Cooperative Stochastic Bandits with Asynchronous Agents and Constrained Feedback

Lin Yang, Yu-Zhen Janice Chen
University of Massachusetts Amherst
{linyang,yuzhenchen}@cs.umass.edu

Stephen Pasteris
University College London
stephen.pasteris@gmail.com

Mohammad H. Hajiesmaili
University of Massachusetts Amherst
hajiesmaili@cs.umass.edu

John C. S. Lui
Chinese University of Hong Kong
cslui@cse.cuhk.edu.hk

Don Towsley
University of Massachusetts Amherst
towsley@cs.umass.edu

Abstract

Motivated by the scenario of large-scale learning in distributed systems, this paper studies a scenario where M agents cooperate together to solve the same instance of a K-armed stochastic bandit problem. The agents have limited access to a local subset of arms and are asynchronous with different gaps between decision-making rounds. The goal is to find the global optimal arm and agents are able to pull any arm, however, they can only observe the reward when the selected arm is local. The challenge is a tradeoff for agents between pulling a local arm with observable feedback, or pulling external arms without feedback and relying on others' observations that occur at different rates. We propose `AAE-LCB`, a two-stage algorithm that prioritizes pulling local arms following an active arm elimination policy, and switches to other arms only if all local arms are dominated by some external arms. We analyze the regret of `AAE-LCB` and show it matches the regret lower bound up to a small factor.

1 Introduction

Multi-armed bandits (MABs) [16, 49] is a remarkably successful online learning framework that has been extensively studied since the 1950s [46]. It has a broad range of applications including datacenter optimization, web advertising, and recommender systems [49, 31, 39, 15]. In the basic MAB problem, a learner repeatedly chooses an action (pulls an arm) in each round, and collects (and observes) the reward associated with the selected arm, but not rewards associated with the unselected arms. The goal of the learner is to maximize the long-term reward collected, and the performance metric is regret, which is the difference between the expected reward for the best arm and that received by the learner.

The distributed/multi-agent/multiplayer MAB problem is an extension of the basic MAB that has been studied extensively recently in different settings [43, 50, 11, 36, 33, 25, 44, 52, 47]. This problem is motivated by several applications such as (1) large-scale learning systems [19] in domains such as finance or recommendation systems; (2) cooperative search by multiple robots [41, 32]; (3) the application of wireless cognitive radio [17, 43, 42, 9]; and (4) distributed learning in distributed systems (e.g., a set of IoT devices learning about an underlying environment) [45, 23, 5, 12, 53].

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

Most prior work on the multi-agent MAB problem assumes that agents have full access to all arms, and hence they solve the same MAB problem with the goal of minimizing the aggregate regret of the agents either in a *competition* setting [3, 14, 17, 52, 11, 13, 43, 42, 9] where agents receive degraded or no rewards when pulling the same arm, or in a *collaboration/cooperation* setting [44, 52, 38, 37, 36, 47], where agents receive independent rewards when they pull the same arm, and agents can communicate to improve their learning performance. In this work, we focus on the cooperative version of this problem, which has been extensively studied in recent years under different settings. In a thread of work, the agents are considered to be connected together over a communication graph [44, 52, 38, 37, 10, 24]. Among them, [52] presents a more practical model in which agents are limited in their available communication capacity with others, and also develops the state-of-the-art algorithm which achieves an asymptotically optimal regret. The above basic models have been extended to several other settings such as pure-exploration through cooperation among agents for identifying the best stochastic arm [29], cooperative multi-agent bandits with heavy tails [25], cooperative kernelised contextual bandits [27], linear bandits with safety constraints [2], and the case with both honest and malicious agents [51]. In [10], the problem is extended to the case where the underlying topology is time-varying. In another trend, the cooperative bandits has been studies over social networks [47, 50, 36]. Last, in a very recent trend the stochastic bandits has extended to the federated setting, in which the learning task is distributed among multiple agents [54, 48, 26].

In this paper, we study a heterogeneous version of the cooperative multi-agent stochastic bandit problem with a set $\mathcal{A} = \{1, \dots, M\}$ of agents and a set $\mathcal{K} = \{1, \dots, K\}$ of arms. Agent $j \in \mathcal{A}$ has access to a subset $\mathcal{K}_j \subseteq \mathcal{K}$ of arms. We refer to arms in $\mathcal{K}_j$ as *local* and the remaining arms in $\mathcal{K} \setminus \mathcal{K}_j$ as *external* arms. Agents come with different learning capabilities and can pull an arm every $1/\theta_j$ rounds, with $0 < \theta_j \leq 1$ as the action rate of agent j. The goal is to collaboratively find the optimal arm in $\mathcal{K}$. An agent can pull any arm and receives a reward; however, it *observes* the reward only when it pulls a local arm. It instantaneously forwards this observation to all other agents. We call this setup *Feedback-constrained Cooperative Multi-agent MAB* (`FC-CMA2B`) and formally define it in Section 2.

The `FC-CMA2B` setting arises naturally in large-scale learning systems that are often geographically distributed for domains such as web search, finance, and recommendation systems. In such large-scale systems, it is usually impossible to run a single bandit algorithm. Instead, a distributed set of agents work to solve the problem, each with access to a subset of the action space. However, the entire learning task is integrated and the goal is to find the best global action, e.g., the most relevant search result in a web search application, or the best possible video recommendation in the YouTube recommendation system among millions of possible recommendations. In the above scenario, learning among different agents may be *asynchronous* in the sense that each agent has its own action (decision making) rate, which is captured by `FC-CMA2B`. In Appendix A, we provide a few concrete application scenarios that could be captured by `FC-CMA2B`.

The algorithmic challenge in `FC-CMA2B` originates from a nontrivial tradeoff that an agent must make between pulling a local arm with an observable reward, and pulling an external arm with the expectation of a larger but not observable reward. One might simply run a cooperative `UCB` algorithm on each agent without any distinction between local and external arms and pull the arm with the highest confidence interval. Then, if the selected arm is local, the agent can broadcast its observation to others. However, this strategy suffers a poor regret. The reason is intuitive. Consider a case where a suboptimal arm is only observable by a slow agent, i.e., an agent with a very small action rate θ_j. In such a case, the upper confidence bound of the suboptimal arm will remain large due to the limited observations available only to the slow agent. Since the indexing policy of cooperative `UCB` suggests pulling the arm with the highest upper confidence bound, then, the suboptimal arm will be continuously pulled by everyone, resulting in a poor regret.

Contributions. This paper makes the following contributions.

Algorithm design using a two-stage learning strategy. To tackle the above challenge of `FC-CMA2B`, a proper bandit algorithm should be able to collect sufficient reward information, which is possible only by pulling local arms. To achieve this, our high-level idea is to distinguish between local and external arms and prioritize the local arms and only switch to external arms when they are clearly better. We implement this idea by proposing `AAE-LCB`, a two-stage learning strategy, which in the

first stage selects local arms based on a carefully-designed *active arm elimination* (AAE) policy, and in the second stage selects external arms based on a *lower confidence bound* (LCB) policy.

More specifically, in the *first stage* of AAE-LCB, each agent pulls only from a dynamically-constructed candidate set of local arms. Then through an AAE process, a local arm is removed from the candidate set when its confidence interval falls below that of at least one other (local or external) arm. With this strategy, AAE-LCB makes a sufficient number of observations of local arms, and resolves issues encountered by the cooperative UCB. For an agent whose local arms do not include the optimal arm, the elimination process continues until all local arms are eliminated, and when so, the algorithm enters a *second stage*, where the agent only pulls external arms.

In the second stage, the agent totally relies on external observations that may occur at different rates. Differences in observation rates complicates decision making regarding choice of external arm such that the agent has to trade off between well observed arms, and those that are not well observed but likely to be optimal. To tackle this tradeoff, AAE-LCB pulls an external arm with the largest LCB, and the intuition is that an arm with greater LCB is both better observed and more likely to be optimal. Note that selecting based on largest UCB may lead to pulling external arms whose upper confidence bounds are large due to a lack of samples.

Regret lower and upper bounds. We analyze the regret of AAE-LCB and show that it achieves a regret of $O(\sum_{i:\Delta_i>0} \Theta_i/\Theta_{i^*} \times K \log T/\Delta_i)$, where Θ_i is the action rate of arm i defined as the sum of the action rates of agents containing arm i and i^* is the optimal arm. We further establish a regret lower bound for FC-CMA2B, that shows AAE-LCB is optimal up to a small factor. For comparison, we investigate the regret of a baseline algorithm, AAE-AAE, which pulls external arms based on another AAE process and show that it exhibits poor performance in the presence of agents with low action rates. Specifically, we show that AAE-AAE suffers a regret of $\Omega(\Theta/\Theta_{\min} \log T)$, where $\Theta = \sum_{j\in\mathcal{A}} \theta_j$ and $\Theta_{\min} = \min_{i\in\mathcal{K}} \sum_{j:i\in\mathcal{K}_j} \theta_j$ are the aggregate action rate of all agents, and the *minimum* aggregate action rate of agents containing any arm i, respectively. This result shows that regret can be arbitrarily bad when $\Theta_{\min}$, i.e., the minimum observation rate among all arms (either optimal or suboptimal), is small. Hence, instead of regret depending on the observation rate of the least observable (and possibly suboptimal) arms, the regret of AAE-LCB depends on that of the optimal arm.

Numerical results. Through brief numerical experiments, we verify our theoretical observations and show that the cooperative extension of UCB and AAE-AAE are either unable to gain sufficient observations for the global optimal arm or vulnerable to the action rate of slow agents with suboptimal arms, while AAE-LCB is robust to both issues and achieves much better performance.

2 Problem Setup

Feedback-constrained Cooperative Multi-agent MAB (FC-CMA2B). We consider a multi-agent stochastic bandit setting with a set $\mathcal{A} = \{1, \dots, M\}$ of independent agents existing over the entire time period, and a set $\mathcal{K} = \{1, \dots, K\}$ of arms. Associated with arms are mutually independent sequences of i.i.d. Bernoulli rewards with mean $0 \le \mu(i) \le 1$, for arm $i \in \mathcal{K}$. Agent $j \in \mathcal{A}$ has full access to a subset $\mathcal{K}_j \subseteq \mathcal{K}, K_j = |\mathcal{K}_j|$, of arms. We refer to $\mathcal{K}_j$ and $\mathcal{K} \setminus \mathcal{K}_j$ as the sets of local and external arms of agent j, respectively. Agents are allowed to pull and receive a reward from any arm from $\mathcal{K}$, however, they only receive observations from local arms. That is to say, if the selected arm is external, the agent does not observe the reward even though the reward is allocated. Last, we make no assumption regarding overlaps among sets $\mathcal{K}_j$.

In addition to heterogeneity in access to arms, agents are also heterogeneous in their decision making capabilities. Specifically, considering the decision rounds $\{1, \dots, T\}$, agent j is able to pull an arm every $\omega_j \in \mathbb{N}^+$ rounds, i.e., decision rounds for agent j are $t = \omega_j, 2\omega_j, \dots, N_j\omega_j$, where $N_j = \lfloor T/\omega_j \rfloor$. Parameter ω_j represents the *inter-round gap* of agent j. For simplicity of analysis, we define $\theta_j := 1/\omega_j$ as the *action rate* of agent j. Intuitively, the larger θ_j, the faster the agent j is in pulling arms.

