Heavy Tailed Reward Distributions for Multiplayer Bandits

Daphne Feng[1], Ricardo Parada[1], Lily Jiang [1], Sophia Yi[1], William Chang[1]
[1]University of California, Los Angeles

Abstract— **The multi-armed bandit (MAB) problem is a foundational model in decision theory and reinforcement learning, balancing exploration and exploitation in sequential decision-making. While extensive research has addressed MABs in both single-agent and multi-agent settings under sub-Gaussian reward assumptions, relatively little work has been done on multi-agent bandits with heavy-tailed reward distributions. In this paper, we extend the study of heavy-tailed bandits to the multi-agent setting, where multiple players interact in environments with varying levels of information asymmetry. We introduce three distinct problem formulations: (A) Action Information Asymmetry with Unobserved Actions and Common Rewards, (B) Reward Information Asymmetry with Observed Actions and Independent Rewards, and (C) Action and Reward Information Asymmetry with Unobserved Actions and Independent Rewards. In each case, agents must implicitly coordinate to maximize collective rewards, overcoming limited information and the challenges posed by heavy-tailed reward distributions. We propose robust algorithms adapted from classical bandit strategies to handle these challenges and establish theoretical performance guarantees. Our results demonstrate that effective learning is possible even in highly asymmetric multi-agent environments with heavy-tailed reward distributions, offering insights into decentralized learning and decision-making under uncertainty.**

I. Introduction

The multi-armed bandit (MAB) problem is a fundamental problem in reinforcement learning and decision theory in which an agent must repeatedly choose among multiple actions (or arms), each associated with an unknown and potentially varying reward distribution. This model captures the exploration-exploitation trade-off, in which the agent decides between selecting actions that have historically yielded high rewards or exploring new actions that may result in better long-term outcomes. This dilemma is central to various decision-making scenarios in which an agent must balance the risk of exploiting known options against the potential benefits of trying unknown ones. The MAB framework has been extensively studied in diverse settings, including stochastic and adversarial environments. It has also been applied in domains such as online recommendation systems, clinical trials, and financial portfolio optimization, where real-time decisions significantly impact outcomes.

The multi-agent multi-armed bandit (MMAB) problem naturally extends the well-studied single-agent MAB framework to scenarios involving multiple agents interacting within the same environment. In MMAB problems, each agent selects an arm simultaneously and independently from their own (marginal) set of actions, resulting in a joint action across all players. A key characteristic of many MMAB problems is information asymmetry, in which each agent lacks direct knowledge of other agents' actions and reward distributions. These restrictions introduce additional challenges: agents must be able to learn the reward distributions of their own set of arms while also inferring the actions of other players to select the optimal joint arm. In all the information asymmetric settings we consider, the players may agree on a policy before the learning phase begins but cannot communicate explicitly afterwards. Moreover, we focus on the cooperative variant of the MMAB problem, in which the goal is to select the best joint actions to maximize the collective rewards.

The vast majority of MAB problems assume that the reward distributions are either sub-Gaussian or light-tailed, allowing for well-behaved concentration properties that facilitate analysis and algorithm design. Under these conditions, established algorithms such as Upper Confidence Bound (UCB) and Thompson Sampling can achieve provable guarantees with relatively straightforward analysis. However, in many real-world cases, reward distributions may exhibit heavy-tailed behavior–i.e., their moments may not be finite beyond a certain order–and the variance of the rewards can be large. This introduces significant challenges in estimating expected rewards and designing optimal strategies, since the classical assumptions no longer hold, necessitating robust algorithms that can handle extreme variations in observed payoffs. These challenges are particularly pronounced in environments where rare but large reward events may skew the agent's learning processes.

In this paper, we extend the heavy-tailed bandit framework to the multi-player setting. In particular, we will focus on three types of multi-agent information asymmetry:

- Action Information Asymmetry with Unobserved Actions, Common Rewards
- Reward Information Asymmetry with Observed Actions, Independent Rewards
- Action and Reward Information Asymmetry with Unobserved Actions, Independent Rewards

which we describe in detail in II-B and refer to as Problem A, Problem B, and Problem C, respectively. Each of these problems introduces distinct challenges related to learning and implicit coordination among agents. By examining the interplay between information asymmetry and heavy-tailed reward distributions, we adapt existing algorithms to solve each of these problems.

