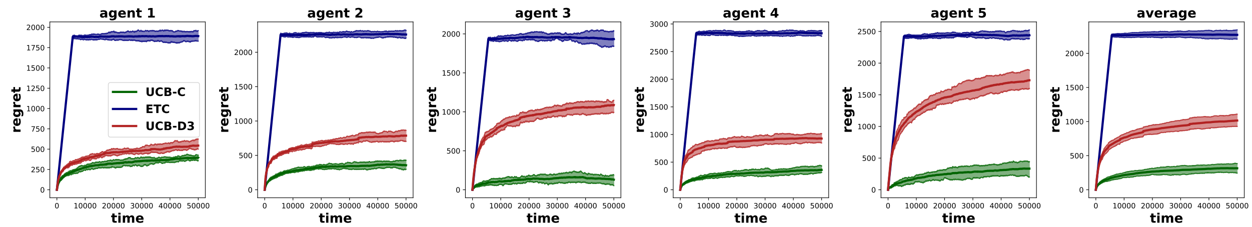
ranked agents and hence the regret must scales as $\Omega\left(\frac{\log(T)}{\Delta^2}\right)$. In contrast, the colliding bandit model is symmetric; if multiple agents pull the same arm simultaneously, then all of them get blocked. Thus, the agents can coordinate in a decentralized way [31] to eliminate blocking and obtain a regret of $O\left(\frac{\log(T)}{\Delta}\right)$.

7 Simulations

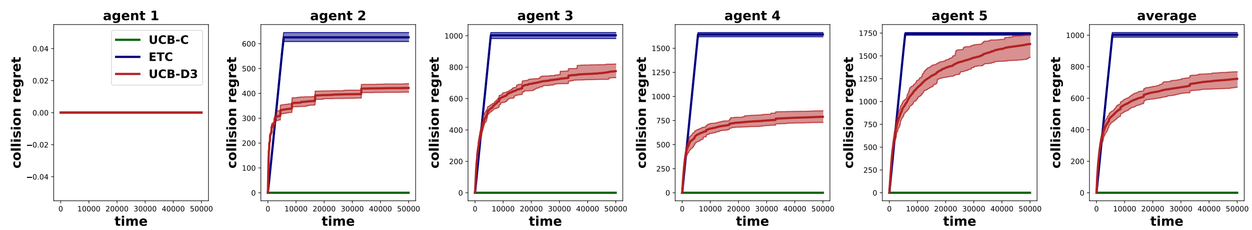
In this section, we conduct further simulations to understand our algorithm and compare it to both, the prior decentralized algorithm ETC and the centralized version of the UCB algorithm.

Simulation Setup We consider 4 random systems to perform simulations to further understand the performance of our algorithm and compare it to both the decentralized ETC and the centralized UCB from [24]. The first two systems are the OSB systems consisting of 10 agents and 10 arms (Figure 4) and 10 agents and 15 arms (Figure 5). In both these systems, a random permutation σ was first chosen and the arm-mean of arm $\sigma(i)$ for agent i was set to 0.9. All other arm-means was chosen randomly and uniformly in $[0, 0.8]$. We then consider two non OSB systems with 5 agents and 7 arms (Figure 3) and 10 agents and 15 arms (Figure 6). In these two systems, every agent j, uniformly spaces the arm-means between 0.1 and 0.9, with each agent having a randomly permutations over the arms to arrange the arm-means. All plots are averaged over 30 trials with confidence intervals of 95%. For brevity, Figures 4,5 and 6 are in the Appendix.

Comparison with other Algorithms - In Figures 4a, 5a, 3a and 6a, we plot the regret of the 10 agents in the two systems, for the three algorithms, with a 95% confidence intervals. We observe that UCB-D3 outperforms ETC and is slightly poorer compared to the centralized UCB algorithm. The centralized UCB algorithm is one in which the central arbitrer matches agents to arms in each round and thus, no agent ever experiences a collision. Thus, the regret in the centralized algorithm is expected to be lower as seen in Figures 2, 4a, 5a, 3a and 6a. This is further demonstrated in Figures 4b, 5b, 3b and 6b where we plot the regret incurred by all algorithms only on account of collisions. As can be seen in these Figures, the collisions incurred in our algorithm are much lower compared to the ETC based decentralized algorithm, thereby incurring lower regret compared to ETC. Although for a few high ranked agents, ETC has lower collisions in Figure 5b, the overall regret of agents is lower with UCB-D3 algorithm as opposed to ETC. The deletion of dominant



(a) Regret plots of all agents



(b) A plot showing the cumulative regret only due to collisions. The centralized UCB ensures that agents never collide and thus do not lose out on regret.

(c) Arms recommended by the agents across phases over different runs of the algorithm.

Figure 3: Simulations on a system with 5 agents and 7 arms. For each agent $i \in [5]$, a permutation over the arms σ_i was chosen, and the arm-means are equally spaced among the 7 arms from 0.1 to 0.9 in the increasing order of permutation. This is thus not a OSB instance. The rewards are binary. The value of $H = 801$ was used for ETC.

arms plays a key role as explained in Section 3.1, which enables our algorithm to have reduced collisions and thus lower regret.

Equilibrium Freezing of UCB-D3 - In Figures 4c, 5c, 3c and 6c, we plot a 'heatmap' of the arms recommended by the agents over the 13 phases. The darker the shade, the higher the frequency (over the different simulation runs), that a particular agent recommended a particular arm in a particular phase. We observe from Figure 5c that after a random phase, all agent always recommend their stable match partner arm. Moreover, the time for an agent to settle into the 'equilibrium' of always recommending their estimated stable match partner arm is larger for lower ranked agents. Nevertheless, Figure 5c shows that after a random time, the agents delete their dominated arms thereby "freezing the system into an equilibrium".

8 Related Work

MAB are widely studied owing to a multitude of applications ([23],[9]). In recent times, as the scale of applications increases, multi-agent MAB problems have come into focus. The paper [24] is the closest to ours, as it introduced the matching bandit model in more generality. However, the algorithms in that paper were either *(i)* centralized and required agents at all points of time to submit a ranked list of arms to a centralized scheduler (platform) or *(ii)* decentralized but needed information on arm-gaps[7]. For the important special case with uniform valuation, we give a decentralized algorithm (and a lower bound), that does not need knowledge of arm-gaps or time horizon. Thus, we partially resolve an open question in [24] on decentralized algorithms for matching bandits.

The literature on multi-agent MAB can be classified into two —*competitive* where different agents compete for limited resources (as in this paper) or *collaborative*, where agents jointly accomplish a shared objective. The canonical model of competitive multi-agent bandits is one wherein if multiple agents play the same arm, they all are blocked and receive no reward (colliding bandit model) [29, 1, 3, 20, 6, 7, 26, 18]. Such models are motivated from applications in wireless networks [4]. However the symmetry in the problem, where if multiple agents choose the same arm, they are all blocked, are crucially used in all decentralized algorithms for that model and are hence not applicable directly to our setup.

The collaborative models, consists of settings where if multiple agents play the same arm simultaneously, then they all receive independent rewards [30, 12, 11, 19, 25, 21]. Such models have primarily been motivated by applications such as internet advertising [13]. However, algorithms there rely on agents collecting independent samples for arms, which are not applicable to our setting.

9 Conclusions

We considered the heterogeneous multi agent matching bandit problem, when there is a global unknown ranking among the agents (uniform valuation). We proposed UCB-D3 a novel decentralized algorithm, which proceeds in phases, and in each phase, an agent 'eliminates' arms it will likely collide. The main insight from our algorithm was that if agents delete their dominated arms (arm optimal for higher ranked agents), then they will incur low regret. We show, both by analyzing our algorithm and by exhibiting a lower bound that the regret of any agent with rank 2 and above decays as $O\left(\frac{\log(T)}{\Delta^2}\right)$. An interesting avenue for future work is to devise decentralized algorithms to the general matching bandit problem without global agent rankings.

References

[1] Pragnya Alatur, Kfir Y Levy, and Andreas Krause. Multi-player bandits: The adversarial case. *Journal of Machine Learning Research*, 21(77):1–23, 2020.

[2] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

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

[7] The regret bounds hold only if certain input parameters are a function of the arm-gap

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

[5] Lilian Besson and Emilie Kaufmann. Multi-player bandits revisited. *arXiv preprint arXiv:1711.02317*, 2017.

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

[7] Etienne Boursier and Vianney Perchet. Sic-mmab: synchronisation involves communication in multiplayer multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 12048–12057, 2019.

[8] Etienne Boursier and Vianney Perchet. Selfish robustness and equilibria in multi-player bandits. *arXiv preprint arXiv:2002.01197*, 2020.

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

[10] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412(19):1832–1852, 2011.

[11] Swapna Buccapatnam, Jian Tan, and Li Zhang. Information sharing in distributed stochastic bandits. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 2605–2613. IEEE, 2015.

[12] Mithun Chakraborty, Kai Yee Phoebe Chua, Sanmay Das, and Brendan Juba. Coordinated versus decentralized exploration in multi-agent multi-armed bandits. In *IJCAI*, pages 164–170, 2017.

[13] Ronshee Chawla, Abishek Sankararaman, Ayalvadi Ganesh, and Sanjay Shakkottai. The gossiping insert-eliminate algorithm for multi-agent bandits. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 3471–3481. PMLR, 2020.

[14] Simon Clark. The uniqueness of stable matchings. *The BE Journal of Theoretical Economics*, 6(1), 2006.

[15] Sanmay Das and Emir Kamenica. Two-sided bandits and the dating market. In *IJCAI*, volume 5, page 19, 2005.

[16] David Gale and Lloyd S Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.

[17] Ramesh Johari, Vijay Kamble, and Yash Kanoria. Matching while learning. *arXiv preprint arXiv:1603.04549*, 2016.