The goal of each agent is to learn the global optimal arm. The regret for agent j in FC-CMA2B is

$$R_T^j := \mu(i^*)N_j - \sum\nolimits_{t\in\{k\omega_j : k=1,\dots,N_j\}} x_t(I_t^j),$$

where i^* is the optimal arm in $\mathcal{K}$, and $I_t^j \in \mathcal{K}$ is the pulled arm by agent j at time t. Agent j receives the reward $x_t(I_t^j)$ whose value is observable only if $I_t^j \in \mathcal{K}_j$. The ultimate goal is to minimize the aggregate regret of the agents, i.e., $R_T = \sum_{j\in\mathcal{A}} R_T^j$.

We note that when two agents select the same arm, they collect stochastically independent rewards, and there is no reward degradation such as occurs in competition settings. We also consider full and truthful cooperation among agents, i.e., after committing to one decision and receiving a reward from the pulled arm, an agent broadcasts observations to others at no cost and no delay. In the supplementary material, we further consider a more general model with communication delays among agents.

Additional notations and terminologies. To facilitate our algorithm design and analysis, we introduce the following notations. Let $\mathcal{A}_i$ denote the set of agents whose local arm sets include arm i,

$$\mathcal{A}_i := \{j \in \mathcal{A} : i \in \mathcal{K}_j\}, \quad \forall i \in \mathcal{K}.$$

By $\Delta(i', i)$, we denote the difference in mean rewards of arms i' and i, i.e., $\Delta(i', i) := \mu(i') - \mu(i)$. When $i' = i^*$, we rewrite $\Delta(i^*, i)$ as Δ_i, which is known as the suboptimality gap in the basic bandit problem. Last, we define Θ and Θ_i, $i \in \mathcal{K}$, as follows

$$\Theta := \sum_{j\in\mathcal{A}} \theta_j, \quad \text{and} \quad \Theta_i := \sum_{j\in\mathcal{A}_i} \theta_j,$$

where Θ_i is the action rate of arm i across all agents in $\mathcal{A}_i$. Intuitively, the larger Θ_i is, the higher the rate at which arm i can be pulled by agents in $\mathcal{A}_i$. The action rate Θ_i plays a key role in characterizing the regret bounds.

3 Algorithm Design

Each agent in the `FC-CMA2B` setting has to resolve a tradeoff between two categories of arms: (i) local arms with observable rewards, and (ii) external arms whose rewards are not observable but may be better than local arms. This heterogeneity in information feedback motivates our two-stage learning algorithm, `AAE-LCB`. In the first stage, agents running `AAE-LCB` are focused on sampling local arms to collect sufficient information that is useful for all agents in finding the global optimal arm. And when with high probability, an agent is confident that its local set does not contain the optimal arm, it moves to the second stage by pulling external arms based on others' observations.

3.1 Stage 1: Selecting a local arm

In order to converge to the global optimal arm, agents must collect sufficient observations in the first stage. Toward this, our idea is to prioritize pulling local arms as much as possible and only switch to external arms when the agent finds an external arm that is *sufficiently better* than all its local arms with high confidence. To implement this idea, we extend the active arm elimination (`AAE`) algorithm [28, 4] to capture the additional issues in `FC-CMA2B`. Similar to the basic `AAE` algorithm, in our algorithms, we start off by constructing a dynamic candidate set of local arms and gradually eliminating arms from it. The elimination process is similar to the basic `AAE`, which eliminates an arm when its confidence interval falls below that of at least one other arm. However, the difference is that while the candidate set includes local arms, the criteria for eliminating an arm is based on a comparison between both local and external arms.

Specifically, we define the width of the confidence interval for arm i and agent j at time t as

$$\texttt{cint}(i, j, t) := \sqrt{\frac{\alpha \log \delta_t^{-1}}{2\hat{n}_t^j(i)}}, \tag{1}$$

where $\hat{n}_t^j(i)$ is the total number of observations (either local or received from other agents) of arm i available to agent j by time t (observations made in time slots from 1 to $t-1$). Also, $\delta_t > 0$ and $\alpha > 2$ are parameters of the confidence interval. The removal of arm i from agent j's candidate set at time t with empirical mean of $\hat{\mu}(i, \hat{n}_t^j(i))$ occurs if there is another arm i', such that the

Algorithm 1 AAE-LCB: A Cooperative Bandit Algorithm for Agent j in the FC-CMA2B setting

1: **Initialization:** $\hat{n}^j(i) = 0$, $\hat{\mu}(i)$, $i \in \mathcal{K}$, $\mathcal{C}_j = \mathcal{K}_j$, $\alpha > 2$, δ_t.
2: **for** each decision round t **do**
3: **for** each new received $x_\tau(i)$, $i \in \mathcal{K}$, $\tau < t$, **do**
4: Execute line (8)-(10) for arm i
5: **end for**
6: **if** $\mathcal{C}_j \neq \emptyset$ **then** ▷ *Stage 1: Pulling local arms through an* AAE *process*
7: Pull arm I_t^j with the least no. of observations from $\mathcal{C}_j$
8: Increase counter for the selected arm by 1: $\hat{n}^j(I_t) \leftarrow \hat{n}^j(I_t) + 1$
9: Update estimate of the mean value of the selected arm $\hat{\mu}(I_t)$
10: Reconstruct candidate set based on updated values of $\hat{n}^j(I_t)$ and $\hat{\mu}(I_t^j)$ as in Equation (2)
11: Broadcast $x_t(I_t^j)$ to other agents
12: **else** ▷ *Stage 2: Pulling external arms through a* LCB *process*
13: Select an external arm with the largest lower confidence bound as in Equation (3)
14: **end if**
15: **end for**

difference in empirical means of i and i' is larger than the widths of the confidence intervals, i.e., $\hat{\mu}(i', \hat{n}_t^j(i')) - \hat{\mu}(i, \hat{n}_t^j(i)) > \texttt{cint}(i', j, t) + \texttt{cint}(i, j, t)$. This means that with high probability, arm i is not optimal. Note that i' could be either a local or an external arm. More formally, agent j constructs the following dynamic candidate set $\mathcal{C}_{j,t}$ at time t during its learning process.

$$\mathcal{C}_{j,t} = \left\{ i \in \mathcal{K}_j : \hat{n}_t^j(i) = 0 \ \texttt{or} \ \hat{\mu}(i, \hat{n}_t^j(i)) - \hat{\mu}(i', \hat{n}_t^j(i')) \leq \texttt{cint}(i, j, t) + \texttt{cint}(i', j, t), \forall i' \in \mathcal{K} \right\}. \tag{2}$$

Note that in the above construction the local candidate set of agent j includes all local arms not yet observed and all others not dominated by any other local or external arm. If the local candidate set is not empty, an agent always prioritize selecting local arms and, following basic AAE, pulls a local arm in the candidate set with the least number of observations, and then updates the empirical mean values and broadcasts the observation to others as in Lines 8-10 of Algorithm 1. We also note that the size of the dynamic candidate set is not necessarily decreasing and it is possible that after pulling some external arms, some local arms are put back into the local candidate set. Hence, in run time, fluctuations between pulling local and external arms might be possible.

3.2 Stage 2: Selecting an external arm

As Stage 1 of the learning process moves forward, it is possible that the local candidate set of an agent will become empty, i.e., an external arm eventually dominates all local arms. In this situation, the question becomes how to pick an external arm. The answer to this question is critical in designing cooperative algorithms with low regret.

Different action rates among different agents can lead arms to suffer different observation limitations that can invalidate the selection indices commonly used in the bandit algorithms. Specifically, when a suboptimal arm is only accessible to slow agents, estimates of the empirical mean reward of this arm will be updated more slowly than those of other arms pulled by faster agents, and thus contain more errors. In this way, those algorithms, which are unaware of confidence of estimates, may be misled into selecting a suboptimal arm with low-level confidence in the FC-CMA2B setting. For example, "slow arms", i.e., those arms that are only in the local sets of slow agents, have much looser confidence interval and a much larger upper confidence bound than others. As a result, fast agents running a cooperative version of AAE or UCB-based algorithms that pull external arms based on estimated upper confidence bounds may be misled into continuously selecting a suboptimal "slow" arm because its upper confidence bound is large due to insufficient number of observations. To address this, we propose to follow a Lower Confidence Bound (LCB) policy, which selects the external arm with the largest lower confidence bound, i.e.,

$$I_t^j = \arg \max_{i \in \mathcal{K}/\mathcal{K}_j} \hat{\mu}(i, \hat{n}_t^j(i)) - \texttt{cint}(i, j, t). \tag{3}$$

An important observation on LCB is that the lower confidence bound of an arm is large only if it is well observed and hence is likely to be the optimal arm. Indeed this is not the case if the selection policy is based on the largest upper confidence bound, since external arms with low observation might have large upper confidence bound.

3.3 Baseline Algorithms: CO-UCB and AAE-AAE

To show the advantages of AAE-LCB, we introduce two baseline algorithms. The first one is CO-UCB, a cooperative version of UCB that the agents do not distinguish between local and external arms. In other words, each agent running CO-UCB, pulls the arm with the largest upper confidence bound and, if the information is observable, broadcasts it to all other agents. Further, each agent will updates its confidence intervals not only based on its local observations, but also, based on the information received from other agents.

The second algorithm is AAE-AAE, which has a similar two-stage structure to prioritize pulling local arms as AAE-LCB, but adapts another layer of active arm elimination for pulling external arms. In the second stage, i.e., when the local set is emptied, AAE-AAE randomly pulls an external arm whose confidence interval have overlaps with others' and dynamically reconstructs the external candidate set similar to the process in the first stage, but for external arms. Because external arms with few observations has loose confidence interval, they satisfy the criteria to be in the candidate set, hence, the agents running AAE-AAE may repeatedly pull them.

In Section 5, we numerically compare the performance of both baseline algorithms with AAE-LCB and show the poor performance of CO-UCB due to the same treatment for local and external arms, and the improvement of AAE-LCB over AAE-AAE due to better treatment of external arms. In addition, in the next section, we theoretically compare the regret of AAE-LCB and AAE-AAE.

4 Regret Results

Two main regret results are highlighted in the following two theorems. We first introduce some terminology to facilitate the presentation of the results. Consider an algorithm π running on agents in FC-CMA2B. We say that π is *consistent*, if its regret satisfies $\mathbb{E}[R_T(\pi)] = O((T\Theta)^\sigma)$ as $T \to +\infty$ for any $\sigma > 0$, and for any set of Bernoulli reward distributions. Further, let $\mathrm{KL}(\mu_i, \mu_i + \Delta_i)$ refer to the the Kullback-Leibler divergence between a Bernoulli of parameter μ_i and $\mu_i + \Delta_i$.