A. Related Works

In this section, we will outline some of the related problems that have been well-studied in the literature.

a) Heavy-Tailed Bandits: Research on the MAB problem dates back to the 1930s [19], formally introduced in the 1950s by [15]. However, the study of MABs with heavy-tailed reward distributions–relaxing the standard sub-Gaussian assumption–has gained attention more recently. The first major contribution to this setting came from [4], which introduced the robust UCB algorithm using mean estimators to achieve logarithmic regret. A year later, [20] introduced the Deterministic Sequencing of Exploration and Exploitation (DSEE) algorithm, which achieves sublinear regret by alternating phases of exploration and exploitation.

Recent work has focused on refining exploration strategies for heavy-tailed rewards. [24] introduced a pure exploration algorithm tailored to this setting, while [16] extended the analysis to linear stochastic bandits, developing the MENU and TOFU algorithms based on median-of-means and truncation techniques to achieve optimal regret bounds. In the Lipschitz bandit setting, [12] proposed two adaptive algorithms, ADTM and ADMM, that enjoy sublinear regret. In a broader reinforcement learning domain, [25] proposed the Heavy-UCRL2 and Heavy-Q-Learning algorithms, leveraging robust mean estimation to establish near-optimal regret bounds for undiscounted reinforcement learning. Further generalizing the problem, [8] explored the adversarial setting and introduced the HTINF and AdaTINF algorithms, which achieve optimal regret bounds in both stochastic and adversarial environments. More recently, [11] developed the MR-UCB and MR-APE algorithms, which utilize minimax-optimal exploration methods to guarantee minimax optimality with minimal prior knowledge.

b) Multi-Agent Multi-Armed Bandits (MMAB): Several studies on the multi-player extension of the MAB problem assume some form of structured communication between agents. For example, in [1], players share their accumulated information with their immediate neighbors within a predefined graph structure. Later works introduced gossip-based approaches, including the ε-greedy strategy with gossip updates in [18], the gossip UCB algorithm in [10], and the accelerated gossip UCB in [13].

Another common setting assumes that all players choose from the same set of arms, creating the challenge of handling collisions when multiple agents select the same arm simultaneously. The dUCB4, algorithm proposed in [9], was among the first to address this issue, achieving a square-logarithmic regret bound. This result was later improved to logarithmic regret in [14] through a posterior sampling approach. Further refinements in [17] minimized regret in collision-dependent reward scenarios.

Beyond decentralized approaches, many studies have explored centralized MMAB algorithms. In the adversarial setting, [2] considers the case where players lack knowledge of the overall graph structure. To overcome this, local leaders are selected to guide followers using an EXP3-based play distribution. The DPE2 algorithm in [21] takes this further by electing a global leader responsible for exploring and communicating the best empirical arm to other players. Addressing the issue of variable communication costs, [23] developed MMAB algorithms with optimal regret and constant communication costs by leveraging a Distributed Online Estimation (DoE) communication policy. Recent work has also examined cooperative MMAB settings with information asymmetry among players, eliminating graph-based communication structures. Notably, [5] and [6] adapt the UCB and DSEE algorithms to achieve optimal regret bounds when players have varying levels of information about each other. For an in-depth overview of multi-player bandit algorithms, [3] provides a comprehensive survey covering approaches based on UCB, ε-Greedy, and Explore-Then-Commit (ETC) strategies.

To our knowledge, there are very few existing works that consider the multi-agent setting for bandits with heavy-tailed reward distributions. One such work is [7], which considers a networked setting where agents communicate with delays. They propose the MP-UCB algorithm, incorporating a message-passing protocol and proving optimal regret bounds in both decentralized and centralized settings. Another relevant study is [22], which examines agents communicating over sparse random graphs with heavy-tailed degree distributions and receiving heavy-tailed rewards. This work introduces the HT-HMUCB and HT-HTUCB algorithms, designed for settings with homogeneous and heterogeneous rewards, respectively. However, both of these studies rely on explicit communication between agents via graph-based structures. In contrast, our work considers a setting in which no communication is permitted during the learning process. Instead, agents must agree on a policy beforehand and adapt based on varying levels of information asymmetry.