[18] Dileep Kalathil, Naumaan Nayyar, and Rahul Jain. Decentralized learning for multiplayer multiarmed bandits. *IEEE Transactions on Information Theory*, 60(4):2331–2345, 2014.

[19] Ravi Kumar Kolla, Krishna Jagannathan, and Aditya Gopalan. Collaborative learning of stochastic bandits over a social network. *IEEE/ACM Transactions on Networking*, 26(4):1782–1795, 2018.

[20] Lifeng Lai, Hai Jiang, and H Vincent 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. IEEE, 2008.

[21] Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. Distributed cooperative decision-making in multiarmed bandits: Frequentist and bayesian algorithms. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 167–172. IEEE, 2016.

[22] Stefan Larsson. Law, society and digital platforms: Normative aspects of large-scale data-driven tech companies. In *The RCSL-SDJ Lisbon Meeting 2018" Law and Citizenship Beyond The States"*, 2018.

[23] Tor Lattimore and Csaba Szepesvári. Bandit algorithms. *preprint*, page 28, 2018.

[24] Lydia T. Liu, Horia Mania, and Michael Jordan. Competing bandits in matching markets. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 1618–1628. PMLR, 2020.

[25] David Martínez-Rubio, Varun Kanade, and Patrick Rebeschini. Decentralized cooperative stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 4531–4542, 2019.

[26] Abbas Mehrabian, Etienne Boursier, Emilie Kaufmann, and Vianney Perchet. 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*, volume 108 of *Proceedings of Machine Learning Research*, pages 1211–1221. PMLR, 2020.

[27] Roger B Myerson. *Game theory*. Harvard university press, 2013.

[28] Alex Rees-Jones and Samuel Skowronek. An experimental investigation of preference misrepresentation in the residency match. *Proceedings of the National Academy of Sciences*, 115(45):11471–11476, 2018.

[29] Jonathan Rosenski, Ohad Shamir, and Liran Szlak. Multi-player bandits–a musical chairs approach. In *International Conference on Machine Learning*, pages 155–163, 2016.

[30] Abishek Sankararaman, Ayalvadi Ganesh, and Sanjay Shakkottai. Social learning in multi agent multi armed bandits. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3):1–35, 2019.

[31] Po-An Wang, Alexandre Proutiere, Kaito Ariu, Yassir Jedra, and Alessio Russo. Optimal algorithms for multiplayer multi-armed bandits. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 4120–4129. PMLR, 2020.

A Sub-Routines used in Algorithm 1

In this section, we provide the pseudo code for the two sub-routines needed to describe Algorithm 1. These are sub-routines related to initial rank estimation and the communication protocol followed by agents.

Algorithm 2 RANK-ESTIMATION (at agent j)

Initialization: `Rank`$\leftarrow N$, `Flag`$\leftarrow$ **FALSE**
for $1 \leq t \leq N-1$ **do** ▷ Rank Estimation
  if $t == 1$ **OR** `Flag`$==$ **False then**
    $I_j(t) = t$ ▷ Play arm t at time t
    if Matched at time t, i.e., $M_t(t) = j$ **then**
      `Rank`$\leftarrow t$, `Flag`$\leftarrow$ **TRUE**
    end if
  else
    $I_j(t) =$`Rank`
  end if
end for
return `Rank`

Algorithm 3 DOMINATED-ARM-DETECTION ($\mathcal{O}, i$, `Rank`) (at agent j)

Input $\mathcal{O} \in [K]$ - Arm to communicate, $i \in \mathbb{N}$ - the phase and `Rank` - the rank of the agent
$\mathcal{C} \leftarrow \emptyset$
$\widehat{S}_i \leftarrow S_i + 2^{i-1} + 1$
for $\widehat{S}_i \leq t < S_{i+1}$ **do** ▷ Communication block of phase i
  if $t - \widehat{S}_i <$ (`Rank` $- 1) * K$ **then**
    $I^{(j)}(t) = ((t - \widehat{S}_i) \mod K) + 1$ ▷ Play arms in round robin
    if Collision Occurs **AND** $t - \widehat{S}_i >$ (`Rank`$-2) * K$ **then**
    ▷ Collision in (`Rank`-1)th sub-block
      $\mathcal{C} \leftarrow \mathcal{C} \cup \{I^{(j)}(t)\}$ ▷ Update set of arms to delete
    end if
  else
    $I^{(j)}(t) = \mathcal{O}$ ▷ Play the most matched arm, which is input to this sub-routine
  end if
end for
return $\mathcal{C}$

B Analysis of the Algorithm and Proof of Theorem 1

B.1 Notation and Definitions needed for the Proof

In order to implement the proof, we specify certain notations and definitions. For every $i \in \mathbb{N}$, denote by $S_i := N + (2^{i-1} - 1) + (i-1)NK$, to be the first time slot in the regret minimization block of phase i. For any phase $i \in \mathbb{N}$, agent $j \in [N]$ and arm $k \in [K]$, denote by $\widetilde{N}_i^{(j)}[k]$ to be the number of times agent j was matched to arm k in phase i. Recall the notation that for all agents $j \in [N]$, its stable match partner arm was denoted as $k_*^{(j)} \in [K]$. Similarly the set of dominated arms for any agent $j \in \{2, \cdots, N\}$ we defined as $\mathcal{D}_j^* := \{k_*^{(1)}, \cdots, k_*^{(j-1)}\}$. Recall that we had set $\mathcal{A}_*^{(j)} := [K] \setminus \mathcal{D}_*^{(j)}$.

Our first definition is whether a given phase is good for a particular agent or not. We call a phase $i \in \mathbb{N}$ **Good for Agent** j if

1. $\mathcal{A}_i^{(j)} = \mathcal{A}_j^*$, i.e., $\mathcal{A}_i^{(j)} = [K] \setminus \{k_*^{(1)}, \cdots, k_*^{(j-1)}\}$.
2. The number of times each arm $k \in \mathcal{A}_*^{(j)} \setminus \{k_*^{(j)}\}$ is matched to agent j in the regret minimization block of phase i is less than or equal to $\frac{10\alpha i}{(\Delta_k^{(j)})^2}$.
3. The arm that is most matched in the regret minimization block of phase i is $k_*^{(j)}$.

We denote by the event $\chi_i^{(j)}$ to be the indicator random variable, i.e.,

$$\chi_i^{(j)} = \mathbf{1}_{\text{Phase } i \text{ is } \textbf{Good} \text{ for Agent } j}.$$

For every agent $j \in [N]$, denote by the random time $\tau^{(j)}$ to be the first phase index, such that all phases larger than $\tau^{(j)}$ is Good for agent j. Formally,

$$\tau^{(j)} := \inf\left\{ i \in \mathbb{N} : \left(\prod_{l \geq i} \chi_l^{(j)} \right) = 1 \right\},$$
$$\widetilde{\tau}^{(j)} := \max(\tau^{(1)}, \cdots, \tau^{(j)}).$$

Notice that after phase $i \geq \widetilde{\tau}^{(j)}$, for all agents $j' \leq j$, $\mathcal{A}_i^{(j')} = \mathcal{A}_*^{(j')}$. In other words, the set of active arms of all agents ranked j and lower are 'frozen' after phase $\widetilde{\tau}^{(j)}$ to the 'correct' set of arms.

We now, describe certain set of events. For any agent $j \in [N]$ and arm $k \in [K] \setminus \{k_*^{(1)}, \cdots, k_*^{(j)}\}$, denote by the event $\mathcal{E}_k^{(j)}$ as

$$\mathcal{E}_k^{(j)} := \left\{ N_k^{(j)}(T) - N_k^{(j)}(S_{\widetilde{\tau}^{(j)}}) \geq \frac{4\alpha \log(T)}{(\Delta_k^{(j)})^2} + 1 \right\} \tag{2}$$

Denote by the event $\mathcal{E}$ as the union, i.e.,

$$\mathcal{E} := \bigcap_{j=1}^{N} \bigcap_{k \in [K] \setminus \{k_*^{(1)}, \cdots, k_*^{(j)}\}} \mathcal{E}_k^{(j)}. \tag{3}$$

Recall that we had defined Δ to be the smallest arm-gap, namely

$$\Delta := \min_{j \in [N]} \min_{k \in \mathcal{A}_*^{(j)} \setminus \{k_*^{(j)}\}} \Delta_k^{(j)},$$

and that $i^* \in \mathbb{N}$ was defined as

$$i^* := \min\left\{ i \in \mathbb{N} : \frac{20NK\alpha i}{\Delta^2} \leq 2^{i-1} \right\}. \tag{4}$$

B.2 Regret Decomposition

Lemma 7.

$$\mathbb{E}[R_T^{(j)}] \leq \mathbb{E}[S_{\widetilde{\tau}(j)}] + (N-1)K\log_2(T) + \sum_{i=1}^{j-1}\sum_{k \in \mathcal{A}_*^{(j)}} \left(\frac{4\alpha \log(T)}{(\Delta_k^{(i)})^2} + 1 \right) + \sum_{k \in \mathcal{A}_*^{(j+1)}} \left(\frac{4\alpha \log(T)}{\Delta_k^{(j)}} \right) + NKT^{4-2\alpha}$$

Proof. The regret can be decomposed as follows.

$$\begin{aligned}
\mathbb{E}[R_T^{(j)}] &= \mathbb{E}[R_T^{(j)}\mathbf{1}_{\mathcal{E}}] + \mathbb{E}[R_T^{(j)}\mathbf{1}_{\mathcal{E}^c}], \\
&\leq \mathbb{E}[R_T^{(j)}\mathbf{1}_{\mathcal{E}}] + T\mathbb{P}[\mathcal{E}^c], \\
&\leq \mathbb{E}[R_T^{(j)}\mathbf{1}_{\mathcal{E}}] + NKT^{4-2\alpha},
\end{aligned}$$

where the last inequality follows from Corollary 9 in the sequel. It now remains to bound the regret on the event $\mathcal{E}$.