Theorem 1 (Asymptotic Regret Lower Bound for FC-CMA2B) *For any consistent algorithm π and any $0 < \sigma < 1$, its expected regret satisfies*

$$\liminf_{T\to+\infty,\Theta/\Theta_{i^*}\to+\infty} \frac{\mathbb{E}[R_T(\pi)]}{(\Theta/\Theta_{i^*})^\sigma \log(T\Theta)} = \Omega\Big(\sum_{i:\Delta_i>0} \frac{\Delta_i}{\mathrm{KL}(\mu_i, \mu_i+\Delta_i)}\Big).$$

The proof for the lower bound consists of two steps. In the first one, we prove that under any given agent action rates, any algorithm suffers a regret of $\Omega(\log(T\Theta))$ for large T. This proof uses techniques commonly used to prove lower bounds for stochastic bandits. In the second step, we prove that, with large enough T and Θ/Θ_{i^*}, no algorithm can have a dependency on $(\Theta/\Theta_{i^*})^\sigma$ for $0 < \sigma < 1$. This is proved by contradiction. The details are given in the supplementary material.

Theorem 2 (Expected Regret for AAE-LCB) *Define $\delta := \min_{l,j} \delta_{l/\theta_j}$. The expected regret of AAE-LCB with parameter $\alpha > 2$ has the following upper bound*

$$\mathbb{E}[R_T] \le \sum_{i:\Delta_i>0} \max\left\{\frac{4\alpha\log\delta^{-1}}{\Delta_i}\left(2+\frac{K\Theta_i}{\Theta_{i^*}}\right), \frac{12\Theta_i\alpha K\log\delta^{-1}}{\Delta_i\Theta_{i^*}}\right\}$$
$$+ 2\left(1+\frac{\Theta_i}{\Theta_{i^*}}\right)\sum_{j\in\mathcal{A}}\sum_{l=1}^{N_j}\sum_{i\in\mathcal{K}_j}\frac{l\Theta_i}{\theta_j}\delta_{l/\theta_j}^{\alpha}.$$

Corollary 1 *With $\delta_t = 1/t$ and $\alpha > 2$, we have the following regret for the AAE-LCB algorithm.*

$$\mathbb{E}[R_T] = O\left(\sum_{i\in\mathcal{K}:\Delta_i>0}\frac{K\Theta_i\log T}{\Theta_{i^*}\Delta_i}\right).$$

The prove is given in Section 4.1. From Corollary 1 and Theorem 1, we observe that the asymptotic regret of the proposed AAE-LCB algorithm matches the lower bound up to a small factor $O(K(\Theta/\Theta_{i^*})^{1-\sigma})$, where σ is arbitrarily close to 1.

We also compare the regret of AAE-LCB to that of AAE-AAE. With $\delta_t = 1/T$, and $\alpha > 4$, the regret of AAE-AAE could be characterized as

$$\mathbb{E}[R_T] = \Omega\left(\frac{\Theta}{\Theta_{\min}} \log T\right), \tag{4}$$

where $\Theta_{\min} := \min_i \Theta_i$. The detailed derivation of the the above regret is in the supplementary materials. The regret of AAE-AAE in (4) strongly depends on $\Theta_{\min}$, i.e., the smallest aggregate action rate among all arms. Hence, as $\Theta_{\min}$ goes to zero, the regret of AAE-AAE significantly degrades. The regret of AAE-LCB shown in Corollary 1, however, depends only on the action rate of the optimal arm. Hence, AAE-LCB outperforms AAE-AAE, in the sense that its performance is independent of the slowest arm.

4.1 A Proof for Theorem 2

There are two contributions to the regret of AAE-LCB: one due to pulling suboptimal local arms, and the other one due to pulling suboptimal external arms. To prove Theorem 2, we analyze these two contributions. Note that, observations are generated whenever agents pull local arms. Thus, the first contribution to regret can be upper bounded by analyzing the relationship between the confidence interval and the number of observations obtained from each arm. By analyzing the algorithm rules of AAE-AAE, we upper bound the expected number of observations required from suboptimal arms to identify the optimal arm if it lies in the local set, as well as the first contribution to regret. Regarding the second contribution to regret, the main difficulty comes form the heterogeneity of action rates associated with the arms. To upper bound this contribution to regret, we first upper bound the time period needed for the agents containing the optimal arm in their local sets generate enough observations such that the lower confidence bound of the optimal arm is higher than those of any others. With this, we can prove the the final result by upper bounding the number of times that suboptimal external arms need to be selected in the above proved time period.

Now, we proceed to formally prove the regret result. First, we categorize decisions made by the agents into Type-I and Type-II decisions. Type-I corresponds to the decisions of an agent when the actual mean values of local arms lie in the confidence intervals calculated by the agent, otherwise, Type-II decision happens, i.e., the actual mean value of some local arm is not within the calculated confidence interval. Specifically, when agent j makes a Type-I decision at time t, we have

$$\mu(i) \in \left[\hat{\mu}\left(i, \hat{n}_t^j(i)\right) - \texttt{cint}(i,j,t), \hat{\mu}\left(i, \hat{n}_t^j(i)\right) + \texttt{cint}(i,j,t)\right], \quad i \in \mathcal{K}_j.$$

With Type-I decisions, an agent can keep the local optimal arm in its candidate set and eventually converges its decisions to the local optimal arm. The following Lemma, whose proof is given in the supplementary material, provides the probability that a Type-I decision happens at a particular decision round.

Lemma 1 *Agent j running* AAE-AAE *with $\alpha > 2$ makes a Type-I decision, in round $t = l/\theta_j$, with a probability of at least* $1 - 2\sum_{i \in \mathcal{K}_j} \frac{l\Theta_i}{\theta_j} \delta_{l/\theta_j}^{\alpha}$.

In the AAE-LCB algorithm, a suboptimal arm pulled by agent j lies either in the candidate set or is the one with the largest lower confidence bound among all arms. Thus, we split the regret analysis into two cases: *Case I: local arm selection*, where a suboptimal decision is made during the arm elimination period for arms lying in the local set; and *Case II: external arm selection*, where a suboptimal decision is made based on the lower confidence bound when selecting external arms. In the following, we analyze the regret in each case separately.

Case I: regret due to local arm selection: For a suboptimal arm i, we upper bound the total number of pulling times by agents in $\mathcal{A}_i$, which is actually equal to $\hat{n}_t^j(i)$. First, we focus on the cases that the algorithm makes a Type-I decision at time t, i.e., the mean value of any arm lies in its confidence interval calculated by agent j. Then, at time t, if agent j in $\mathcal{A}_i$ selects arm i, we have

$$2\texttt{cint}(i,j,t) + 2\texttt{cint}(i^*,j,t) \geq \Delta_i. \tag{5}$$

Otherwise, we have

$$\begin{aligned}\hat{\mu}(i, \hat{n}_t^j(i)) + \texttt{cint}(i,j,t) \leq & \mu_i + 2\texttt{cint}(i,j,t) < \mu_i + \Delta_i - \texttt{cint}(i^*,j,t) \\ = & \mu_{i^*} - 2\texttt{cint}(i^*,j,t) \leq \hat{\mu}(i^*, \hat{n}_t^j(i^*)) - \texttt{cint}(i^*,j,t),\end{aligned}$$

implying that arm i is strictly dominated by i^*, hence, i can not be selected by agent j, contradicting the assumption that i is selected by j. It follows from Equation (5) that

$$\max\left\{2\texttt{cint}(i,j,t), 2\texttt{cint}(i^*,j,t)\right\} \geq \frac{\Delta_i}{2},$$

and by the definition of $\texttt{cint}(.)$ in Equation (1), we have

$$\min\left\{\hat{n}_t^j(i), \hat{n}_t^j(i^*)\right\} \leq \frac{8\alpha\log\delta_t^{-1}}{\Delta_i^2}. \tag{6}$$

Now, we focus on Type-II decisions. Let Q denotes the number of Type-II decisions. By Lemma 1, we have

$$\mathbb{E}\left[Q\right] \leq 2\sum_{j\in\mathcal{A}}\sum_{l=1}^{N_j}\sum_{i\in\mathcal{K}_j}\frac{l\Theta_i}{\theta_j}\delta_{l/\theta_j}^{\alpha}. \tag{7}$$

By combining Equations (6) and (7), we have

$$\min\left\{\mathbb{E}\left[\hat{n}_T^j(i)\right], \mathbb{E}\left[\hat{n}_T^j(i^*)\right]\right\} \leq \frac{8\alpha\log\delta^{-1}}{\Delta_i^2} + \mathbb{E}\left[Q\right]. \tag{8}$$

Then, we have

$$\mathbb{E}\left[\hat{n}_T^j(i^*)\right] \geq \frac{T\Theta_{i^*} - \mathbb{E}\left[Q\right]}{K} \geq \frac{\Theta_{i^*}n_T(i)}{\Theta_i K} - \frac{\mathbb{E}\left[Q\right]}{K}, \tag{9}$$

where the first inequality is based on the fact that the expected number of decision rounds with the optimal arm in the candidate set is at least $T\Theta_{i^*} - \mathbb{E}\left[Q\right]$, and the second one is based on the fact that $T \geq \hat{n}_T(i)/\Theta_i$. Combining the results in Equations (8) and (9), we get

$$\mathbb{E}\left[\hat{n}_T(i)\right] \leq \max\left\{\frac{8\alpha\log\delta^{-1}}{\Delta_i^2}, \frac{8\alpha K\Theta_i\log\delta^{-1}}{\Theta_{i^*}\Delta_i^2}\right\} + \left(1+\frac{\Theta_i}{\Theta_{i^*}}\right)\mathbb{E}\left[Q\right]. \tag{10}$$