B. Our contribution

In this work, we focus on the multi-player extension of the bandit problem with heavy tails. Specifically, we examine multiple cases of information asymmetry between the players. We introduce the problem definition and notation in Section II-A. We then present our extension to the multi-agent setting in Section II-B. Finally, we provide our proposed solutions and analysis of each case in Sections III and **??**, respectively.

II. Preliminary

A. Bandits with Heavy Tail Problem Definition

Consider the classical stochastic multi-armed bandit problem described as follows: an agent has K actions (or arms) to choose from at each time step. With each arm $\boldsymbol{a} \in \mathcal{A} := \{1, \dots, K\}$, associate a probability distribution $\nu_{\boldsymbol{a}}$ with finite mean $\mu_{\boldsymbol{a}}$, which are unknown to the agent. At each round $t \geq 1$, the agent chooses an arm $\boldsymbol{a} \in \mathcal{A}$ and receives a reward drawn from $\nu_{\boldsymbol{a}}$ independent of the previously drawn arm. Let $\mu_1 = \max_{\boldsymbol{a} \in \mathcal{A}} \mu_{\boldsymbol{a}}$ be the highest mean reward among all arms. Then, the expected regret R_T at round T is defined as:

$$R_T = \mathbb{E}\left[T\mu_1 - \sum_{t=1}^{T} \mu_{\boldsymbol{a}}\right] = T\mu_1 - \sum_{t=1}^{T} \mathbb{E}[\mu_{\boldsymbol{a}}]$$

Equivalently, if we let $\Delta_{\boldsymbol{a}} = \mu_1 - \mu_{\boldsymbol{a}}$ be the sub-optimality gap of action $\boldsymbol{a}$, we can write the regret in the form:

$$R_T = \sum_{\boldsymbol{a} \in \mathcal{A}} \Delta_{\boldsymbol{a}} \mathbb{E}[n_{\boldsymbol{a}}(T)]$$

where $n_{\boldsymbol{a}}(T) = \sum_{t=1}^{T} \mathbb{I}\{\boldsymbol{a}_t = \boldsymbol{a}\}$ is the number of times action $\boldsymbol{a}$ was chosen by the player by the end of round T. The goal of this problem is to minimize the expected regret when the unknown distributions $\nu_{\boldsymbol{a}}$ are heavy-tailed, i.e., not necessarily sub-Gaussian (and in fact, heavier than sub-Gaussian).

B. Extension to Multi-agent Setting

We extend the bandits with heavy tail problem to the multi-player setting, where instead of a single action generating rewards, we have a joint set of actions. We will first explain the general framework, then describe how we extend it to the different forms of information asymmetry in Problems A, B, and C.

Consider a set of M players, $1, \ldots, M$, in which each player i has a set $\mathcal{A}_i$ of K_i arms to choose from. We define the joint action space as $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_M$. At each time t, each player selects an arm from their respective sets simultaneously, denoting the M-tuple of chosen arms as $\boldsymbol{a} = (a_1, \ldots, a_M) \in \mathcal{A}$. After choosing a joint action, each player observes the reward $\mu_{\boldsymbol{a}}$ sampled from $\nu_{\boldsymbol{a}}$, which is independent from the previous rounds. As in the single-agent setting, we define the expected regret R_T as follows:

$$R_T = T\mu_1 - \sum_{t=1}^{T} \mathbb{E}[\mu_{\boldsymbol{a}}]$$

where the expectation is now taken with respect to the joint action taken $\boldsymbol{a} \in \mathcal{A}$ taken by the players at time t. Our goal is to minimize the expected regret when the unknown distributions $\nu_{\boldsymbol{a}}$ are heavy-tailed. We note that the players are not allowed to communicate once the learning begins, though they are allowed to agree on a strategy beforehand. They are also allowed to know the individual action space available to each player prior to learning.