We do so, by noticing that the total regret can be bounded by the regret incurred upto time $S_{\widetilde{\tau}(j)}$ (which we upper bound by 1). Subsequently after time $S_{\widetilde{\tau}(j)}$, agent j incurs a regret only if either *(i)* it experiences a collision, namely any agent $i \in \{1, \cdots, j-1\}$ and agent j pull the same arm or *(ii)* agent j is matched to a sub-optimal arm. However, since we are on the event $\mathcal{E}$, we know that there is an upper bound on both these quantities. Namely, we have the following regret decomposition.

$$\mathbb{E}[R_T^{(j)}\mathbf{1}_{\mathcal{E}}] \leq \mathbb{E}[S_{\widetilde{\tau}(j)}] + \underbrace{\mathbb{E}\left[\sum_{i=1}^{j-1}\sum_{k \in \mathcal{A}_*^{(j)}} \left(N_k^{(i)}(T) - N_k^{(i)}(S_{\widetilde{\tau}(j)})\right)\mathbf{1}_{\mathcal{E}}\right]}_{\text{Regret due to collisions}} + \underbrace{\mathbb{E}\left[\sum_{k \in \mathcal{A}_*^{(j)} \setminus \{k_*^{(j)}\}} \Delta_k^{(j)} \left(N_k^{(j)}(T) - N_k^{(j)}(S_{\widetilde{\tau}(j)})\right)\mathbf{1}_{\mathcal{E}}\right]}_{\text{Regret due to sub-optimal arm-matches}} + \underbrace{(N-1)K\log_2(T)}_{\text{Regret due to Communications}}.$$

This decomposition follows since agent j will not be matched to an arm, if and only if, it collides with another agent $i < j$. Moreover, the total number of times, agent j can collide is upper bounded by the total number of times, any agent ranked less than j is matched to any arm in $\mathcal{A}_*^{(j)}$. Now, from the definition of event $\mathcal{E}$, we know that for all arms $k \in \mathcal{A}_*^{(j)}$, and all agents $i < j$, $\left(N_k^{(i)}(T) - N_k^{(i)}(S_{\widetilde{\tau}(j)})\right) \leq \frac{4\alpha\log(T)}{(\Delta_k^{(i)})^2}$ and for all arms $k \in \mathcal{A}_*^{(j)} \setminus \{k_*^{(j)}\}$, we have $\left(N_k^{(j)}(T) - N_k^{(j)}(S_{\widetilde{\tau}(j)})\right) \leq \frac{4\alpha\log(T)}{(\Delta_k^{(j)})^2}$.

The regret due to communications follows as in a time horizon of T, there are a total of $\lfloor \log_2(T) \rfloor$ phases, and in each phase, the communication block lasts $(N-1)K$ rounds.

□

Lemma 8. *The following bound holds*

$$\mathbb{P}\left[\left(\mathcal{E}_k^{(j)}\right)^{\complement}\right] \leq 2T^{3-2\alpha},$$

where $\mathcal{E}_k^{(j)}$ is defined in Equation (2).

Proof. The proof follows from standard large-deviation results for the UCB algorithm.

$$\begin{aligned}
\mathbb{P}\left[\left(\mathcal{E}_k^{(j)}\right)^{\complement}\right] &\leq \mathbb{P}\left[\bigcup_{t=S_{\widetilde{\tau}(j)}}^{T} I^{(j)}(t) = k, N_k^{(j)}(t) = \frac{4\alpha\log(T)}{(\Delta_k^{(j)})^2}\right], \\
&\leq \sum_{t=1}^{T} \mathbb{P}\left[I^{(j)}(t) = k, N_k^{(j)}(t) = \frac{4\alpha\log(T)}{(\Delta_k^{(j)})^2}, t \geq S_{\widetilde{\tau}(j)}\right], \\
&\stackrel{(a)}{\leq} \sum_{t=1}^{T} 2T^{2(1-\alpha)}, \\
&\leq 2T^{3-2\alpha}.
\end{aligned}$$

Step (a) follows from Theorem 1 in [2]. Note that on the event that $t \geq S_{\widetilde{\tau}(j)}$, arm k is sub-optimal for agent j, and the arm-gap between the best arm for agent j at any time $t \geq S_{\widetilde{\tau}(j)}$ and arm $k \in [K]$ is given by $\Delta_k^{(j)}$. □

Corollary 9. *The following bound holds*

$$\mathbb{P}[\mathcal{E}^{\complement}] \leq 2NKT^{3-2\alpha},$$

where $\mathcal{E}$ was defined in Equation (3).

Proof. The proof follows from Lemma 8 and an union bound over a total of at-most NK events of the form $\mathcal{E}_k^{(j)}$. □

B.3 Bound on Mean and Exponential Moment of $\widetilde{\tau}^{(j)}$

Since for all $i \in \mathbb{N}$, $S_i = N + (2^{i-1} - 1) + (i-1)NK$, it suffices to bound the exponential moment $\mathbb{E}[2^{\widetilde{\tau}^{(j)}}]$ and the mean $\mathbb{E}[\widetilde{\tau}^{(j)}]$ to complete the regret guarantee. In order to do so, we first start by analyzing the probability that a phase is bad for an agent and then use that to bound the exponential moment of $\tau^{(j)}$. We shall now bound the probability that a phase is bad for a particular agent.

Lemma 10. *For any phase $i > i^*$, any agent j and arm $k \in \mathcal{A}_*^{(j)} \setminus \{k_*^{(j)}\}$, we have*

$$\mathbb{P}[\chi_i^{(j)} = 0, i \geq \widetilde{\tau}^{(j-1)}] \leq 2N^2\left(\frac{2}{e^5}\right)^i.$$

Proof. The proof follows from the basic properties of the UCB algorithm, which we can bound as follows. Recall the notation that for any agent $j \in [N]$, phase $i \in \mathbb{N}$ and arm $k \in [K]$, the quantity $N_k^{(j)}[i]$ denotes the number of times agent j was matched to arm k in the regret minimization blocks upto and including phase i. For any phase $i \geq i^*$, agent j and arm $k \in \mathcal{A}_*^{(j)} \setminus \{k_*^{(j)}\}$, we have

$$\mathbb{P}\left[N_k^{(j)}[i] - N_k^{(j)}[i-1] > \left\lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \right\rceil, i \geq \widetilde{\tau}^{(j-1)}\right] \leq \mathbb{P}\left[\bigcup_{t \geq S_{i-1} + \lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \rceil}^{S_i} N_k^{(j)}(t) = \left\lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \right\rceil, I^{(j)}(t) = k, i \geq \widetilde{\tau}^{(j-1)}\right]. \quad (5)$$

Now, by applying an union bound to the RHS, we obtain from the preceding display that

$$\mathbb{P}\left[\bigcup_{t \geq S_{i-1} + \lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \rceil}^{S_i} N_k^{(j)}(t) = \left\lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \right\rceil, I^{(j)}(t) = k, i \geq \widetilde{\tau}^{(j-1)}\right] \leq \sum_{t=S_{i-1}}^{S_i} \mathbb{P}\left[N_k^{(j)}(t) = \left\lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \right\rceil, I^{(j)}(t) = k, i \geq \widetilde{\tau}^{(j-1)}\right]. \quad (6)$$

The classical large-deviaition estimate for UCB from [2] gives that

$$\mathbb{P}\left[N_k^{(j)}(t) = \left\lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \right\rceil, I^{(j)}(t) = k, i \geq \widetilde{\tau}^{(j-1)}\right] \leq 2e^{-5i}, \quad (7)$$

for all times $t \in \{S_{i-1}, \cdots, S_i\}$. We can use that estimate, since on the event that $i \geq \widetilde{\tau}^{(j-1)}$, we have that the set of active arms of agent j in phase i, denoted by $\mathcal{A}_i^{(j)} = \mathcal{A}_*^{(j)}$. Thus, arm $k_*^{(j)}$ is the best arm for agent j in phase i. Now, combining Equations (5),(6),(7), we get

$$\begin{aligned}\mathbb{P}\left[N_k^{(j)}[i] - N_k^{(j)}[i-1] > \left\lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \right\rceil, i \geq \widetilde{\tau}^{(j-1)}\right] &\leq \sum_{t=S_{i-1}}^{S_i} 2e^{-5i}, \\ &\leq 2\left(\frac{2}{e^5}\right)^i. \end{aligned} \quad (8)$$

To conclude the proof, notice the following fact.

Proposition 11. *For every $i \geq i^*$,*

$$\{\chi_i^{(j)} = 0, i \geq \widetilde{\tau}^{(j-1)}\} \subseteq \bigcup_{j'=1}^{j} \bigcup_{k \in \mathcal{A}_*^{(j')} \setminus \{k_*^{(j')}\}} \left\{[N_k^{(j)}[i] - N_k^{(j)}[i-1] \geq \left\lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \right\rceil, i \geq \widetilde{\tau}^{(j-1)}\right\},$$

where i^* is defined in Equation (4).