Case II: regret due to external arm selection: Now, we aim at upper bounding the expected number of selection times for a suboptimal arm i by the agents outside set $\mathcal{A}_i$. Again, we assume that agent j makes a Type-I decision at time slot t. Consider the case that $I_t^j = i$ and i is not within $\mathcal{K}_j$. By algorithm rules, we have that arm i has the largest lower confidence bound. We claim that

$$2\sqrt{2}\texttt{cint}(i^*,j,t) \geq \Delta_i. \tag{11}$$

Otherwise, we have

$$\begin{aligned}\hat{\mu}(i,\hat{n}_t^j(i)) - \texttt{cint}(i,j,t) \leq& \mu(i) = \mu(i^*) - \Delta_i\\ <& \hat{\mu}(i^*,\hat{n}_t^j(i^*)) + \texttt{cint}(i^*,j,t) - 2\texttt{cint}(i^*,j,t)\\ =& \hat{\mu}(i^*,\hat{n}_t^j(i^*)) - \texttt{cint}(i^*,j,t),\end{aligned}$$

contradicting the rules of the algorithm. Thus, at time t, the selected arm i satisfies

$$\Delta_i \leq 2\sqrt{\frac{\alpha K\log\delta_t^{-1}}{t\Theta_{i^*} - Q}}.$$

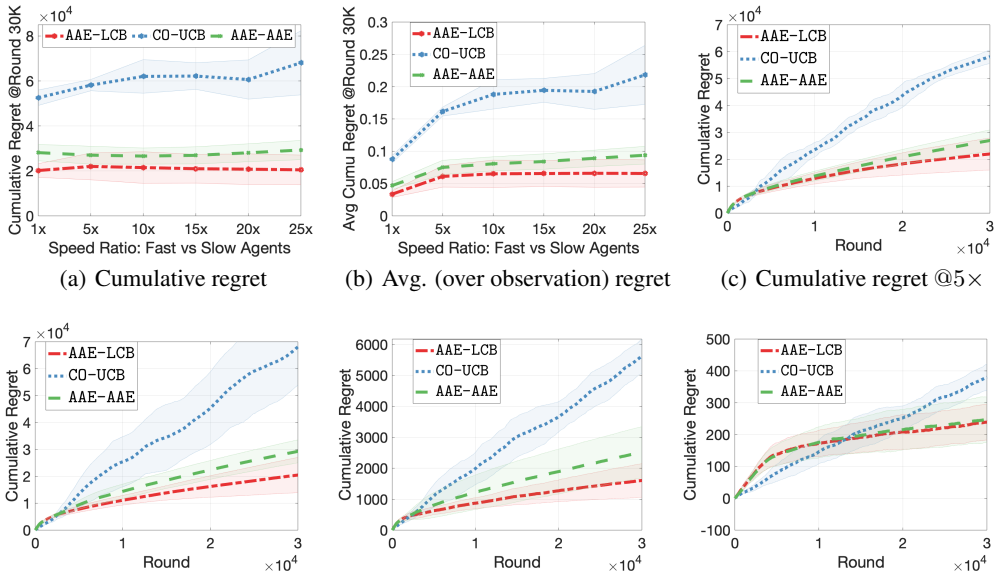
The above equation is obtained by replacing $\hat{n}_t^j(i^*)$ in equation $2\sqrt{2}\texttt{cint}(i^*,j,t) \geq \Delta_i$ with $[t\Theta_{i^*} - Q]/K$, since $\hat{n}_t^j(i^*) \geq [t\Theta_{i^*} - Q]/K$.

For any agent j, the largest time slot when the agent makes a Type-I decision and a suboptimal arm i lies in the candidate set is $4\frac{\alpha K\log\delta^{-1}}{\Delta_i^2\Theta_{i^*}} + \frac{Q}{\Theta_{i^*}}$. Then, the regret spent on the arm i in other agents is upper bounded by $4\frac{\alpha K\log\delta^{-1}}{\Delta_i}\frac{\Theta_i}{\Theta_{i^*}} + \frac{Q\Theta_i}{\Theta_{i^*}} + Q$.

Summing up the above two pieces of regret and the expected number of Type-II decisions yields

$$\begin{aligned}\mathbb{E}\left[R_T\right] \leq& \sum_{i:\Delta_i>0}\max\left\{\frac{4\alpha\log\delta^{-1}}{\Delta_i}\left(2+\frac{K\Theta_i}{\Theta_{i^*}}\right), \frac{12\Theta_i\alpha K\log\delta^{-1}}{\Delta_i\Theta_{i^*}}\right\}\\ &+ 2\left(1+\frac{\Theta_i}{\Theta_{i^*}}\right)\sum_{j\in\mathcal{A}}\sum_{l=1}^{N_j}\sum_{i\in\mathcal{K}_j}\frac{l\Theta_i}{\theta_j}\delta_{l/\theta_j}^{\alpha}.\end{aligned}$$

This completes the proof.



(a) Cumulative regret (b) Avg. (over observation) regret (c) Cumulative regret @5×

(d) Cumulative regret @25× (e) Per-agent regret (fast agent) (f) Per-agent regret (slow agent)

Figure 1: Regret of AAE-LCB vs. AAE-AAE vs. CO-UCB with two groups of "fast" and "slow" agents and varying action rate ratio between them. Notable observations: AAE-LCB outperforms CO-UCB significantly and outperforms AAE-AAE slightly. Comparing (c) and (d) shows that improvement of AAE-LCB over CO-UCB increases as the difference between action rates increases. Comparing (e) and (f) shows substantial regret degradation of CO-UCB in fast agent due to existence of slow agents.

5 Numerical Experiments

Our goal in this section is to numerically investigate the performance of AAE-LCB and compare it to that of AAE-AAE and CO-UCB (see Section 3.3), and show that AAE-LCB effectively resolves the challenge slow agents present, while neither AAE-AAE nor CO-UCB do so. More specifically, by comparing AAE-LCB with CO-UCB, our goal is to verify the importance of two-stage learning in the design of AAE-LCB, and by comparing AAE-LCB and AAE-AAE, our goal is to verify the importance of using LCB as the indexing policy for external arm selection.

Experimental setup. We assume there are $K = 100$ arms with Bernoulli rewards with average rewards uniformly randomly taken from *Ad-Clicks* [1]. In our experiments, we report the cumulative regret after 30,000 rounds, which corresponds to the number of decision rounds of the fastest agent. All reported values are averaged over 20 independent trials. We have 20 agents, each with 12 arms selected from among a set of $K = 100$ arms into two categories of 10 "fast" agents each with action rate of 1, and 10 "slow" agents with varying action rates of less than 1. We compare the performances of AAE-LCB and AAE-AAE and CO-UCB that are introduced in Section 3.3.

Experimental results. In Figures 1(a) and 1(b), we fix the action rate of the fast agents and vary the ratio between action rate of fast and slow agents from $5\times$ to $25\times$ with steps of 5. The results show relatively sound performance of AAE-LCB as the speed ratio between fast and slow agents increases. However, the performance of CO-UCB decreases substantially as shown in Figure 1(b) for average per-agent regret, which is mainly due to treating local and external arms in the same way. This further verifies our intuitions for the need to distinguish between local and external arms as the main motivation for developing AAE-LCB. AAE-AAE and AAE-LCB exhibit similar performance showing the importance of prioritizing the local arms, however, AAE-LCB performs slightly better since it uses a better external arm selection policy than that of AAE-AAE. The evolution of cumulative regrets over time in Figures 1(c) and 1(d) and per-agent regret for fast and slow agents in Figures 1(e) and 1(f) also demonstrate that AAE-LCB outperform the alternatives.

Note that the above experimental scenarios are designed to demonstrate the "average-case" performance, since rewards of arms are generated according to real data traces and all arms are randomly allocated to fast and slow agents. However, another important consideration on the efficiency of a learning algorithm is its performance in the worst case. From our theoretical results in Section 4, baseline algorithms suffers severe performance degradation and our algorithm outperforms the baseline algorithms a lot only when the optimal arm lies in a fast agent and there exists some arm with few observations. Specifically, the regret of the `AAE-AAE` algorithm degrades with the smallest aggregate action rate of agents containing arm i, $i \in \mathcal{K}$, i.e. $\Theta_{\min}$, while that of `AAE-LCB` depends on the action rate of the agent which the optimal arm lies in. To validate the efficiency of `AAE-LCB` in the worst case, we straightforwardly add a synthetic scenario and generate a special worst-case instance where the optimal arm only lies in fast agents. Specifically, we rank the arms in a descending order based on their reward means, and allocate a half of the arms with high reward means only to the fast agents and the other half only to the slow agents. The results are demonstrated in Figure 2, and shows as the ratio between the action rates of slow and fast agents increases, `AAE-LCB` outperforms `AAE-AAE` significantly.

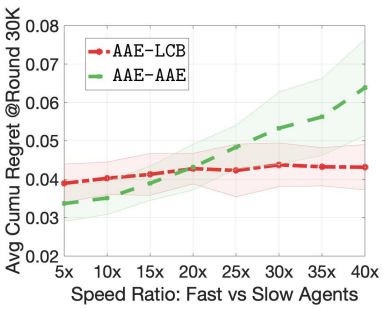


Figure 2: Performance comparison in the "worst" case

6 Conclusion and Future Directions

In this paper, we proposed bandit algorithms with near-optimal regret for a cooperative stochastic bandit problem among multiple agents playing the same instance of the problem, each with limited access to the set of arms and with different decision-making capabilities.

A limitation of this work is that in the current result, there is a gap between the regret of `AAE-LCB` and the regret lower bound. We leave developing an algorithm with the optimal regret for the `FC-CMA2B` setting as an open problem. Additionally, a critical, yet practically relevant question is how to extend the algorithms to the case with communication delay and cost between agents. We address the communication delay in the supplementary material of the paper. Regarding communication cost, the question is how to design algorithms that can provide a tradeoff between the regret and communication costs. While this tradeoff has been studied extensively in recent works [18, 21, 29, 50], none of them consider this tradeoff in the presence of heterogeneous agents. This task is challenging since in each decision-making round, each agent should make a nontrivial decision on whether or not broadcast with other agents or a subset of agents. Another question arises by considering topological constraints for agents. In this work, we focus our analysis to a set of fully-connected agents. In practice, however, geographically distributed agents might have limited access to other agents over an underlying graph. This setting also has been studied in recent works such as [36, 47]; however, there is no work on the heterogeneous version of this setting. Last, we do not see any negative societal impacts of our work.

Acknowledgments and Disclosure of Funding

Don Towsley acknowledges the support from U.S. Army Research Laboratory Army Research Laboratory under Cooperative Agreement W911NF-17-2-0196 (IoBT CRA). Lin Yang, Janice Yuzhen Chen, Stephen Pasteris and Don Towsley acknowledge the support from U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement W911NF-16-3-0001. Mohammad Hajiesmaili's research was supported by NSF CAREER 2045641, CNS 1908298, 2102963, and CPS 2136199. The work of John C.S. Lui was supported in part by the GRF 14200420 and SRFS2122-4202.

References

[1] Kaggle avito context ad clicks 2015. `https://www.kaggle.com/c/avito-context-ad-clicks`.

[2] S. Amani and C. Thrampoulidis. Decentralized multi-agent linear bandits with safety constraints. In *Proc. of AAAI*, 2021.

[3] 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.

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

[5] O. Avner and S. Mannor. Multi-user lax communications: a multi-armed bandit approach. In *Proc. of IEEE INFOCOM*, pages 1–9, 2016.

[6] B. Awerbuch and R. Kleinberg. Competitive collaborative learning. *Journal of Computer and System Sciences*, 74(8):1271–1288, 2008.

[7] Y. Bar-On and Y. Mansour. Individual regret in cooperative nonstochastic multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 3116–3126, 2019.

[8] D. Basu, C. Dimitrakakis, and A. Y. Tossou. Privacy in multi-armed bandits: Fundamental definitions and lower bounds. *arXiv:1905.12298v2*, 2019.

[9] L. Besson and E. Kaufmann. Multi-player bandits revisited. In *Algorithmic Learning Theory*, pages 56–92. PMLR, 2018.

[10] I. Bistritz and N. Bambos. Cooperative multi-player bandit optimization. *Advances in Neural Information Processing Systems*, 33, 2020.

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

[12] 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.

[13] E. Boursier, E. Kaufmann, A. Mehrabian, and V. Perchet. A practical algorithm for multiplayer bandits when arm means vary among players. In *AISTATS 2020*, 2020.