We now define the above extension for each of Problem A, B, and C of Information Asymmetric Bandits:

Problem A: Action Information Asymmetry with Unobserved Actions, Common Rewards. In this setting, each player will observe the same reward $\mu_{\boldsymbol{a}}$, but is unable to observe the actions of other players. This means that the players need to infer the actions of the other players during the learning process.

Problem B: Reward Information Asymmetry with Observed Actions, Independent Rewards. Next, we consider the setting where each player can observe the actions of the other players but the rewards each player receives are i.i.d. (and therefore different). Each player can only observe their own reward, and not the rewards of the other players.

As in Problem A, for each round t, each player i chooses an arm from their respective sets to form a joint arm $\boldsymbol{a} \in \mathcal{A}$. Once this joint action is selected, each player will receive an i.i.d. copy of the rewards from the same distribution $\nu_{\boldsymbol{a}}$. That is, the reward is now a vector $\mu_{\boldsymbol{a}}$ where $\mu_{\boldsymbol{a}}^i$ is the reward that player i observes. Thus, the expected regret of player i is obtained by replacing $\mu_{\boldsymbol{a}}$ in the definition of regret with $\mu_{\boldsymbol{a}}^i$. However, we note that the expected regret is the same for all players since the rewards obtained are i.i.d.

Problem C: Action and Reward Information Asymmetry with Unobserved Actions, Independent Rewards. This setting combines the challenges of Problem A and Problem B, where in each round t, each player cannot observe the actions taken by the other players, and every player will observe an i.i.d. copy of the rewards. Hence, the reward is a vector $\mu_{\boldsymbol{a}}$, where $\mu_{\boldsymbol{a}}^i$ is the reward player i observes, and the expected regret is the same as in Problem B.

III. Main Results

A. Problem A

In Problem A, players cannot observe the actions of other players, which introduces potential miscoordination. Particularly, if the actual joint action taken deviates from what players intended (due to miscoordination), players will update their estimates for the incorrect action. To overcome these challenges, we propose the `mRUCB-A` algorithm.

In `mRUCB-A`, each player first estimates the reward of every joint arm by calculating its robust upper-confidence bound (RUCB), which is a more resilient estimate of the true mean reward compared to the standard empirical mean under a heavy-tailed reward distribution. The RUCB for a joint arm a at round t is defined as

$$\mathrm{RUCB}_{\boldsymbol{a}}(t) = \begin{cases} \infty & \text{if } n_{\boldsymbol{a}}(t) = 0 \\ \widehat{\mu}_{\boldsymbol{a}}(t) + \alpha_{\boldsymbol{a}}(t) & \text{otherwise.} \end{cases} \tag{1}$$

where $\widehat{\mu}_{\boldsymbol{a}}(t)$ is the robust mean estimate for $\boldsymbol{a}$ for round t, $n_{\boldsymbol{a}}(t)$ is the number of times that $\boldsymbol{a}$ is observed up to round t, and the radius of the confidence interval is

$$\alpha_{\boldsymbol{a}}(t) = v^{\frac{1}{1+\varepsilon}} \left(\frac{c \log(T^{\gamma})}{n_{\boldsymbol{a}}(t)} \right)^{\frac{\varepsilon}{1+\varepsilon}},$$

ensuring that, with high probability, $\widehat{\mu}_{\boldsymbol{a}}(t) \in [\mu_{\boldsymbol{a}} - \alpha_{\boldsymbol{a}}(t), \mu_{\boldsymbol{a}} + \alpha_{\boldsymbol{a}}(t)]$.