Proof. It suffices to establish that

$$\bigcap_{j'=1}^{j} \bigcap_{k \in \mathcal{A}_*^{(j')} \setminus \{k_*^{(j')}\}} \left\{ [N_k^{(j)}[i] - N_k^{(j)}[i-1] \leq \left\lceil \frac{10\alpha i}{(\Delta_k^{(j)})^2} \right\rceil, i \geq \widetilde{\tau}^{(j-1)} \right\} \subseteq \{\chi_i^{(j)} = 1, i \geq \widetilde{\tau}^{(j-1)}\}.$$

This follows as, the phase i^* is such that $\frac{10\alpha i}{\Delta^2} NK < 2^{i-1}$. Suppose all events on the LHS hold. Then, agent j is matched at-most $\frac{10\alpha i}{\Delta^2}$ times to any arm $\mathcal{A}_*^{(j)} \setminus \{k_*^{(j)}\}$. Furthermore, as agent j can only collide from agents ranked 1 through to $j-1$. Thus, the total number of collisions experienced by agent j in phase i is at-most $NK \frac{10\alpha i}{\Delta^2}$. However, as the total number of time slots in phase i is at-least $20NK \frac{\alpha i}{\Delta^2}$, it follows that agent j must be matched to arm $k_*^{(j)}$ at-least $NK \frac{10\alpha i}{\Delta^2}$ times. Thus, agent j is matched to arm $k_*^{(j)}$ the most number of times in phase i. □

Thus, from Proposition 11, and applying an union bound using Equation (8), we get

$$\begin{aligned} \mathbb{P}[\chi_i^{(j)} = 0, i \geq \widetilde{\tau}^{(j-1)}] &\leq 2jK \left(\frac{2}{e}\right)^i, \\ &\leq 2NK \left(\frac{2}{e}\right)^i. \end{aligned}$$

□

We use this to now compute the mean of $\widetilde{\tau}^{(j)}$.

Proposition 12. *For every $j \in [N]$, we have*

$$\mathbb{E}[\widetilde{\tau}^{(j)}] \leq j(i^* + 3NK),$$

where i^* is defined in Equation (4).

Proof.

$$\begin{aligned} \mathbb{E}[\widetilde{\tau}^{(j)}] &= \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j)} \geq x], \\ &= \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j)} \geq x, \widetilde{\tau}^{(j-1)} > x] + \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j)} \geq x, \widetilde{\tau}^{(j-1)} \leq x], \\ &= \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j-1)} > x] + \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j)} \geq x, \widetilde{\tau}^{(j-1)} \leq x], \\ &\leq \mathbb{E}[\widetilde{\tau}^{(j-1)}] + i^* + \sum_{x \geq i^*} \mathbb{P}[\widetilde{\tau}^{(j)} \geq x, \widetilde{\tau}^{(j-1)} \leq x], \\ &\stackrel{(a)}{\leq} \mathbb{E}[\widetilde{\tau}^{(j-1)}] + i^* + \sum_{x \geq i^*} 2N^2 \left(\frac{2}{e^5}\right)^x, \\ &\leq \mathbb{E}[\widetilde{\tau}^{(j-1)}] + i^* + 3NK, \\ &\stackrel{(b)}{\leq} j(i^* + 3NK). \end{aligned}$$

Step (a) follows Lemma 10 and step (b) follows from the fact that $\widetilde{\tau}^{(0)} = 0$ almost-surely. □

Similarly, we can compute the exponential moment of $\widetilde{\tau}^{(j)}$.

Proposition 13. *For every $j \in [N]$, we have*

$$\mathbb{E}[2^{\widetilde{\tau}^{(j)}}] \leq 1 + j(2^{i^*} + 4NK),$$

where i^* is defined in Equation (4).

Proof.

$$\begin{aligned}
\mathbb{E}[2^{\widetilde{\tau}^{(j)}}] &= \sum_{x \geq 1} \mathbb{P}[2^{\widetilde{\tau}^{(j)}} \geq x], \\
&= \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j)} \geq \log_2(x)], \\
&= \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j)} \geq \log_2(x), \widetilde{\tau}^{(j-1)} > \log_2(x)] + \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j)} \geq \log_2(x), \widetilde{\tau}^{(j-1)} \leq \log_2(x)], \\
&= \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j-1)} > \log_2(x)] + \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j)} \geq \log_2(x), \widetilde{\tau}^{(j-1)} \leq \log_2(x)], \\
&= \mathbb{E}[2^{\widetilde{\tau}^{(j-1)}}] + \sum_{x \geq 1} \mathbb{P}[\widetilde{\tau}^{(j)} \geq \log_2(x), \widetilde{\tau}^{(j-1)} \leq \log_2(x)], \\
&\overset{(a)}{\leq} \mathbb{E}[2^{\widetilde{\tau}^{(j-1)}}] + 2^{i^*} + \sum_{x \geq 2^{i^*}} 2NK \left(\frac{2}{e^5}\right)^{\log_2 x}, \\
&= \mathbb{E}[2^{\widetilde{\tau}^{(j-1)}}] + 2^{i^*} + \sum_{x \geq 2^{i^*}} 2NKx^{1 - \frac{5}{\ln(2)}}, \\
&\leq \mathbb{E}[2^{\widetilde{\tau}^{(j-1)}}] + 2^{i^*} + 4NK, \\
&\overset{(b)}{\leq} 1 + j(2^{i^*} + 4NK).
\end{aligned}$$

Step (a) follows from Lemma 10 and step (b) follows from the fact that $\widetilde{\tau}^{(0)} = 0$ almost-surely. □

Corollary 14. *For every $j \in [N]$, we have*

$$\begin{aligned}
\mathbb{E}[S_{\widetilde{\tau}^{(j)}}] &\leq N + j(2^{i^*} + 4NK) + jNK(i^* + 3NK), \\
&= N + j(2^{i^*} + i^* + 4(NK)^2 + 3NK),
\end{aligned}$$

where i^* is defined in Equation (4).

Proof. We know that for any $i \in \mathbb{N}$, $S_i := N + (2^{i-1} - 1) + (i-1)NK$. The result then follows from Propositions 12 and 13. □

C Incentive Compatibility and Robustness to Selfish Players

In this section, we discuss robustness properties when any agent deviates from executing the UCB-D3 algorithm in order to maximize the collected reward. We consider a desirable robustness property called ε Nash Equilibrium for multi-agent algorithms, recently proposed in [8]. Roughly, this property guarantees that no agent can significantly increase (by at-most additive ε) its rewards by unilaterally deviating from the UCB-D3; protocol. Although this is a weaker concept compared to the classical Nash-Equilibrium used in the theory of repeated games [27], is nevertheless a useful property for practical algorithms in multi-agent bandits to posses [8].

We set some notations to define this concept. For any agent $j \in [N]$, time horizon T and *algorithm profile* $s := (s_1, \cdots, s_N)$ executed by the N agents respectively, denote by $\text{Rew}_T^{(j)}(s)$ to be the reward collected by agent j in T time slots, when agent $j \in [N]$ executes algorithm s_i. For an algorithm profile s, an algorithm s' and any agent $j \in [N]$, denote by $\text{Rew}_T^{(j)}(s_{-j}, s')$ to be the reward obtained by agent j when it executes algorithm s' and the other agents play according to the profile s.

Definition 15. *An algorithm profile s is an $\varepsilon := (\varepsilon_j)_{j=1}^N$ Nash Equilibrium if, for every agent $j \in [N]$ and algorithm s', $\mathbb{E}[Rew_T^{(i)}(s_{-j}, s')] \leq \mathbb{E}[Rew_T^{(j)}(s)] + \varepsilon_j$.*

Proposition 16. *The UCB-D3 algorithm profile is $\varepsilon : (\varepsilon_j)_{j=1}^N$ stable where, for all $j \in [N]$, $\varepsilon_j = \sum_{l=1}^{j-1} \mathbf{1}_{(\mu_{jl} > \mu_{jk_j^*})} \frac{\mu_{jl}}{\mu^{(l)}} \mathbb{E}[R_T^{(l)}] + \mathbb{E}[R_T^{(j)}]$, where for all $j' \in [N]$, $\mathbb{E}[R_T^{(j')}]$ is given in Equation (1).*

This proposition gives that for agent ranked j, $\varepsilon_j = O\left(j \frac{\mu_{\max}^{(j)}}{\mu_{\min}^{(j)}} \mathbb{E}[R_T^{((j))}]\right)$, where $\mu_{\max}^{(j)}$ ($\mu_{\min}^{(j)}$) is the maximum (minimum) arm-mean for agent j. This establishes that UCB-D3 is approximately incentive compatible, namely, even if an agent deviates from the UCB-D3 algorithm, the possible improvement in reward is $O(\log(T))$.

Proof of Proposition 16. We bound the equilibrium property of UCB-D3; as follows. Observe that agent ranked j will only collide with agents ranked 1 through $j-1$. Now, if all agents 1 through to $j-1$ are all playing arms $k_*^{(1)}, \cdots k_*^{(j-1)}$ respectively (their individual best arms), then the best arm (by definition) for agent j to play will be arm $k_*^{(j)}$. On the other hand, when any agent $j' \leq j-1$ does not play arm $k_*^{(j')}$, the maximum expected reward collected by agent j can be at-most $\max(\mu_{jk_*^{(j')}}, \mu_{jk_*^{(j)}})$. Under the UCB-D3; strategy profile, the expected number of times any agent $j \in [N]$, plays an arm in the set $[K] \setminus \{k_*^{(1)}, \cdots, k_*^{(j)}\}$ is at-most $\frac{1}{\mu^{(j)}} \mathbb{E}[R_T^{(j)}]$, where $\mu^{(j)} := \min_{k \in [K]} \mu_{jk}$ is the smallest arm-gap. Notice that for all agents j, $\mu^{(j)} > 0$, by model assumptions. This then gives us the following decomposition

$$\sup_{s'} \mathbb{E}[\text{Rew}_T^{(j)}(s_{-j}, s')] \leq \sum_{l=1}^{j-1} \mathbf{1}_{\mu_{jl} > \mu_{jk_*^{(j)}}} \frac{\mu_{jl}}{\mu^{(l)}} \mathbb{E}[R_T^{(l)}] + \mu_{jk_*^{(j)}} T, \tag{9}$$

where $\mathbb{E}[R_T^{(l)}]$ is given in Theorem 1. Similarly, from the definition of regret, we have

$$\mathbb{E}[\text{Rew}_T^{(j)}(s)] \geq \mu_{jk_*^{(j)}} T - \mathbb{E}[R_T^{(j)}], \tag{10}$$

where $\mathbb{E}[R_T^{(j)}]$ is given in Theorem 1. Thus, from Equations (9) and (10), we get that

$$\sup_{s'} \left(\mathbb{E}[\text{Rew}_T^{(j)}(s_{-j}, s')] - \mathbb{E}[\text{Rew}_T^{(j)}(s)] \right) \leq \sum_{l=1}^{j-1} \mathbf{1}_{\mu_{jl} > \mu_{jk_*^{(j)}}} \frac{\mu_{jl}}{\mu^{(l)}} \mathbb{E}[R_T^{(l)}] + \mathbb{E}[R_T^{(j)}].$$

$\square$

D Proof of Regret Lower Bound

We will use the following notations throughout the proof of the lower bound.