[14] E. Boursier and V. Perchet. SIC-MMAB: synchronisation involves communication in multi-player multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 12071–12080, 2019.

[15] G. Bresler, D. Shah, and L. F. Voloch. Collaborative filtering with low regret. In *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science*, pages 207–220, 2016.

[16] S. Bubeck, N. 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.

[17] S. Bubeck, Y. Li, Y. Peres, and M. Sellke. Non-stochastic multi-player multi-armed bandits: Optimal rate with collision information, sublinear without. In *Conference on Learning Theory*, pages 961–987, 2020.

[18] S. Buccapatnam, J. Tan, and L. Zhang. Information sharing in distributed stochastic bandits. In *Proc. of IEEE INFOCOM*, pages 2605–2613, 2015.

[19] N. Cesa-Bianchi, T. Cesari, and C. Monteleoni. Cooperative online learning: Keeping your neighbors updated. In *Algorithmic Learning Theory*, pages 234–250. PMLR, 2020.

[20] N. Cesa-Bianchi, C. Gentile, Y. Mansour, and A. Minora. Delay and cooperation in nonstochastic bandits. In *Conference on Learning Theory*, pages 605–622. PMLR, 2016.

[21] M. Chakraborty, K. Y. P. Chua, S. Das, and B. Juba. Coordinated versus decentralized exploration in multi-agent multi-armed bandits. In *IJCAI*, pages 164–170, 2017.

[22] C. Cortes, G. DeSalvo, C. Gentile, M. Mohri, and S. Yang. Online learning with sleeping experts and feedback graphs. In *International Conference on Machine Learning*, pages 1370–1378, 2019.

[23] S. J. Darak and M. K. Hanawal. Multi-player multi-armed bandits for stable allocation in heterogeneous ad-hoc networks. *IEEE Journal on Selected Areas in Communications*, 37(10):2350–2363, 2019.

[24] R. Della Vecchia and T. Cesari. An efficient algorithm for cooperative semi-bandits. In *Algorithmic Learning Theory*, pages 529–552. PMLR, 2021.

[25] A. Dubey and A. Pentland. Cooperative multi-agent bandits with heavy tails. In *Proc. of ICML*, 2020.

[26] A. Dubey and A. Pentland. Differentially-private federated linear bandits. In *Proc. of NeurIPS*, 2020.

[27] A. Dubey and A. Pentland. Kernel methods for cooperative multi-agent contextual bandits. 2020.

[28] E. Even-Dar, S. Mannor, and Y. Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7(Jun):1079–1105, 2006.

[29] E. Hillel, Z. S. Karnin, T. Koren, R. Lempel, and O. Somekh. Distributed exploration in multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 854–862, 2013.

[30] S. Ito, D. Hatano, H. Sumita, K. Takemura, T. Fukunaga, N. Kakimura, and K.-I. Kawarabayashi. Delay and cooperation in nonstochastic linear bandits. In *Proc. of NeurIPS*, 2020.

[31] J. Jiang, R. Das, G. Ananthanarayanan, P. A. Chou, V. Padmanabhan, V. Sekar, E. Dominique, M. Goliszewski, D. Kukoleca, R. Vafin, et al. Via: Improving internet telephony call quality using predictive relay selection. In *Proc. of ACM SIGCOMM*, pages 286–299, 2016.

[32] L. Jin, S. Li, L. Xiao, R. Lu, and B. Liao. Cooperative motion generation in a distributed network of redundant robot manipulators with noises. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(10):1715–1724, 2017.

[33] D. Kalathil, N. Nayyar, and R. Jain. Decentralized learning for multiplayer multiarmed bandits. *IEEE Transactions on Information Theory*, 60(4):2331–2345, 2014.

[34] V. Kanade, H. B. McMahan, and B. Bryan. Sleeping experts and bandits with stochastic action availability and adversarial rewards. In *Artificial Intelligence and Statistics*, pages 272–279, 2009.

[35] R. Kleinberg, A. Niculescu-Mizil, and Y. Sharma. Regret bounds for sleeping experts and bandits. *Machine learning*, 80(2-3):245–272, 2010.

[36] R. K. Kolla, K. Jagannathan, and A. Gopalan. Collaborative learning of stochastic bandits over a social network. *IEEE/ACM Transactions on Networking*, 26(4):1782–1795, 2018.

[37] P. Landgren, V. Srivastava, and N. E. Leonard. Distributed cooperative decision-making in multiarmed bandits: Frequentist and bayesian algorithms. In *Proc. of IEEE CDC*, pages 167–172, 2016.

[38] P. Landgren, V. Srivastava, and N. E. Leonard. Social imitation in cooperative multiarmed bandits: partition-based algorithms with strictly local information. In *Proc. of IEEE CDC*, pages 5239–5244, 2018.

[39] J. Langford and T. Zhang. The epoch-greedy algorithm for contextual multi-armed bandits. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, pages 817–824. Citeseer, 2007.

[40] F. Li, J. Liu, and B. Ji. Combinatorial sleeping bandits with fairness constraints. *IEEE Transactions on Network Science and Engineering*, 2019.

[41] S. Li, R. Kong, and Y. Guo. Cooperative distributed source seeking by multiple robots: Algorithms and experiments. *IEEE/ASME Transactions on mechatronics*, 19(6):1810–1820, 2014.

[42] K. Liu and Q. Zhao. Decentralized multi-armed bandit with multiple distributed players. In *Proc. of Information Theory and Applications Workshop (ITA)*, pages 1–10, 2010.

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

[44] D. Martínez-Rubio, V. Kanade, and P. Rebeschini. Decentralized cooperative stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 4529–4540, 2019.

[45] S. McQuade and C. Monteleoni. Global climate model tracking using geospatial neighborhoods. In *Proc. of AAAI*, 2012.

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

[47] A. Sankararaman, A. Ganesh, and S. 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.

[48] C. Shi and C. Shen. Federated multi-armed bandits. In *Proc. of AAAi*, 2021.

[49] A. Slivkins. Introduction to multi-armed bandits. *arXiv preprint arXiv:1904.07272*, 2019.

[50] B. Szorenyi, R. Busa-Fekete, I. Hegedus, R. Ormándi, M. Jelasity, and B. Kégl. Gossip-based distributed stochastic bandit algorithms. In *International Conference on Machine Learning*, pages 19–27, 2013.

[51] D. Vial, S. Shakkottai, and R. Srikant. Robust multi-agent multi-armed bandits. *arXiv preprint arXiv:2007.03812*, 2020.

[52] P.-A. Wang, A. Proutiere, K. Ariu, Y. Jedra, and A. Russo. Optimal algorithms for multiplayer multi-armed bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 4120–4129, 2020.

[53] W. Xia, T. Q. Quek, K. Guo, W. Wen, H. H. Yang, and H. Zhu. Multi-armed bandit based client scheduling for federated learning. *IEEE Transactions on Wireless Communications*, 2020.

[54] Z. Zhu, J. Zhu, J. Liu, and Y. Liu. Federated bandit: A gossiping approach. In *Proc. of ACM Sigmetrics*, 2021.

A Motivating Application Examples

There are two new features in our multi-agent bandit setting: asynchronous sampling and partial access to arms. The asynchronous sampling has clear practical motivation due to the nature of multi-agent (or distributed) decision-making. In the following, we focus on the justification of practical relevance for the latter on partial access to arms. We highlight three different examples in the context of the online shortest path routing (OSPR), clinical trials, and crowdsourcing, which are classical applications of bandits. Indeed, our current model does not fully captures the following examples; instead, it provides initial modeling with interesting and nontrivial mathematical challenges that are addressed in this paper.

OSPR. In the basic version of OSPR, a bandit algorithm could be applied to select a path (arm) with minimum delay. Now consider an extended version of OSPR in a heterogeneous network that includes multiple virtual private networks (VPN) each represented by an agent (or a gateway, in networking terminology). In this scenario, the local arms of an agent represent the paths including the link in their VPN, and the paths in other VPNs are external arms. In the case of selecting a path that goes through some nodes in another VPN, the path information might not be observable to the rest of the network, which resides outside of the VPN, i.e., other agents. This scenario could be captured by our bandit setting in which an external arm (a path in another VPN) is selected and the reward is allocated (the delay), but it is not observable. On the other hand, when an agent selects a path in its local VPN, they can share the information with the representative agents of other VPNs to eventually find the globally shortest path in the entire network.

Cooperative clinical trial. A doctor in a clinic is required to offer a treatment plan for patients with some disease, while the clinic, which the doctor sits in, can only provide partial options due to lack of medical equipment. The goal of the doctor is to maximize his reputation or the number of treatment successes. In the medical decision-making setting, the plan for the disease corresponds to the arm, and the doctor in the clinic is an agent. If the doctor decides on a specific treatment option that is not offered in their clinic, the patient may not come back for follow-up appointments, and in this case, the patient's medical record or feedback might be missing. Thus, the treatment plans offered by the clinic are the local arms in `FC-CMA2B`. Cooperation among the doctors in different clinics helps to accelerate learning the best treatment plan, by sharing medical statistics.

Crowdsourcing. In crowdsourcing, workers are allowed to register to any agents, each of which is tailored for specific tasks. An agent maintains the profile of registered workers, with a reputation score (e.g., the success rate of finishing a job) updated with their performance in satisfying tasks. In real-world applications, workers are allowed to be anonymous to some agents with the aim to protect personal privacy, with only the reputation score revealed to the agent. For anonymous workers, the agent treats them as guests and no identity or profile can be tracked for them. Hence, the observed reward is useless. In crowdsourcing with multiple agents, anonymous workers correspond to external arms, and registered workers correspond to local arms. In this model, an agent selects anonymous workers only based on reputation scores by other agents. Implicitly, different agents cooperate by maintaining the common reputation score for registered workers.

It is worth noting that, to the best of our knowledge, `FC-CMA2B` is the first model that tackles a multi-agent bandit setting where agents have access to a subset of arms. The basic setting that we considered in this paper, however, could be extended to better capture more convincing practical applications. We highlight a few practically relevant, and (to our belief) feasible extensions. First, our model could be extended to the case that instead of exact information, a perturbed, yet useful observation is communicated between agents. This extension makes the setting much more interesting from a practical perspective since in this setting agents can share some limited information with others to incentivize them to use their local resources (arms). Another practical extension is adding the communication delay in the model as we tackled it in the Appendix E. One can consider communication costs in the model and the goal becomes to provide low regret algorithms with low communication costs.

B Extensive Literature Review

In addition to cooperative stochastic bandits, the cooperative nonstochastic bandits were introduced in [6] with the latest results in [20, 7, 30]. Specifically, [6] introduces the cooperation setting where agents share the distribution over actions without delay. They bound the average regret for the case that some of the agents are dishonest and behave in an arbitrary Byzantine manner. [20] studies the tradeoff between cooperation and delay, which is ruled by the underlying communication network topology, and proves a bound for the average regret. The authors in [7] prove an individual regret bound that holds simultaneously for all agents, solving an open problem left in [20].