Next, each player selects the joint arm with the highest RUCB. However, miscoordination may occur if any joint arms have equal RUCBs, and if players have different RUCBs for the same joint arm. Ties in the multiplayer setting cannot be broken arbitrarily as in the single-player UCB algorithm because this will lead to players choosing different best joint arms. To resolve this, we define an *order relation* on $\mathcal{A}$, which players agree on before learning and obey for the remainder of the game. We define the joint items order relation as follows:

Definition 1: Let M be the number of players and let $\boldsymbol{a} = (a_1, \ldots, a_M), \boldsymbol{b} = (b_1, \ldots, b_M)$ be two joint arms in $\mathcal{A}$. We say $\boldsymbol{a} < \boldsymbol{b}$ if and only if there exists an $n \in \{1, 2, \ldots, M\}$ such that for all $i < n, a_i = b_i$ and $a_n < b_n$.

In the event of a tie in RUCB values, players choose the *lesser* joint arm according to this ordering. This coordinated tie-breaking rule ensures that players choose the same joint arm starting from the first round, and thereby maintain identical RUCB estimates over $\mathcal{A}$ since all players receive common rewards.

Algorithm 1: `mRUCB-A`

```
Denote $\boldsymbol{a}$ to be a joint arm and $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_M$
  to be the joint action space for $M$ players. Players
  agree to a predetermined ordering on $\mathcal{A}$.
Initialization:
    $\forall \boldsymbol{a} \in \mathcal{A}$, set $n_{\boldsymbol{a}}(0) \leftarrow 0$ and $\widehat{\mu}_{\boldsymbol{a}}(0) \leftarrow 0$.
for $t = 1, \ldots, T$ do
    foreach $\boldsymbol{a} \in \mathcal{A}$ do
        compute $RUCB_{\boldsymbol{a}}(t)$ (see 1)
    end
    // Joint arm selection Let $\boldsymbol{a}(t) \in \mathcal{A}$ be the joint
      arm with the highest $RUCB$; in the case of ties,
      players choose the lesser joint arm according to
      the defined ordering relation.
    Each player selects their corresponding individual
      arm as specified by $\boldsymbol{a}(t)$ and receives a common
      reward.
    // Update statistics
    Set $n_{\boldsymbol{a}}(t) \leftarrow n_{\boldsymbol{a}}(t) + 1$;
    Update the robust mean estimator for $\boldsymbol{a}(t)$.
end
```

B. Problem B

In contrast to Problem A, each player in the Problem B setting can observe the actions of other players but receives i.i.d. rewards. This means that players' rewards are generated independently from the same probability distribution. As a result, multiple players who pull the same joint arm may receive different rewards. This independence in rewards makes the `mRUCB-A` algorithm ineffective, as it relies on the assumption of common rewards and would lead to miscoordination among players. To approach this reward asymmetry problem, we propose the `mRUCB-Intervals` algorithm.

Recall that in Problem A, we defined the radius of the confidence interval to be $\alpha_{\boldsymbol{a}}(t)$ and showed that the true mean reward $\mu_{\boldsymbol{a}}$ of an arm lies within the confidence interval $[\hat{\mu}_{\boldsymbol{a}}(t) - \alpha_{\boldsymbol{a}}(t), \hat{\mu}_{\boldsymbol{a}}(t) + \alpha_{\boldsymbol{a}}(t)]$ with high probability. It follows that if the confidence interval for one pulled joint arm is below and disjoint (that is, does not overlap with) the confidence interval of another pulled joint arm, then the true mean reward of the first joint arm is less than that of the second joint arm with high probability.

From its definition, we observe that $\alpha_{\boldsymbol{a}}(t)$ decreases as the number of times $n_{\boldsymbol{a}}(t)$ that $\boldsymbol{a}$ is pulled increases. Consequently, the confidence intervals of each joint arm shrink as they are pulled more. We can leverage this property to our advantage by eliminating joint arms when their confidence intervals fall below and disjoint those of other arms. We eliminate such arms because, as established earlier, they are likely to have a true mean reward lower than that of the other arms.

Similar to `mRUCB-A`, players agree on a predetermined ordering of joint arms in the joint action space $\mathcal{A}$. In this setting, however, players cycle through this ordering by pulling their respective individual arm within each joint arm each round. Each player has their own set of confidence intervals associated with the joint arms $\boldsymbol{a} = (a_1, \ldots, a_M) \in \mathcal{A}$ for each round.