1. Can assume without loss of generality (W.l.o.g.) that the rank of any agent $i \in [N]$ is i.
2. Any agent related symbol is a superscript. Arm related is a sub-script. Thus, for any time t, the number of times arm $k \in [K]$ is played by agent j is $N_k^{(j)}(t)$. The number of time the agent j is blocked up to time t is given as $C^{(j)}(t)$.
3. Distribution of agent $i \in [N]$ and arm $k \in [K]$ is given by ν_{jk}, which has mean μ_{jk}. W.l.o.g. let us assume $\max_k \mu_{jk} > 0$.
4. The stable match partner of any agent $j \in [N]$ is given by $k_*^{(j)} \in [K]$. The set of dominated arms for the agent j is given as $\mathcal{D}_*^{(j)} = \{k_*^{(j)} : 1 \leq j' \leq j-1\}$, the set of non-dominated arms is given as $\mathcal{A}_*^{(j)} = [K] \setminus \mathcal{D}_*^{(j)}$.
5. For any agent $j \in [N]$, arm $k \in [K]$, $\Delta_k^{(j)} := \mu_{jk_*^{(i)}} - \mu_{jk}$, the arm-gap. This can be negative.

D.1 Divergence Decomposition

We need to setup a few notations for the proof of divergence decomposition lemma. The proof generalizes the framework in Chapter 15 of [23] for the multi-agent framework.[8]

Canonical multi-agent bandit model: We now define the (N-agent, K-arm, T-horizon) bandit models. The canonical bandit model (N-agent, K-arm, T-horizon) lies in a measurable space $\{\Omega, \mathcal{F}\}$. Let $K^{(j)}(t)$ denotes the arm chosen by the j-th agent on time t, and $X^{(j)}(t)$ denotes the rejection or reward obtained from that arm for agent j in round t. We denote the rejection by the symbol $\emptyset$. Therefore, $K^{(j)}(t) \in [K]$, and $X^{(j)}(t) \in [0,1] \cup \{\emptyset\}$ for all $j \in [N]$ and $t \in [T]$. Also, $K^{(j)}(t)$, and $X^{(j)}(t)$ for all $j \in [N]$ and $t \in [T]$ are measurable with respect to $\mathcal{F}$. Let $H(t) = (K^{(j)}(t'), X^{(j)}(t') \forall j \in [N], \forall t' \leq t)$ be the random variable representing the history of actions taken and rewards seen up to and including round t. We have $H(t) \in \mathcal{H}(t) \equiv \Big([K]^N \times ([0,1] \cup \{\emptyset\})^N\Big)^t$. We may set $\Omega \equiv \mathcal{H}(T)$ and the sigma algebra generated by the history as $\mathcal{F} \equiv \sigma(H(T))$.

Environment: The bandit environment is specified by $\nu = (\nu_{jk}, \forall j \in [N], k \in [K])$ where ν_{jk} is the distribution of rewards obtained when arm k is matched to agent j in this environment.

Policy: A policy is a sequence of distribution of possible request to the arms from the agents (which can assimilate any coordination among the agents) conditioned on the past events. More formally, the policy $\boldsymbol{\pi} = \{\boldsymbol{\pi}_t(\cdot) : t \in [T]\}$ where $\boldsymbol{\pi}_t(\cdot) \equiv \{\pi_t(k, j|\cdot) : \forall k \in [K], j \in [N]\}$ is the function that maps the history upto time $t-1$ to the action $K^{(j)}(t), \forall j \in [N]$. Further, $\pi_t(k, j|\cdot) : \mathcal{H}(t-1) \to [0,1]$ denotes the probability, as a function of history $H(t-1)$ of agent j playing arm k.

[8]See, [5] for a related approach for regret lower bound proof in the colliding bandit models [3].

Probability Measure: Each environment $\boldsymbol{\nu}$ and policy π jointly induces a probability distribution over the measurable space $\{\Omega, \mathcal{F}\}$ denoted by $\mathbb{P}_{\nu,\pi}$. Let $\mathbb{E}_{\nu,\pi}$ denote the expectation induced. The density of a particular history up to time T, under an environment $\boldsymbol{\nu}$ and a policy π, can be defined as

$$\begin{aligned}
&d\mathbb{P}_{\nu,\pi}\left(\mathbf{k}(t), \mathbf{x}(t) : t \in [T]\right) \\
&= \prod_{t=1}^{T} \pi_t(\mathbf{k}(t)|h(t-1)) p_\nu(\mathbf{x}(t)|\mathbf{k}(t)) d\lambda(\mathbf{x}(t);\boldsymbol{\nu}) d\rho(\mathbf{k}(t)).
\end{aligned}$$

Here, $\lambda(\mathbf{x};\boldsymbol{\nu}) = \prod_{j=1}^{N} \lambda_j(x^{(j)})$ is the dominating measure over the rewards with $\lambda_j(x^{(j)}) = \delta_\emptyset + \sum_k \nu_{jk}$.[9] Also, $\rho(\mathbf{k})$ is the counting measure on the collective action of the agents.

Lemma 17 (Divergence Decomposition)**.** *For two bandit instances $\boldsymbol{\nu} = \{\nu_{jk} : j \in [N], k \in [K]\}$, and $\boldsymbol{\nu}' = \{\nu_{jk} : j \in [N], k \in [K]\}$, and any admissible policy π the following divergence decomposition is true*

$$D(\mathbb{P}_{\nu,\pi}, \mathbb{P}_{\nu',\pi}) = \sum_{j=1}^{N}\sum_{k=1}^{K} \mathbb{E}_{\nu,\pi}[N_k^{(j)}(T)] D(\nu_{jk}, \nu'_{jk}).$$

Proof. The divergence between two measures, which correspond to two different environments under a policy π, $\mathbb{P}_{\nu,\pi}$ and $\mathbb{P}_{\nu',\pi}$ can be expressed as

$$\begin{aligned}
&D(\mathbb{P}_{\nu,\pi}, \mathbb{P}_{\nu',\pi}) \\
&= \mathbb{E}_{\nu,\pi}\left[\sum_{t=1}^{T} \log\left(\frac{d\mathbb{P}_{\nu,\pi}}{d\mathbb{P}_{\nu',\pi}}\right)\right] \\
&\overset{(i)}{=} \mathbb{E}_{\nu,\pi}\left[\log\left(\frac{p_\nu(\mathbf{x}(t)|\mathbf{k}(t))}{p_{\nu'}(\mathbf{x}(t)|\mathbf{k}(t))}\right)\right] \\
&\overset{(ii)}{=} \mathbb{E}_{\nu,\pi}\left[\sum_{t=1}^{T} \log\left(\frac{\prod_{j:x^{(j)}(t)\neq\emptyset} p_\nu(x^{(j)}(t)|\mathbf{k}(t))}{\prod_{j:x^{(j)}(t)\neq\emptyset} p_{\nu'}(x^{(j)}(t)|\mathbf{k}(t))}\right)\right] \\
&= \mathbb{E}_{\nu,\pi}\left[\sum_{t=1}^{T} \sum_{j:x^{(j)}(t)\neq\emptyset} \mathbb{E}_\nu\left[\log\left(\frac{p_\nu(x^{(j)}(t)|\mathbf{k}(t))}{p_{\nu'}(x^{(j)}(t)|\mathbf{k}(t))}\right) \Bigg| \mathbf{k}(t)\right]\right] \\
&\overset{(iii)}{=} \mathbb{E}_{\nu,\pi}\left[\sum_{t=1}^{T} \sum_{j:x^{(j)}(t)\neq\emptyset} D\left(\nu_{jk^{(j)}(t)}, \nu'_{jk^{(j)}(t)}\right)\right] \\
&= \sum_{j=1}^{N}\sum_{k=1}^{K} \mathbb{E}_{\nu,\pi}\left[\sum_{t=1}^{T} \mathbf{1}_{(k^{(j)}(t)=k, x^{(j)}(t)\neq\emptyset)} D\left(\nu_{jk}, \nu'_{jk}\right)\right] \\
&\overset{(iv)}{=} \sum_{j=1}^{N}\sum_{k=1}^{K} \mathbb{E}_{\nu,\pi}\left[N_k^{(j)}(T)\right] D\left(\nu_{jk}, \nu'_{jk}\right).
\end{aligned}$$

[9] Here $\delta_\emptyset$ is the dirac measure on $\emptyset$ denoting the rejection event. For multiple a pair environments we can define a dominating measure as $\lambda(\mathbf{x};\boldsymbol{\nu}_1,\boldsymbol{\nu}_2) = \sum_{i=1,2}\lambda(\mathbf{x};\boldsymbol{\nu}_i)$. This is used in the proof of Lemma 17.