Our work is different from all above related literature, since we consider heterogeneous agents each with access to a subset of arms and different decision capabilities. Besides, we consider fully-connected agents, while most of prior work on cooperative bandits considers an underlying graph connecting the agents. Extension of our result to the case with topology constraints and developing efficient communication protocol among agents are interesting future directions.

We note that asynchronous online learning is getting attention in recent works such as [19, 22]. Specifically, [19] considers a model, in which there is a network of agents, and in each round some of the agents are activated to make decisions. However, their model assumes full information feedback and ours is dedicated to the bandit setting. In the context of bandit setting, the asynchronous nature of our work is related to the category of sleeping bandits [40, 35, 34], in which some arms could be "sleeping" or "unavailable" at some rounds. In these models, typically the set of available arms changes without following any structure. In contrast, in our model, all arms are always available, but there is a structured limitation on the observability of feedback dictated by the local subset and action rate of agents. Note that the asynchronous nature our work considers is also different from the one in [14], which concerns the learning horizon of agents, i.e., whether an agent joins the game at the beginning, while we study the decision-making rates of agents.

Last, we note that in literature there is multi-agent bandits with competition. In this model, when an agent selects an arm, it collects the reward only if no other agent pulls the same arm. Also, in some cases, the reward of an arm will be degraded if it is being pulled by multiple agents. This model is motivated by the application of distributed channel selection in wireless cognitive radio systems. Several variants of this model have been studied in recent works, e.g., [3, 14, 17, 52, 11, 13, 43, 42, 9]. This line of work stands in clear contrast to our work, since we focus on the cooperative version of multi-agent bandit problems where rewards are independently collected across agents.

C Summary of Notations

We list all notations used in this paper in Table 1.

D Supplementary Proofs and Analysis

In the first subsection, we first provide details on the derivation of the confidence intervals in `AAE-LCB`, which will be used in our later proofs.

D.1 Analysis of Confidence Intervals in `AAE`

Let $X_{i,s},\ s = 1, 2, \ldots, n$, be the random variable by which the s-th observed reward for arm $i \in \mathcal{S}$ is generated. For any positive a, we have

$$\begin{aligned}
\Pr\left\{\hat{\mu}_{i,n} \geq \mu_i + a\right\} \leq & \Pr\left\{\frac{1}{n}\sum_{s=1}^{n}[X_{i,s} - \mu_i] \geq a\right\} \\
= & \Pr\left\{\exp\left(\theta\sum_{s=1}^{n}[X_{i,s} - \mu_i]\right) \geq \exp\left(\theta a n\right)\right\} \\
\leq & \mathbb{E}\left\{\exp\left(\theta\sum_{s=1}^{n}[X_{i,s} - \mu_i] - \theta a n\right)\right\} \\
= & \exp(-n\theta a)\mathbb{E}\left\{\exp(\theta[X_{i,s} - \mu_i])\right\}^n,
\end{aligned}$$

Table 1: Summary of notations related to FC-CMA2B

Notation	Description
t	Index of time slot
T	The number of time slots
K	The number of arms
M	The number of agents
$\mathcal{K}$	Set of all arms
$\mathcal{A}$	Set of agents
$\mathcal{K}_j$	Local subset of arms for agent j with observable reward
$\mathcal{A}_i$	Set of agents containing arm i
θ_j	Action rate of agent j
w_j	Gap of adjacent decision rounds of agent j
N_j	Total number of decisions made by agent j
Θ_i	Aggregate action rate of agents containing arm i
Θ	Aggregate action rate of all agents
$x_t(i)$	Reward of arm i at time slot t
$\mu(i)$	Mean reward of arm i
$\Delta(i, i')$	Difference of mean rewards between arm i and i'
Δ_i	Difference of mean rewards between the optimal arm and arm i
I_t^j	Chosen arm by agent j at t
δ_t	Algorithm parameter used in AAE-LCB and AAE
δ	The minimum value for δ_t over the entire time horizon
$n_t(i)$	Total number of observations on arm i up to t
$\hat{n}_t^j(i)$	Number of observations on arm i available to agent j up to t
$\hat{\mu}(i, n)$	Empirical reward mean of arm i with n observations
$\texttt{cint}(i, j, t)$	Width of the confidence interval for arm i and agent j at time t
$\mathcal{C}_{j,t}$	Candidate set of agent j defined in AAE-LCB and AAE-AAE
R_T	Regret over T time slots
R_T^j	Individual regret of agent j over time horizon T
d_j	The largest delay from any agent to agent j
D	The largest delay between any two agents
$\mathrm{KL}(a, b)$	The Kullback-Leibler divergence between a Bernoulli of parameter a and b
$\mathrm{KL}(\mathbb{P}_1, \mathbb{P}_2)$	The Kullback-Leibler divergence between two random distributions $\mathbb{P}_1$ and $\mathbb{P}_2$

where θ can be any positive. The inequality bases on Chebyshev's inequality.

Let $\phi(\theta) = \log \mathbb{E}\left\{\exp(\theta[X_{i,s} - \mu_i])\right\}$, we have

$$\begin{aligned}\Pr\left\{\hat{\mu}_{i,n} \geq \mu_i + a\right\} &\leq \exp(-n\theta a)\exp(n\phi(\theta)) \\ &\leq \inf_{\theta} \exp(-n(a\theta - \phi(\theta)) \\ &= \exp(-n\phi^*(a)).\end{aligned}$$

where $\phi^*(a)$ is defined as $\sup_\theta (a\theta - \phi(\theta))$.

Replacing a with $(\phi^*)^{-1}\left(\frac{\log(1/\delta)}{n}\right)$, we have

$$\Pr\left\{\hat{\mu}_{j,n} \geq \mu_j + a\right\} \leq \exp(-n\phi^*(a)) = \exp\left(-n\frac{\log(1/\delta)}{n}\right) = \delta.$$

In summary,

$$\Pr\left\{\hat{\mu}_{j,n} \geq \mu_j + (\phi^*)^{-1}\left(\frac{\log(1/\delta)}{n}\right)\right\} \leq \delta.$$

Similarly, we can derive the probability for $\hat{\mu}_{j,n} \leq \mu_j + (\phi^*)^{-1}\left(\frac{\log(1/\delta)}{n}\right)$.

For the bounded random variable $X_{i,s}$, we have

$$(\phi^*)^{-1}\left(\frac{\log(1/\delta)}{n}\right) = \sqrt{\frac{\log(1/\delta)}{2n}}.$$

To construct the confidence intervals, we set the confidence probability as $\delta = 1/n^\alpha$. Accordingly, the upper/lower confidence bound for the i-th arm is $\hat{\mu}_{i,s} + \sqrt{\frac{\alpha \log(n)}{2n}}$ and $\hat{\mu}_{i,s} - \sqrt{\frac{\alpha \log(n)}{2n}}$.

D.2 A Proof for Lemma 1

Based on the results in D.1, for any arm i with n observations, we have

$$\Pr\left(\mu(i) > \hat{\mu}(i,n) + \sqrt{\frac{\alpha \log \delta^{-1}}{2n}}\right) \leq \delta^\alpha.$$

Then, by applying the above inequality to the decision of agent j at time $t = l/\theta_j$, we get (12).

$$\begin{aligned} &\Pr\left(\mu(i) > \hat{\mu}\left(i, \hat{n}^j_{l/\theta_j}(i)\right) + \sqrt{\frac{\alpha \log \delta^{-1}_{l/\theta_j}}{2\hat{n}^j_{l/\theta_j}(i)}}\right) \\ \leq &\sum_{s=1}^{l\Theta_i/\theta_j} \Pr\left(\mu(i) > \hat{\mu}(i,s) + \sqrt{\frac{\alpha \log \delta^{-1}_{l/\theta_j}}{2s}}\right) \leq \frac{l\Theta_i}{\theta_j}\delta^\alpha_{l/\theta_j}. \end{aligned} \tag{12}$$

The above equation shows that the probability that the true mean value of arm i is above the upper confidence bound in agent j at time l/θ_j is not larger than $\frac{l\Theta_i}{\theta_j}\delta^\alpha_{l/\theta_j}$. Similarly, for the lower confidence interval we have

$$\Pr\left(\mu(i) < \hat{\mu}\left(i, \hat{n}^j_{l/\theta_j}(i)\right) - \sqrt{\frac{\alpha \log \delta^{-1}_{l/\theta_j}}{2\hat{n}^j_{l/\theta_j}(i)}}\right) \leq \frac{l\Theta_i}{\theta_j}\delta^\alpha_{l/\theta_j}.$$

Thus, the probability that the mean value of any arm in $\mathcal{K}_j$ at time l/θ_j lies in the confidence interval is lower bounded by

$$1 - 2\sum_{i \in \mathcal{K}_j} \frac{l\Theta_i}{\theta_j}\delta^\alpha_{l/\theta_j}.$$

This completes the proof.

D.3 A Proof for Theorem 1

Theorem 1 can be proved in two steps.

(1) In the first step, we prove the lower bound without considering the influence of Θ/Θ_{i^*}. To do that, we assume the action rate of each agent is a constant. The techniques for the proof of the lower bound in this case have been investigated extensively. For the completion of analysis, we provide the details as follows. Let us define $\mathcal{E}_K$ as the class of K-armed stochastic bandits where each arm has a Bernoulli reward distribution. Assume that policy π is consistent over $\mathcal{E}_K$, i.e., for any bandit problem $\nu \in \mathcal{E}_K$ and any $\sigma' > 0$, whose regret satisfies

$$R_T(\pi, \nu) = O((T\Theta)^{\sigma'}), \text{ as } T \to +\infty.$$

Let $\nu = [P_1, P_2, \ldots, P_K]$ and $\nu' = [P'_1, P'_2, \ldots, P'_K]$ be two reward distributions such that $P_k = P'_k$ except for $k = i$. Specifically, we choose $P'_i = \mathcal{N}(\mu_i + \lambda)$ and $\lambda > \Delta_i$. For stochastic bandits, we have the following divergence decomposition equation (one can refer to [8] for more details).

$$\mathrm{KL}(\mathbb{P}_{\nu,\pi}, \mathbb{P}_{\nu',\pi}) = \mathbb{E}_{\nu,\pi}\left[n_T(i)\right] \mathrm{KL}(P_i, P'_i),$$

where $\mathbb{P}_{\nu,\pi}$ is the distribution of T-round action-reward histories induced by the interconnection between policy π and the environment ν, and $\mathrm{KL}(\mathbb{P}_{\nu,\pi}, \mathbb{P}_{\nu',\pi})$ measures the relative entropy between $\mathbb{P}_{\nu,\pi}$ and $\mathbb{P}_{\nu',\pi}$.

In addition, from the high-probability Pinsker inequality, we have

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

where A is any event defined over $\mathbb{P}_{\nu,\pi}$ and $\mathbb{P}_{\nu',\pi}$. By definition, the regret of policy π over ν and ν' satisfies

$$R_T(\nu,\pi) \geq \frac{T\Delta_i}{2}\mathbb{P}_{\nu,\pi}\left(n_T(i) \geq \frac{T\Theta}{2}\right),$$

and

$$R_T(\nu',\pi) \geq \frac{T(\lambda-\Delta_i)}{2}\mathbb{P}_{\nu',\pi}\left(n_T(i) < \frac{T\Theta}{2}\right).$$

The above equation bases on the fact that the suboptimality gaps in ν' is larger than $\lambda - \Delta_i$.