Initially, each player's confidence interval for each joint arm $\boldsymbol{a} \in \mathcal{A}$ is set to $(-\infty, \infty)$. To obtain finite confidence intervals, players must follow the predetermined ordering and pull each $\boldsymbol{a}$ once. Each of the M players independently selects from K possible individual arms, so the total number of joint arms is given by $|\mathcal{A}| = K^M$. Thus, this process occurs exactly K^M times.

After pulling each $\boldsymbol{a}$ at least once, if a player observes that the confidence interval of a joint arm $\boldsymbol{a}_i$ becomes below and disjoint from the other intervals in their set, they will sabotage the process by selecting a different individual arm the next time that joint arm $\boldsymbol{a}_i$ is meant to be pulled. As a result, the pulled joint arm will be different from the one in the intended order. Because players can observe actions, the other players will be able to infer from the changed joint arm that another player has eliminated the joint arm $\boldsymbol{a}_i$, and they will follow suit. This process repeats until the horizon T is reached.

Theorem 2: If players follow the `mRUCB-Intervals` algorithm (Algorithm 2) in the Problem B setting, then we have the following regret bound:

$$R_T \leq 2c\left(4^{\frac{1+\varepsilon}{\varepsilon}} v^{\frac{1}{\varepsilon}}\right) \log(T) \sum_{\boldsymbol{a} \in \mathcal{A}} \frac{1}{\Delta_{\boldsymbol{a}}^{\frac{1}{\varepsilon}}} + 2M \sum_{\boldsymbol{a} \in \mathcal{A}} \Delta_{\boldsymbol{a}}. \quad (2)$$

The full proof is in the appendix.

C. Problem C

Although players receive i.i.d. rewards in Problem B, the observability of actions allows players to coordinate using a predetermined ordering relation and elimination of suboptimal arms. However, this coordination strategy fails in Problem C, as players are unable to observe each other's actions or rewards.

To address this challenge, we propose the `mHT-DSEE` algorithm, which employs a structured exploration-exploitation strategy. Players first agree on a fixed ordering of joint arms and define a function $K(\lambda)$ that tends to infinity to ensure that later exploration phases collect more samples as the duration of exploitation phases grows exponentially. During the λ-th exploration phase (starting from $\lambda = 1$), each joint arm is pulled $K(\lambda)$ times to ensure sufficient feedback despite the heavy-tailed reward distribution. During the exploitation phase, each player commits to the arm with the highest RUCB until the next power of 2 round.

Initially, players may converge to different perceived optimal arms due to variance in their observations. However, as λ increases, the probability of selecting a suboptimal arm

Algorithm 2: `mRUCB-Intervals`

```
Each player $i$ has joint arms $\boldsymbol{a}(t)$, composed of individual arms from all players, in their desired sets $\mathcal{A}_i$.
Each player has a set of individual arms $\boldsymbol{a}_i(t)$.
Players agree to a predetermined ordering of the joint arms on $\mathcal{A}$.
For each player $i$ and joint arm $\boldsymbol{a}(t) \in \mathcal{A}$, initialize the confidence interval $I^i_{\boldsymbol{a}}(t) \leftarrow (-\infty, \infty)$.
for $t = 1, \dots, K^M$ do
    for each player $i \in \{1, \dots, M\}$ do
        Pull the individual arm $\boldsymbol{a}_i(t)$ corresponding to the joint arm $\boldsymbol{a}(t)$ once according to the predetermined ordering.
        Observe their i.i.d. reward and update $I^i_{\boldsymbol{a}}(t) \leftarrow [\hat{\mu}_{\boldsymbol{a}}(t) - \alpha_{\boldsymbol{a}}(t), \hat{\mu}_{\boldsymbol{a}}(t) + \alpha_{\boldsymbol{a}}(t)]$.
    end
end
For each $\boldsymbol{a}$, $n_{\boldsymbol{a}} \leftarrow 1$.
for $t = K^M + 1, \dots, T$ do
    for each player $i \in \{1, \dots, M\}$ do
        Identify the expected next joint arm $\boldsymbol{a}(t)$ in the predetermined ordering.
        if for this $\boldsymbol{a}(t)$, $I^i_{\boldsymbol{a}}(t)$ is below and disjoint all other arms in the player's desired set $\mathcal{A}_i$ then
            Pull another individual arm that is not $\boldsymbol{a}_i(t)$ and remove $\boldsymbol{a}(t)$ from $\mathcal{A}_i$.
        else
            Pull $\boldsymbol{a}_i(t)$.
        end
        Observe the actual joint arm $\boldsymbol{a}'(t)$ pulled for that round.
        if $\boldsymbol{a}(t) \neq \boldsymbol{a}'(t)$ then
            Remove $\boldsymbol{a}(t)$ from $\mathcal{A}_i$.
        end
        Observe their i.i.d. reward and update $I^i_{\boldsymbol{a}'}(t) \leftarrow [\hat{\mu}_{\boldsymbol{a}'}(t) - \alpha_{\boldsymbol{a}'}(t), \hat{\mu}_{\boldsymbol{a}'}(t) + \alpha_{\boldsymbol{a}'}(t)]$.
    end
    $n_{\boldsymbol{a}'} \leftarrow n_{\boldsymbol{a}'} + 1$
end
```