In the above series equation (i) is true because the density of the policy cancels out for the two different environments. Equation (ii) holds because if for some set of actions $\mathbf{k}(t)$ agent j observes $x^{(j)}(t) = \emptyset$ that indicates agent j is rejected on that round. This is independent of the environment. In particular, we have $p_\nu(x^{(j)}(t)|\mathbf{k}(t)) = p_{\nu'}(x^{(j)}(t)|\mathbf{k}(t))$ if $x^{(j)}(t) = \emptyset$ for any $\mathbf{k}(t)$. In deriving equation (iii) we make use of the definition of divergence. Equation (iv) uses the definition of $N_k^{(j)}(t)$ the total number of times agent j successfully plays arm k up to time T. □

D.2 Proof of Regret Decomposition (Lemma 4)

Proof. We fix any agent $j \in [N]$ for the rest of the proof. We have the expected regret for the agent j, under a policy π and any bandit instance $\boldsymbol{\nu}$ as

$$R_T^{(j)}(\boldsymbol{\nu}, \pi) = \sum_{k=1}^{K} \Delta_k^{(j)} \mathbb{E}_{\nu,\pi}[N_k^{(j)}(T)] + \sum_{k=1}^{K} \mu_{jk_*^{(j)}} \mathbb{E}_{\nu,\pi}[C^{(j)}(T)].$$

This is true as for each collision the agent j obtains $\mu_{jk_*^{(j)}}$ regret (0 reward) in expectation, and for each successful play of arm k it obtains $\Delta_k^{(j)}$ regret. Therefore, a trivial regret lower bound is

$$R_T^{(j)}(\boldsymbol{\nu}, \pi) \geq \sum_{k=1}^{K} \Delta_k^{(j)} \mathbb{E}_{\nu,\pi}[N_k^{(j)}(T)].$$

For an OSB instance, we know that the number of times the agents 1 to $(j-1)$ plays arm $k_*^{(j)}$ successfully, the agent j should either move to a sub-optimal arm (as the arm $k_*^{(j)}$ is the optimal arm for agent j in an OSB instance) or it is blocked. In the best possible scenario, the agent j successfully plays its second best arm, in each of these instances. This holds as $\Delta_{\min}^{(j)} \leq \mu_{jk_*^{(j)}}$ for non-negative rewards. Therefore, the regret from the events when agents 1 to $(j-1)$ plays arm $k_*^{(j)}$ successfully, is lower bounded by

$$R_T^{(j)}(\boldsymbol{\nu}, \pi) \geq \sum_{j'=1}^{j-1} \Delta_{\min}^{(j)} \mathbb{E}_{\nu,\pi}[N_{k_*^{(j)}}^{(j')}(T)].$$

Therefore, the combined regret lower bound is given as

$$R_T^{(j)}(\boldsymbol{\nu}, \pi) \geq \max\left\{ \sum_{k=1}^{K} \Delta_k^{(j)} \mathbb{E}_{\nu,\pi}[N_k^{(j)}(T)], \sum_{j'=1}^{j-1} \Delta_{\min}^{(j)} \mathbb{E}_{\nu,\pi}[N_{k_*^{(j)}}^{(j')}(T)] \right\}$$

□

D.3 Proof of Regret Lower Bound (Theorem 5)

Proof. We consider any instance in the class of OSB $\boldsymbol{\nu}$, universally consistent policy π, agent $j \in [N]$, and arm $k \in [K] \setminus \{k_*^{(j')} : 1 \leq j' \leq j\}$. Let us consider the instance $\boldsymbol{\nu}'$ (which is specific to the j and k pair) where $\nu'_{j'k'} = \nu_{j'k'}$ for all $j' \neq j, k' \neq k$, ν'_{jk} such that $D(\nu_{jk}, \nu'_{jk}) \leq D_{\inf}(\nu_{jk}, \mu_{jk_*^{(j)}}, \mathcal{P}) + \epsilon$ and $\mu'_{jk} \equiv \mu(\nu'_{jk}) > \mu_{jk_*^{(j)}}$ for some $\epsilon > 0$. Note, for $\mu_{jk_*^{(j)}} < 1$ and $\Delta_k^{(j)} > 0$, which holds by

assumption, the distribution ν'_{jk} exists by definition of $D_{\inf}(\cdot)$. In short, for the j-th agent we make the k-th arm optimal. The optimal arm for agent j in the instance $\boldsymbol{\nu}'$ is the arm k.
For any event A (and its complement A^c), due to Pinsker's inequality we have

$$D(\mathbb{P}_{\nu,\pi}, \mathbb{P}_{\nu',\pi}) \geq \log\left(\frac{1}{2(\mathbb{P}_{\nu,\pi}(A)+\mathbb{P}_{\nu,\pi}(A^c))}\right). \tag{11}$$

Let us now consider the event $A = \{N_k^{(j)}(T) \geq T/2\}$. Therefore, due to the regret decomposition lemma 4, we have the regrets:

1. In instance $\boldsymbol{\nu}$ as $R_T(\boldsymbol{\nu}, \pi) \geq R_T^{(j)}(\boldsymbol{\nu}, \pi) \geq \Delta_k^{(j)} \frac{T}{2} \mathbb{P}_{\nu,\pi}\left(\{N_k^{(j)}(T) \geq T/2\}\right)$.
2. In instance $\boldsymbol{\nu}'$ as $R_T(\boldsymbol{\nu}', \pi) \geq R_T^{(j)}(\boldsymbol{\nu}', \pi) \geq (\mu'_{jk} - \mu_{jk_j^*}) \frac{T}{2} \mathbb{P}_{\nu,\pi}\left(\{N_k^{(j)}(T) < T/2\}\right)$.

As the only change in reward distribution happens in agent j, arm k pair, we have from Lemma 17:

$$D(\mathbb{P}_{\nu,\pi}, \mathbb{P}_{\nu',\pi}) = D(\nu_{jk}, \nu'_{jk})\mathbb{E}_{\nu,\pi}[N_k^{(j)}(T)] \leq \left(\epsilon + D_{\inf}(\nu_{jk}, \mu_{jk_*^{(j)}}, \mathcal{P})\right) \mathbb{E}_{\nu,\pi}[N_k^{(j)}(T)].$$

The last inequality holds true by construction of ν'_{jk}.
Substituting the above three relations in Equation (11) we obtain for any $\epsilon > 0$.

$$\left(\epsilon + D_{\inf}(\nu_{jk}, \mu_{jk_*^{(j)}}, \mathcal{P})\right) \mathbb{E}_{\nu,\pi}[N_k^{(j)}(T)] \geq \log\left(\frac{1}{2(\mathbb{P}_{\nu,\pi}(A)+\mathbb{P}_{\nu,\pi}(A^c))}\right) \geq \log\left(\frac{T\min((\mu'_{jk}-\mu_{jk_*^{(j)}}), \Delta_k^{(j)})}{4(R_T^{(j)}(\boldsymbol{\nu},\pi)+R_T^{(j)}(\boldsymbol{\nu}',\pi))}\right).$$

Here, the final inequality hold as the policy π is assumed to be universally consistent. Therefore, taking the following holds after taking the

$$\lim_{\epsilon\to 0}\liminf_{T\to\infty} \frac{\mathbb{E}_{\nu,\pi}[N_k^{(j)}(T)]}{\log T} \geq \lim_{\epsilon \to 0} \frac{1}{\epsilon + D_{\inf}(\nu_{jk}, \mu_{jk_*^{(j)}}, \mathcal{P})} = \frac{1}{D_{\inf}(\nu_{jk}, \mu_{jk_*^{(j)}}, \mathcal{P})}.$$

As the above bound is true for any uniformly consistent policy π, and for agent j, and arm $k \in [K] \setminus \{k_*^{(j')} : 1 \leq j' \leq j\}$. We use the regret decomposition lemma (Lemma 4) to obtain the final aysmptotic regret lower bound for any agent $j \in [N]$ as

$$\liminf_{T\to\infty} \frac{R_T^{(j)}(\boldsymbol{\nu})}{\log T} \geq \max\left\{\sum_{j'=1}^{j-1} \frac{\Delta_{\min}^{(j)}}{D_{\inf}(\nu_{j'k_*^{(j)}}, \mu_{j'k_*^{(j')}}, \mathcal{P})}, \sum_{k \notin \mathcal{A}_*^{(j)} \setminus k_*^{(j)}} \frac{\Delta_k^{(j)}}{D_{\inf}(\nu_{jk}, \mu_{jk_*^{(j)}}, \mathcal{P})}\right\}$$

Here, we use the fact that $k_*^{(j)} \notin \mathcal{D}_*^{(j')} \cup \{k_*^{(j')}\}$ for all $j' < j$. Also note, $\sum_{j'=1}^{0}(\cdot) = 0$ and $\mathcal{D}_*^{(1)} = \emptyset$ for the highest ranked arm. □

D.4 Proof of Corollary 6

The above corollary follows readily from Theorem 5. Let for agent j' from 1 to $j-1$ the optimal arm be j' with mean $1/2$ and all the other arms have mean $1/2 - \Delta$, where $\Delta > 0$ is small enough. Also, let the j-th agent have the arm means between $1/2$ for the j-th arm and $1/4$ for any other arm. For $\mathcal{P}$ the class of Bernoulli rewards, we have $D_{\inf}(\nu_{j'k_*^{(j)}}, \mu_{j'k_*^{(j')}}, \mathcal{P}) \leq \Delta^2/4$ for all $j' \leq j-1$, and $\Delta_{\min}^{(j)} = 1/4$. Therefore, the regret of the j-th agent is lower bounded as $\frac{(j-1)\log(T)}{16\Delta^2}$.