Concluding the above two equations and lower bounding Δ_i and $(\lambda - \Delta_i)/2$ by $\kappa(\Delta_i, \lambda) := \min\{\Delta_i, \lambda - \Delta_i\}/2$ yields

$$\mathbb{P}_{\nu,\pi}\left(n_T(i) \geq \frac{T\Theta}{2}\right) + \mathbb{P}_{\nu',\pi}\left(n_T(i) < \frac{T\Theta}{2}\right) \leq \frac{R_T(\nu,\pi) + R_T(\nu',\pi)}{\kappa(\Delta_i,\lambda)T}.$$

We have

$$\begin{aligned}
\mathrm{KL}(P_i, P_i')\mathbb{E}_{\nu,\pi}\left[n_T(i)\right] \geq& \log\left(\frac{\kappa(\Delta_i,\lambda)}{2}\frac{T\Theta}{R_T(\nu,\pi)+R_T(\nu',\pi)}\right) \\
=& \log(\frac{\kappa(\Delta_i,\lambda)}{2}) + \log(T\Theta) - \log(R_T(\nu,\pi)+R_T(\nu',\pi)) \\
\geq& \log(\frac{\kappa(\Delta_i,\lambda)}{2}) + (1-\sigma')\log(T\Theta) + C,
\end{aligned}$$

where C is a constant. The last inequality is based on the assumption that the algorithm is consistent. Taking $\lambda = \Delta_i$, for large T, we can lower bound the regret of any consistent policy π as follows:

$$\liminf_{T\to+\infty}\frac{R_T}{\log(T\Theta)} \geq \liminf_{T\to+\infty}\frac{\sum_i \mathbb{E}_{\nu,\pi}\left[n_T(i)\right]\Delta_i}{\log(T\Theta)} = O(\sum_i \frac{\Delta_i}{\mathrm{KL}(P_i, P_i')}).$$

(2) In the second step, we proceed to prove that the regret lower bound has a further asymptotically linear dependency on $(\Theta/\Theta_{i^*})^\sigma$ for any $0 < \sigma < 1$. We prove this by contradiction: assume that $0 < \sigma'' < 1$ exists and the regret of some algorithm has an asymptotically linear dependency on $(\Theta/\Theta_{i^*})^{\sigma''}$. That is

$$\limsup_{T\to+\infty, \frac{\Theta}{\Theta_{i^*}}\to+\infty}\frac{R_T(\pi,\nu)}{(\Theta/\Theta_{i^*})^{\sigma''}\log(T\Theta)} = O(\sum_i \frac{\Delta_i}{\mathrm{KL}(P_i, P_i')}). \tag{13}$$

By similar reasoning, we have

$$\mathrm{KL}(\mathbb{P}_{\nu,\pi}, \mathbb{P}_{\nu',\pi}) \geq \log \frac{1}{2\left(\mathbb{P}_{\nu,\pi}(A) + \mathbb{P}_{\nu',\pi}(A^c)\right)}, \tag{14}$$

Redefine A as event $n_T(i^*) < (T\Theta)/2$. Similarly, we have

$$R_T(\nu,\pi) \geq \frac{T\Delta_i}{2}\mathbb{P}_{\nu,\pi}\left(n_T(i^*) < \frac{T\Theta}{2}\right),$$

and

$$R_T(\nu',\pi) \geq \frac{T(\lambda-\Delta_i)}{2}\mathbb{P}_{\nu',\pi}\left(n_T(i^*) \geq \frac{T\Theta}{2}\right).$$

Then, it follows from Equation (14) that

$$\begin{aligned}\mathrm{KL}(P_i, P_i')\mathbb{E}_{\nu,\pi}\left[n_T(i^*)\right] &\geq \log(\frac{\kappa(\Delta_i,\lambda)}{2}) + \log(T\Theta) - \log(R_T(\nu,\pi) + R_T(\nu',\pi)) \\ &\geq \log(\frac{\kappa(\Delta_i,\lambda)}{2}) + \log(T\Theta) - \sigma'' \log(\frac{\Theta}{\Theta_{i^*}}) - \log\log(T\Theta) + C \\ &\geq \log(\frac{\kappa(\Delta_i,\lambda)}{2}) + (1-\sigma'')\log(T\Theta) - \log\log(T\Theta) + C,\end{aligned}$$

where the last inequality uses the fact that $\Theta_{i^*} > 1/T$.

And, when $\Theta/\Theta_{i^*} \to +\infty$, we will have

$$T\Theta_{i^*} < \frac{1}{\mathrm{KL}(P_i, P_i')}((1-\sigma'')\log(T\Theta) - \log\log(T\Theta)).$$

Thus, the above inequality will not hold, since $T\Theta_{i^*} \geq \mathbb{E}_{\nu,\pi}\left[n_T(i^*)\right]$. This contradicts the consistency condition in Equation (13). That means, for given T, any algorithm incurs a linear regret with respect to $(\Theta/\Theta_{i^*})^\sigma$, for any $0 < \sigma < 1$, when Θ/Θ_{i^*} is large enough. We conclude our results below.

$$\liminf_{T\to+\infty,\ \frac{\Theta}{\Theta_{i^*}}\to+\infty} \frac{R_T}{(\Theta/\Theta_{i^*})^\sigma \log(T\Theta)} = \Omega\left(\sum_i \frac{\Delta_i}{\mathrm{KL}(P_i, P_i')}\right), \text{ for any } 0 < \sigma < 1.$$

This completes the proof.

D.4 The Regret Analysis of `AAE-AAE`

Last, we provide analysis on the regret result of the baseline algorithm `AAE-AAE` in Equation (4).

Consider a scenario where the local subset of each agent only contains one different arm. We assume there exists a slow agent with the smallest action rate containing a suboptimal arm $\tilde{i}$. Since there is no delay between agents, the empirical mean values and confidence intervals for arms by different agents is the same. In the following equation, we use the results in Appendix D.1 to calculate the probability that there exists some arm whose mean value is above its confidence interval of width $\frac{1}{2}\sqrt{\frac{\alpha\log T}{2\hat{n}^j_{l/\theta_j}(i)}}$, i.e., for any j, we have

$$\begin{aligned}&\Pr\left(\exists i, j, l : \mu(i) > \hat{\mu}\left(i, \hat{n}^j_{l/\theta_j}(i)\right) + \frac{1}{2}\sqrt{\frac{\alpha\log T}{2\hat{n}^j_{l/\theta_j}(i)}}\right) \\ &\quad= \Pr\left(\exists i, l = 1, 2, \ldots, N_j : \mu(i) > \hat{\mu}\left(i, \hat{n}^j_{l/\theta_j}(i)\right) + \frac{1}{2}\sqrt{\frac{\alpha\log T}{2\hat{n}^j_{l/\theta_j}(i)}}\right) \\ &\quad\leq \sum_{i\in\mathcal{K}}\sum_{s=1}^{T} \Pr\left(\mu(i) > \hat{\mu}(i,s) + \frac{1}{2}\sqrt{\frac{\alpha\log T}{2s}}\right) \\ &\quad\leq \sum_{i\in\mathcal{K}} \frac{1}{T^{0.25\alpha}} \leq \frac{K}{T^{0.25\alpha-1}}.\end{aligned}$$

The first inequality uses the fact that there is at most T observations for each arm since each agent only contains one arm. Thus, the probability that the mean values of all arms lie in the confidence interval of width $\frac{1}{2}\sqrt{\frac{\alpha\log T}{2\hat{n}^j_{l/\theta_j}(i)}}$ is at least $1 - 2K/(T^{0.25\alpha-1})$.

We assume the mean values of all arms lie in the confidence interval of width $\frac{1}{2}\sqrt{\frac{\alpha\log T}{2\hat{n}^j_{l/\theta_j}(i)}}$. At any time slot t, arm $\tilde{i}$ will not be eliminated if

$$\frac{1}{2}\sqrt{\frac{\alpha\log T}{2n_t(\tilde{i})}} \geq \Delta_{\tilde{i}}$$

Thus, under the above assumption, the number of observations needed to eliminate arm $\tilde{i}$ in the slow agent is at least

$$\frac{\alpha \log T}{8\Delta_{\tilde{i}}^2}.$$

Combining with the probability that the above assumption holds, we can provide a lower bound for the expected number of observations needed to eliminate arm i in the slow agent as follows.

$$\frac{\alpha \log T}{8\Delta_{\tilde{i}}^2}\left(1-\frac{2K}{T^{0.25\alpha-1}}\right).$$

In other words, the expected number of mistakes made by the system can be as large as

$$\Omega\left(\frac{\Theta}{\Theta_{\min}}\frac{\alpha \log T}{8\Delta_{\tilde{i}}^2}\left(1-\frac{2K}{T^{0.25\alpha-1}}\right)\right).$$

Thus, the expected regret is at least $\Omega\left(\frac{\Theta}{\Theta_{\min}}\log T\right)$. This completes the proof.

E Extension to Multi-Agent Bandits with Delays

The goal of `FC-CMA2B` is to capture the heterogeneity in the action rates of different agents. The basic `FC-CMA2B` model introduced in this paper, however, can be extended to capture several additional practically relevant features, such as the cost of cooperation, delays in broadcasting observations, topology constraints, and malicious agents, each of which presents different additional challenges.

Here we extend our results to the case where there are communication delays between agents. Delays between agents are measured in units of decision rounds. We define d_j to be the largest delay from any agent to agent j with $D = \max_j d_j$. The delay between agents can be interpreted as the case in which agents are located on an underlying connected graph and the cooperation could be done by routing over a network. Then, assuming links with unit delay, the delay d_j is the longest path from any agent to agent j and D is the diameter of the graph.

The `AAE-LCB` algorithm can be directly applied to the above case without any rule change. The regret analysis of `AAE-LCB` for `FC-CMA2B` with delays, however, needs to account for delays. The following result shows that `AAE-LCB` attains a regret with a linear dependency on the maximum delay parameter D.

Theorem 3 (Expected Regret of `AAE-LCB` under `FC-CMA2B` with Delay) *The expected regret of* `AAE-LCB` *has the following upper bound,*

$$\mathbb{E}\left[R_T\right] \leq \sum_{i:\Delta_i>0} \max\left\{\frac{4\alpha\log\delta^{-1}}{\Delta_i}\left(2+\frac{K\Theta_i}{\Theta_{i^*}}\right)+F_i\Delta_i, \frac{12\Theta_i\alpha K\log\delta^{-1}}{\Delta_i\Theta_{i^*}}+KD\Theta_i\Delta_i\right\}$$
$$+2\left(1+\frac{\Theta_i}{\Theta_{i^*}}\right)\sum_{j\in\mathcal{A}}\sum_{l=1}^{N_j}\sum_{i\in\mathcal{K}_j}\frac{l\Theta_i}{\theta_j}\delta_{l/\theta_j}^{\alpha}+D\Theta,$$

where

$$F_i := \sum_{j\in\mathcal{A}_i}\min\left\{d_j\theta_j, \frac{8\alpha\log\delta^{-1}}{\Delta_i^2}\right\}.$$

The proof of Theorem 3 follows the same logic flow as the proof of Theorem 2, and given later in this section.