decreases significantly, so, with high probability, each player ultimately commits to the true optimal arm. Furthermore, because exploration phases are scheduled at powers of 2, the overall regret is bounded by $O(K(T)\log(T))$. A formal justification for this result is given in the proof of Theorem 3.

Theorem 3: If the players follow `mHT-DSEE` in Algorithm 3 in the setting of Problem C, then we have the following regret bound:

$$R_T \leq O\left(K(T)\log(T)\right) \tag{3}$$

The full proof is in the appendix. Note that the dependence on ϵ is hidden by $O(\cdot)$. In the proof we end up bounding the regret by an integral of the form $\left(\int_1^\infty t^{-\frac{K_0(t)}{c}\left(\frac{\gamma}{v^{1/(1+\epsilon)}}\right)^{(1+\epsilon)/\epsilon}} dt\right)$. Furthermore, note that in comparison to the regret bounds for Problems A and B, this is a gap dependent bound, where the gaps are hidden by the $O(\cdot)$ notation.

Algorithm 3: `mHT-DSEE`

```
1  Players agree on a fixed ordering of $\mathcal{A}$ that remains constant throughout the game.
2  Choose a monotonic function $K(\lambda) : \mathbb{N} \to \mathbb{N}$ such that $\lim_{t\to\infty} K(\lambda) = \infty$. Initialize $\lambda = 1$.
3  for each joint arm $\boldsymbol{a} \in \mathcal{A}$ do
4      For $K(\lambda)$ rounds, each player pulls their individual arms corresponding to the joint arm $\boldsymbol{a}$.
5  end
6  for each player $i \in \{1, \dots, M\}$ do
7      Player $i$ calculates the RUCB for each joint arm $\boldsymbol{a}$.
8      Player $i$ selects the arm with the highest RUCB. In case of a tie, a random selection is made. The chosen arm is then committed to until the next power of 2.
9  end
10 When $t = 2^n$ for some $n \geq \lfloor \log_2(K(1)K^M) \rfloor + 1$, return to step (3) and begin a new exploration phase, incrementing $\lambda$ by 1.
```

IV. Conclusion

In this paper, we extended the study of multi-armed bandits with heavy-tailed reward distributions to the multi-agent setting, considering various forms of information asymmetry among agents. By introducing three distinct problem formulations–Action Information Asymmetry with Unobserved Actions and Common Rewards, Reward Information Asymmetry with Observed Actions and Independent Rewards, and Action and Reward Information Asymmetry with Unobserved Actions and Independent Rewards–we have highlighted the unique challenges that arise in these decentralized learning environments. Our proposed robust algorithms, adapted from classical bandit strategies, provide effective solutions for handling heavy-tailed reward distributions while maintaining theoretical performance guarantees. These results not only demonstrate the feasibility of optimal learning in such complex, highly asymmetric settings but also contribute valuable insights to the broader field of decentralized decision-making and coordination under uncertainty. Future work can explore further refinements of these algorithms and their applicability to even more general multi-agent environments, including those with adversarial dynamics and more complex communication constraints.