D.5 Gap of Upper vs Lower Regret Bounds

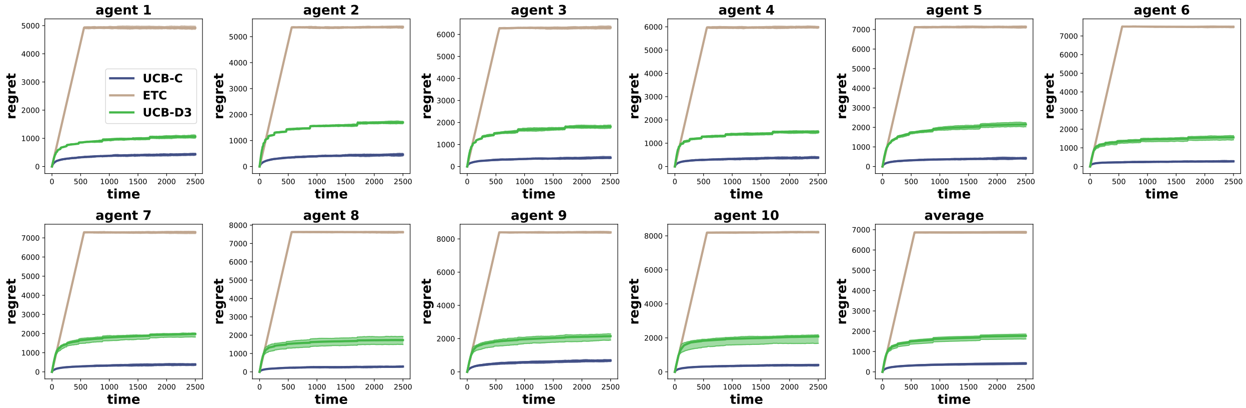
In the above proof, when an agent 1 to $j-1$ plays the stable arm (which is also optimal in a OSB instance) arm for agent j the authors of this paper lack an understanding of the following.

1. Whether agent j is able to anticipate the collision and move to a sub-optimal arm that is collision free. If such anticipation is missing the agent accrues $\mu_{jk_*^{(j)}}$ regret (not $\Delta_{\min}^{(j)}$).
2. Given that the agent j avoids a collision, it is unclear whether the agent successfully plays the second-best arm. It is possible that it accrues $\Delta_{\max}^{(j)}$ regret instead in the worst case.

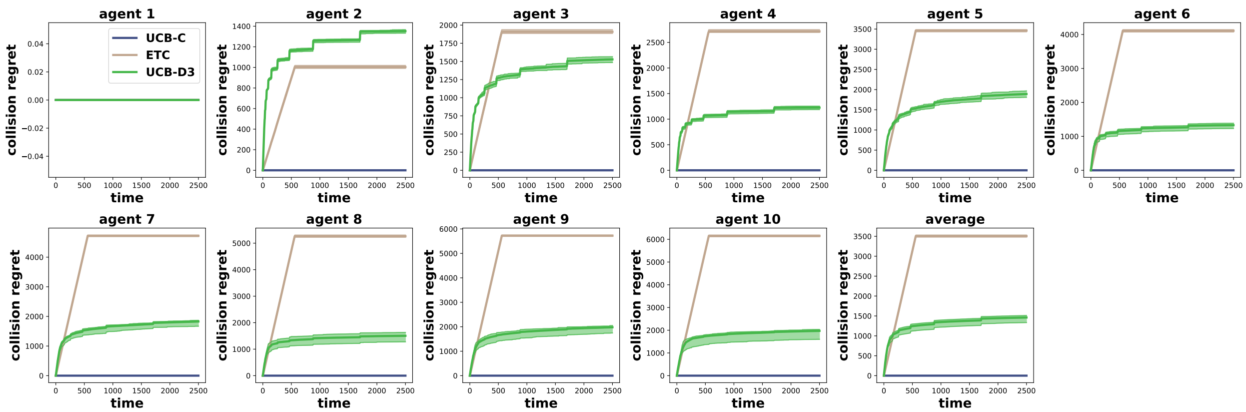
The above scenarios lead to the gap in the upper versus lower regret bounds. For closing such gaps it becomes essential to formalize anticipatory behavior among multi-agent decentralized bandits. This is beyond the scope of the current paper. Transfer of knowledge from lower to higher rank agents also is not possible here which makes mimicking a centralized algorithm difficult in our setting.

E Additional Simulation Results

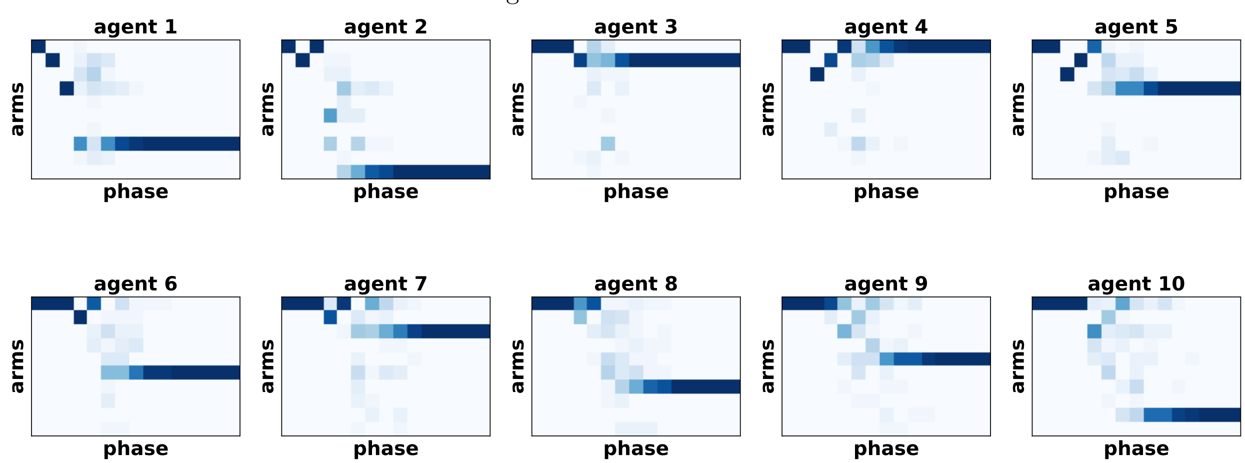
In this section, we present the additional plots described in Section 7.



(a) Regret plots of all agents.

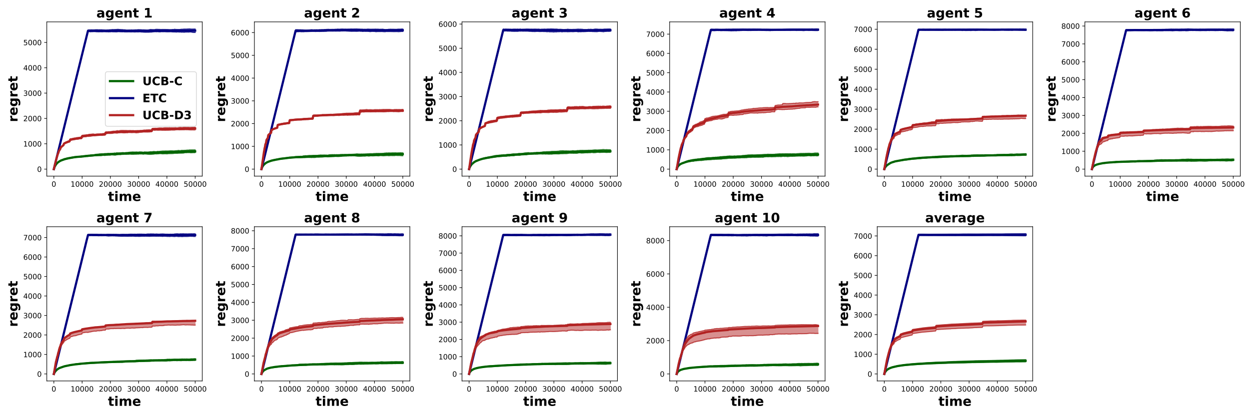


(b) A plot showing the cumulative regret only due to collisions. The centralized UCB ensures that agents never collide and thus do not lose out on regret.

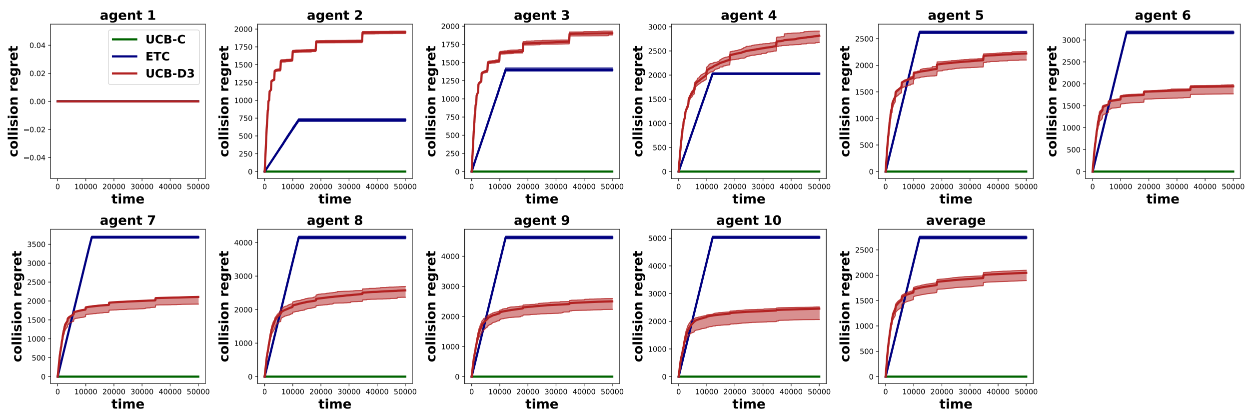


(c) Arms recommended by the agents across phases over different runs of the algorithm.