Comparing the regret bounds in Theorems 2 and 3, the new one includes additional terms that are linear in the delays. Specifically, as D increases to T, the regret will be linear, which is consistent with the fact that algorithms for `FC-CMA2B` suffers a linear regret when agents are totally separated.

We also briefly examine the impact of delay. Toward this, we consider three additional scenarios with average delays of 1000, 3000 and 5000 slots. Specifically, for average 1000 delay case, the mean

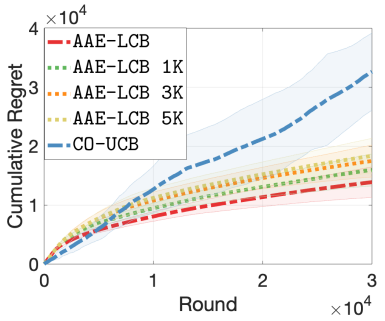


Figure 3: Regret of AAE-LCB with different delay values

delay between agent j and j', $d_{j,j'}$, is uniformly randomly picked from $[1000-10, 1000+10]$; and at each time slot, the exact delay is taken uniformly randomly from $[d_{j,j'}-2, d_{j,j'}+2]$. In Figure E, we report the evolution of cumulative regret of AAE-LCB and CO-UCB without any delay as well as AAE-LCB with delays. with $K=100$, $M=10$, 30 arms per agent. The results shows the regret of our algorithm for FC-CMA2B increases as the delay increases.

E.1 A Proof for Theorem 3

The proof of Theorem 3 mainly follows that of Theorem 2 except incorporating delays in the analysis.

Case I: local arm selection: For a sub-optimal arm i, we first upper bound the number of observations made by agents in $\mathcal{A}_i$, which is $n_t(i)$. Also, we reuse $\hat{n}_t^j(i)$ as the total number of observations received by agents j up to t. Then, we have

$$n_t(i) \leq \hat{n}_t^j(i) + \sum_{j \in \mathcal{A}_i} \min\left\{d_j\theta_j, n_t^j(i)\right\}, \ \forall\, j \in \mathcal{A}_i, \tag{15}$$

where on the right hand side, the second term refers to an upper bound of the number of outstanding observations, i.e., the observations that have not been received due to the delay between agents and $n_t^j(i)$ is number of observations maded by agent j. We also consider two types of decisions: Type-I corresponds to the decisions of an agent when the mean values of all arms lie in the confidence intervals calculated by the agent; and Type-II decisions refer to others. First, we focus on the cases that the algorithm makes a Type-I decision at time t, i.e., the mean value of any arm lies in its confidence interval calculated by agent j. Then, at time t, if agent j in $\mathcal{A}_i$ selects arm i, we have

$$\sqrt{\frac{2\alpha\log\delta_t^{-1}}{\hat{n}_t^j(i)}} + \sqrt{\frac{2\alpha\log\delta_t^{-1}}{\hat{n}_t^j(i^*)}} \geq \Delta_i. \tag{16}$$

Otherwise, there is

$$\begin{aligned}\hat{\mu}(i,\hat{n}_t^j(i)) + \sqrt{\frac{\alpha\log\delta_t^{-1}}{2\hat{n}_t^j(i)}} &\leq \mu_i + 2\sqrt{\frac{\alpha\log\delta_t^{-1}}{2\hat{n}_t^j(i)}} < \mu_i + \Delta_i - \sqrt{\frac{2\alpha\log\delta_t^{-1}}{\hat{n}_t^j(i^*)}} \\ = \mu_{i^*} - \sqrt{\frac{2\alpha\log\delta_t^{-1}}{\hat{n}_t^j(i^*)}} &\leq \hat{\mu}(i^*,\hat{n}_t^j(i^*)) - \sqrt{\frac{\alpha\log\delta_t^{-1}}{2\hat{n}_t^j(i^*)}},\end{aligned}$$

implying that arm i is strictly dominated by i^*, hence, i can not be selected by agent j, contradicting the assumption that i is selected by j. It follows from Equation (16) that

$$\max\left\{\sqrt{\frac{2\alpha\log\delta_t^{-1}}{\hat{n}_t^j(i)}}, \sqrt{\frac{2\alpha\log\delta_t^{-1}}{\hat{n}_t^j(i^*)}}\right\} \geq \frac{\Delta_i}{2}.$$

Thus, we have

$$\min\left\{\hat{n}_t^j(i), \hat{n}_t^j(i^*)\right\} \leq \frac{8\alpha\log\delta_t^{-1}}{\Delta_i^2}. \tag{17}$$

Again, we define Q as the number of Type-II decisions. Then, by Lemma 1 that is still valid for the delayed system, we have

$$\mathbb{E}[Q] \leq 2 \sum_{j \in \mathcal{A}} \sum_{l=1}^{N_j} \sum_{i \in \mathcal{K}_j} \frac{l\Theta_i}{\theta_j} \delta_{l/\theta_j}^{\alpha}. \tag{18}$$

By combining Equations (17) and (18), we have

$$\min\left\{\mathbb{E}\left[\hat{n}_T^j(i)\right], \mathbb{E}\left[\hat{n}_T^j(i^*)\right]\right\} \leq \frac{8\alpha \log \delta^{-1}}{\Delta_i^2} + \mathbb{E}[Q]. \tag{19}$$

Then, using Equation (15), we have

$$\begin{aligned} \mathbb{E}\left[\hat{n}_T^j(i^*)\right] &\geq \mathbb{E}[n_T(i^*)] - \sum_{j \in \mathcal{A}_{i*}} d_j\theta_j \geq \frac{T\Theta_{i^*} - \mathbb{E}[Q]}{K} - \sum_{j \in \mathcal{A}_{i*}} d_j\theta_j \\ &\geq \frac{\Theta_{i^*} n_T(i)}{\Theta_i K} - \frac{\mathbb{E}[Q]}{K} - \sum_{j \in \mathcal{A}_{i^*}} d_j\theta_j, \end{aligned} \tag{20}$$

where the second inequality is based on the fact that the expected number of decision rounds with the optimal arm in the candidate set is at least $T\Theta_{i^*} - \mathbb{E}[Q]$, and the third inequality is based on the fact that $T \geq n_T(i)/\Theta_i$.

Last, combining the results in Equations (15), (19), and (20), we get

$$\begin{aligned} \mathbb{E}[n_T(i)] &\leq \max\left\{\frac{8\alpha \log \delta^{-1}}{\Delta_i^2} + F_i, \frac{8\alpha K\Theta_i \log \delta^{-1}}{\Theta_{i^*}\Delta_i^2} + K\frac{\Theta_i}{\Theta_{i^*}} \sum_{j \in \mathcal{A}_{i^*}} d_j\theta_j\right\} + \left(1 + \frac{\Theta_i}{\Theta_{i^*}}\right)\mathbb{E}[Q] \\ &\leq \max\left\{\frac{8\alpha \log \delta^{-1}}{\Delta_i^2} + F_i, \frac{8\alpha K\Theta_i \log \delta^{-1}}{\Theta_{i^*}\Delta_i^2} + KD\Theta_i\right\} + \left(1 + \frac{\Theta_i}{\Theta_{i^*}}\right)\mathbb{E}[Q], \end{aligned}$$

where

$$F_i = \sum_{j \in \mathcal{A}_i} \min\left\{d_j\theta_j, \frac{8\alpha \log \delta^{-1}}{\Delta_i^2}\right\}.$$

Case II: external arm selection: Now, we aim at upper bounding the expected number of selection times for a suboptimal arm i by the agents outside set $\mathcal{A}_i$. Again, we assume that agent j makes a Type-I decision at time slot t. Consider the case that $I_t^j = i$ and i is not within $\mathcal{K}_j$. By algorithm rules, we have that arm i has the largest lower confidence bound. We prove that the following inequality must hold in this case.

$$\sqrt{\frac{\alpha \log \delta_t^{-1}}{\hat{n}_t^j(i^*)}} \geq \frac{1}{2}\Delta_i, \tag{21}$$

Otherwise, we have

$$\begin{aligned} \hat{\mu}(i, \hat{n}_t^j(i)) - \sqrt{\frac{\alpha \log_t \delta_t^{-1}}{2\hat{n}_t^j(i)}} &\leq \mu(i) = \mu(i^*) - \Delta_i \\ &< \hat{\mu}(i^*, \hat{n}_t^j(i^*)) + \sqrt{\frac{\alpha \log \delta_t^{-1}}{2\hat{n}_t^j(i^*)}} - 2\sqrt{\frac{\alpha \log \delta_t^{-1}}{2\hat{n}_t^j(i^*)}} \\ &= \hat{\mu}(i^*, \hat{n}_t^j(i^*)) - \sqrt{\frac{\alpha \log \delta_t^{-1}}{2\hat{n}_t^j(i^*)}}, \end{aligned}$$

contradicting the rules of the algorithm.

Thus, at time t, the selected arm i satisfies

$$\Delta_i \leq 2\sqrt{\frac{\alpha K \log \delta_t^{-1}}{(t-D)\Theta_{i^*} - Q}}.$$

The above equation is obtained by replacing $\hat{n}_t^j(i^*)$ in Equation (21) with $[(t-D)\Theta_{i^*}-Q]/K$, since $\hat{n}_t^j(i^*) \geq [(t-D)\Theta_{i^*}-Q]/K$.

For any agent j, the largest time slot when the agent makes a Type-I decision and a suboptimal arm i lies in the candidate set is

$$4\frac{\alpha K \log \delta^{-1}}{\Delta_i^2 \Theta_{i^*}} + D + \frac{Q}{\Theta_{i^*}}.$$

Then, the regret spent on the arm i in other agents is upper bounded by

$$4\frac{\alpha K \log \delta^{-1}}{\Delta_i}\frac{\Theta_i}{\Theta_{i^*}} + D\Theta_i + \frac{Q\Theta_i}{\Theta_{i^*}} + Q.$$

Summing up the above two pieces of regret and the expected number of Type-II decisions yields

$$\begin{aligned}\mathbb{E}[R_T] \leq & \sum_{i:\Delta_i>0} \max\left\{\frac{4\alpha \log \delta^{-1}}{\Delta_i}\left(2+\frac{K\Theta_i}{\Theta_{i^*}}\right) + F_i\Delta_i, \frac{12\Theta_i \alpha K \log \delta^{-1}}{\Delta_i \Theta_{i^*}} + KD\Theta_i\Delta_i\right\} \\ & + 2\left(1+\frac{\Theta_i}{\Theta_{i^*}}\right)\sum_{j\in\mathcal{A}}\sum_{l=1}^{N_j}\sum_{i\in\mathcal{K}_j}\frac{l\Theta_i}{\theta_j}\delta_{l/\theta_j}^{\alpha} + D\Theta.\end{aligned}$$

This completes the proof.