References

[1] B. Awerbuch and R. Kleinberg. Competitive collaborative learning. *Journal of Computer and System Sciences*, 74(8):1271–1288, 2008. Learning Theory 2005.

[2] Y. Bar-On and Y. Mansour. Individual regret in cooperative nonstochastic multi-armed bandits. *CoRR*, abs/1907.03346, 2019.

[3] E. Boursier and V. Perchet. A survey on multi-player bandits, 2024.

[4] S. Bubeck, N. Cesa-Bianchi, and G. Lugosi. Bandits with heavy tail, 2012.

[5] W. Chang, M. Jafarnia-Jahromi, and R. Jain. Online learning for cooperative multi-player multi-armed bandits. *CoRR*, abs/2109.03818, 2021.

[6] W. Chang and Y. Lu. Optimal cooperative multiplayer learning bandits with noisy rewards and no communication. *arXiv preprint arXiv:2311.06210*, 2023.

[7] A. Dubey et al. Cooperative multi-agent bandits with heavy tails. In *International conference on machine learning*, pages 2730–2739. PMLR, 2020.

[8] J. Huang, Y. Dai, and L. Huang. Adaptive best-of-both-worlds algorithm for heavy-tailed multi-armed bandits. In *international conference on machine learning*, pages 9173–9200. PMLR, 2022.

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

[10] P. Landgren, V. Srivastava, and N. E. Leonard. On distributed cooperative decision-making in multiarmed bandits. *CoRR*, abs/1512.06888, 2015.

[11] K. Lee and S. Lim. Minimax optimal bandits for heavy tail rewards. *IEEE Transactions on Neural Networks and Learning Systems*, 35(4):5280–5294, 2024.

[12] S. Lu, G. Wang, Y. Hu, and L. Zhang. Optimal algorithms for Lipschitz bandits with heavy-tailed rewards. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4154–4163. PMLR, 09–15 Jun 2019.

[13] D. Martínez-Rubio, V. Kanade, and P. Rebeschini. Decentralized cooperative stochastic multi-armed bandits. *CoRR*, abs/1810.04468, 2018.

[14] N. Nayyar, D. Kalathil, and R. Jain. On regret-optimal learning in decentralized multi-player multi-armed bandits, 2016.

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

[16] H. Shao, X. Yu, I. King, and M. R. Lyu. Almost optimal algorithms for linear stochastic bandits with heavy-tailed payoffs. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[17] C. Shi and C. Shen. Multi-player multi-armed bandits with collision-dependent reward distributions. *IEEE Transactions on Signal Processing*, 69:4385–4402, 2021.

[18] B. Szorenyi, R. Busa-Fekete, I. Hegedus, R. Ormandi, M. Jelasity, and B. Kegl. Gossip-based distributed stochastic bandit algorithms. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 19–27, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.

[19] W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[20] S. Vakili, K. Liu, and Q. Zhao. Deterministic sequencing of exploration and exploitation for multi-armed bandit problems, 2013.

[21] P.-A. Wang, A. Proutiere, K. Ariu, Y. Jedra, and A. Russo. Optimal algorithms for multiplayer multi-armed bandits. In S. Chiappa and R. Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 4120–4129. PMLR, 26–28 Aug 2020.

[22] X. Wang and M. Xu. Multi-agent multi-armed bandit with fully heavy-tailed dynamics, 2025.

[23] L. Yang, X. Wang, M. Hajiesmaili, L. Zhang, J. C. S. Lui, and D. Towsley. Cooperative multi-agent bandits: Distributed algorithms with optimal individual regret and constant communication costs, 2023.

[24] X. Yu, H. Shao, M. R. Lyu, and I. King. Pure exploration of multi-armed bandits with heavy-tailed payoffs. In *UAI*, pages 937–946, 2018.

[25] V. Zhuang and Y. Sui. No-regret reinforcement learning with heavy-tailed rewards, 2021.