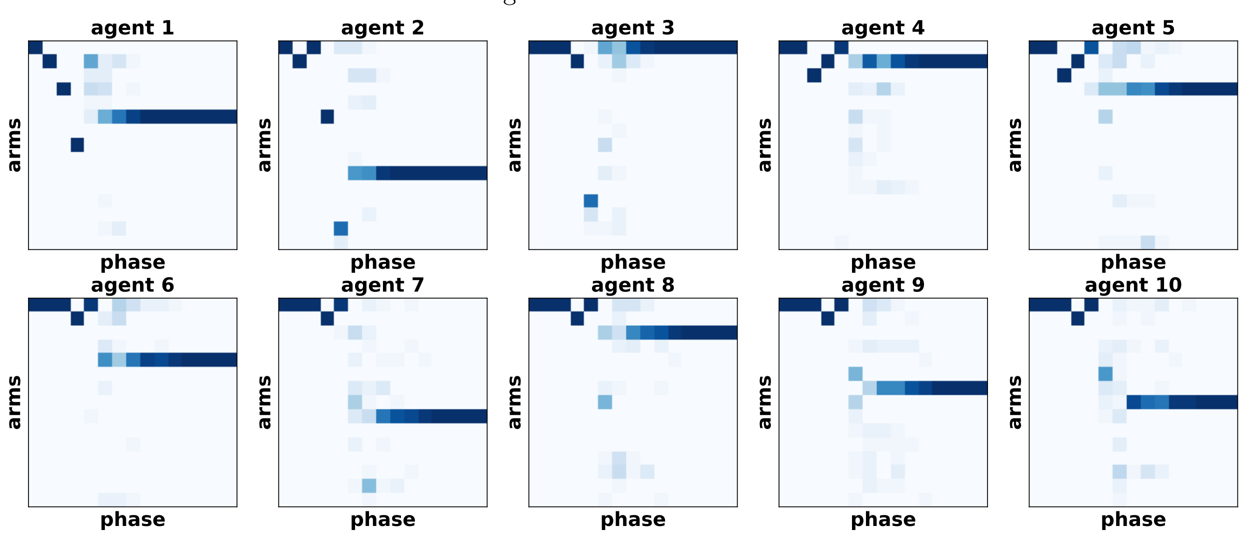
Figure 4: Simulations on a system with 10 agents and 10 arms. The arm-means for sub-optimal arms for each agent are chosen i.i.d. uniformly over $[0, 0.8]$, while the arm-mean of agent $i \in [10]$ for arm $\sigma(i)$ (its optimal stable match arm) was set to 0.9. The rewards are binary. Here, $\sigma(\cdot)$ denotes a permutation. This is thus a OSB instance. The value of $H = 1117$ used for ETC.



(a) Regret plots of all agents

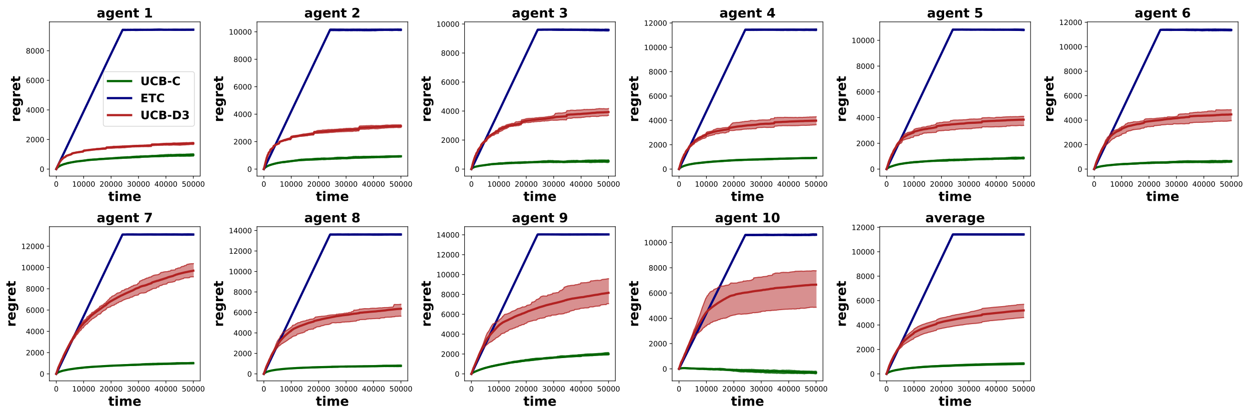


(b) A plot showing the cumulative regret only due to collisions. The centralized UCB ensures that agents never collide and thus do not lose out on regret.

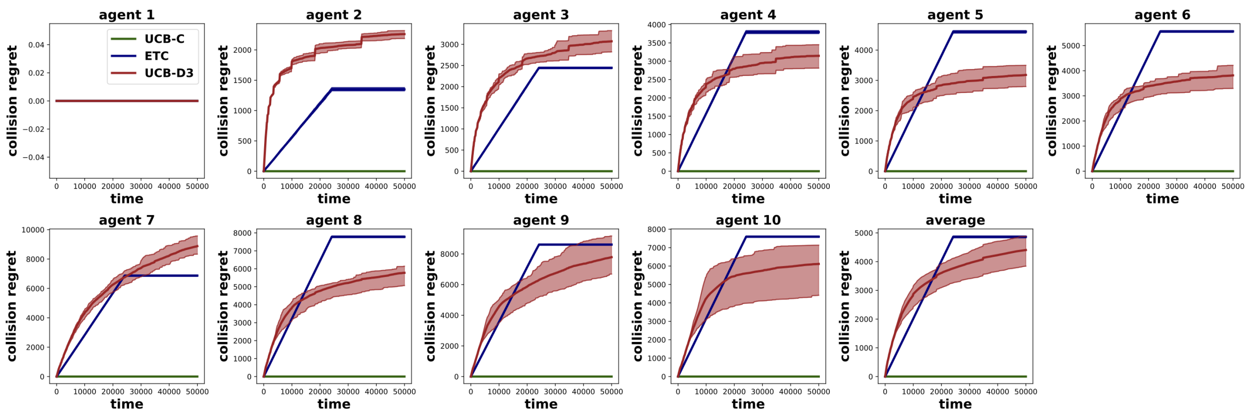


(c) Arms recommended by the agents across phases over different runs of the algorithm.

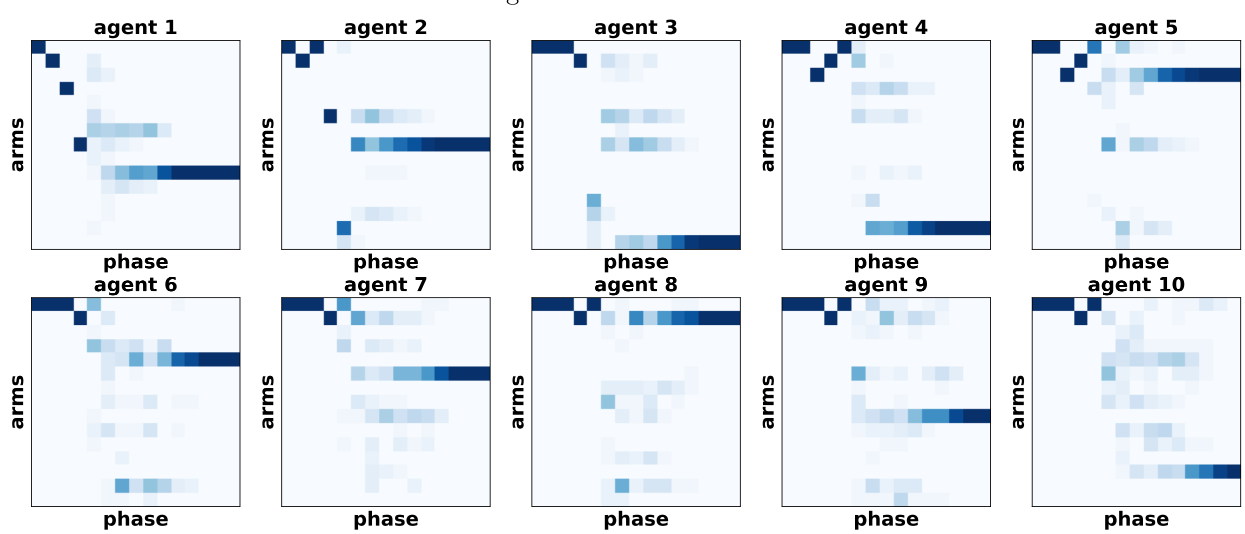
Figure 5: Simulations on a system with 10 agents and 15 arms. The arm-means for sub-optimal arms for each agent are chosen i.i.d. uniformly over $[0, 0.8]$, while the arm-mean for agent $i \in [10]$ and arm $\sigma(i)$ (stable match partner arm) was set to 0.9. The rewards are binary. Here, $\sigma(\cdot)$ denotes a permutation. This is thus a OSB instance. The value of $H = 805$ was used for ETC.



(a) Regret plots of all agents



(b) A plot showing the cumulative regret only due to collisions. The centralized UCB ensures that agents never collide and thus do not lose out on regret.



(c) Arms recommended by the agents across phases over different runs of the algorithm.

Figure 6: Simulations on a system with 10 agents and 15 arms. For each agent $i \in [10]$, a permutation over the arms σ_i was chosen, and the arm-means are equally spaced among the 7 arms from 0.1 to 0.9 in the increasing order of permutation. This is thus not a OSB instance. The rewards are binary. The value of $H = 1610$ was used for ETC.