Multiplayer Combinatorial Bandits Under Information Asymmetry

Anonymous authors
Paper under double-blind review

Abstract

In this paper, we extend linear combinatorial bandits Gai et al. (2012) to a multiplayer setting with information asymmetry Chang et al. (2022); Chang and Lu (2024), where each player controls an arm and independently decides whether to pull it, with coordination allowed only before rounds begin. We analyze three scenarios: action asymmetry (players can't observe others' actions but receive identical rewards per iteration), reward asymmetry (players observe actions but receive private i.i.d. rewards), and combined asymmetry. We derive near-optimal, gap-independent regret bounds for all scenarios: For action or reward asymmetry, we achieve $\tilde{\mathcal{O}}(\sqrt{T})$, which improves significantly from Gai et al. (2010); for both action and reward asymmetry, we achieve near-optimal bounds similar to that of Chang et al. (2022). We finally generalize our results to settings where players decide either not to pull or to pull one out of multiple arms, and achieve similar bounds in similar settings as above.

Keywords: multi-agent learning, combinatorial bandits, UCB algorithm, information asymmetry, regret analysis.

1 Introduction

Multi-Armed Bandit (MAB) problems are a foundational model involving sequential decision-making under uncertainty, where an agent selects from $\mathcal{K}$ arms to maximize rewards over time. A common approach to solve the problem is using the upper confidence bound (UCB) Lai and Robbins (1985), which selects the arm with the highest upper bound on each round.

Combinatorial MAB extends traditional MAB by allowing actions that comprise multiple arms, with agents observing both individual and total rewards - a framework that efficiently handles exponential action spaces by leveraging information overlap. This has direct applications in online advertising, where an advertiser iteratively selects webpage subsets for ad placement, learning user click probabilities across different page combinations to maximize overall engagement. Similarly, viral marketing presents an analogous challenge: marketers must select optimal seed nodes in social networks while learning influence probabilities between users to maximize campaign spread.

On another branch of MAB, information asymmetric MAB algorithms have been extensively covered by Chang et al. (2022); Chang and Lu (2024; 2025); Chang and Karthik (2025). Three particular sub-problems arise: action asymmetry (where players cannot observe other players' actions but receive identical rewards per iteration), reward asymmetry (where players observe actions but receive private i.i.d. rewards), and combined asymmetry (both action and reward asymmetry). As mentioned above, information asymmetry has covered multiple settings, but no extension towards combinatorial work has been discussed at all.

As we approach this intersection, we bring up a crucial question: *Can we extend the information asymmetric MAB system to accommodate for combinatorial MAB?* We show that the answer is affirmative, by presenting two novel formulations: the 0-1 problem (each player decides whether to pull their assigned arm) and the joint arm problem (each player selects from their private set of arms).

Our Contribution

We extend combinatorial bandits to multiplayer settings with information asymmetry, considering two formulations: the 0-1 problem (binary pull/pull or no-pull decisions) and its generalization to joint arm selection. Our main contributions are:

(1) Near-optimal algorithms for three asymmetry levels: We develop UCB-based algorithms that achieve $\tilde{\mathcal{O}}(\sqrt{T})$ regret for action- or reward-only asymmetry (Problems A and B), significantly improving upon existing $\mathcal{O}(\log T/\Delta^2_{\min})$ gap-dependent bounds. For full asymmetry (Problem C), we adapt deterministic exploration to achieve regrets similar to Chang et al. (2022).

(2) Unified framework from binary to multi-arm: Our algorithms naturally extend from the 0-1 setting to the general K-arm setting, with the dependence on K appearing only as a multiplicative factor, implying that multi-arm complexity doesn't fundamentally change the problem structure.

(3) Coordination without communication: We extend state-of-the-art mechanisms Chang et al. (2022); Chang and Lu (2024) that enable decentralized coordination despite information constraints, with only additive $O(|\mathcal{F}|)$ overhead for signaling sabotage.

2 Related Works

Combinatorial Bandits The combinatorial bandit setting has been widely studied under various structural assumptions. Chen et al. (2013) propose a general CMAB framework with the CUCB algorithm achieving tight regret bounds. Gai et al. (2010) model multiuser channel allocation as a combinatorial bandit with the MLPS algorithm, extended in Gai et al. (2012) to linear rewards via LLR. Wang and Zhu (2022) introduce TS-Explore for combinatorial pure exploration, improving over UCB methods. In adversarial settings, Cesa-Bianchi and Lugosi (2012) develop ComBand with optimal regret bounds, while Combes et al. (2015) propose ESCB and COMBEXP. Chen et al. (2014) and Liu et al. (2024) contribute to pure exploration and multivariant extensions, respectively.

Cooperative Multiplayer Bandits In cooperative multiplayer settings, players collaborate to identify optimal arms from a common pool with communication constrained by graph structure Awerbuch and Kleinberg (2008). Proposed algorithms include ϵ-greedy approaches Szorenyi et al. (2013), gossip-based UCB methods Landgren et al. (2016); Martínez-Rubio et al. (2019), and leader-based strategies Wang et al. (2020). In adversarial settings, Bar-On and Mansour (2019) introduced leader coordination with EXP3. Other models consider neighbor reward observation Cesa-Bianchi et al. (2016) and asynchronous participation Bonnefoi et al. (2017); Cesa-Bianchi et al. (2020). Proutiere and Wang (2019) extended collision-based frameworks to Lipschitz settings with the DPE algorithm.

Competing Bandits The competing bandits framework Liu et al. (2020) incorporates player preferences in collision models: only the highest-priority player receives rewards when multiple players select the same arm. The centralized CUB solution has players report UCB values to a coordinator. Cen and Shah (2022) showed optimal logarithmic regret with platform-managed reassignments, while Jagadeesan et al. (2021) introduced equilibrium concepts via negotiated transfers. An Explore-Then-Commit algorithm Liu et al. (2020) attains optimal regret assuming known gaps, with Sankararaman et al. (2021) later removing this assumption. Liu et al. (2021) developed decentralized UCB with collision avoidance.

Game Theory Several works bridge bandits and game theory. Mao (2024) proposes decentralized algorithms for equilibria in Markov games. Bistritz and Leshem (2018) introduce Game of Thrones for fully decentralized multiplayer bandits without communication. Brânzei and Peres (2021) characterize utility alignment effects in multiplayer stochastic bandits, while Bravo et al. (2018) analyze Nash equilibria convergence using no-regret learning. Ontanón (2013; 2017) develop methods for combinatorial bandits in Monte Carlo Tree Search. Heliou et al. (2017) show exponential weights achieve Nash equilibria in potential games under bandit feedback. Scheid et al. (2024) propose IPA algorithms for principal-agent settings with misaligned objectives, and Kang et al. (2021) study behavioral aspects through energy-based measures.

3 Preliminaries

We extend combinatorial bandits Cesa-Bianchi and Lugosi (2012); Gai et al. (2012) to a multiplayer setting with information asymmetry Chang et al. (2022), where M players collaborate to maximize their collective reward over T rounds. Each player $m \in [M]$ controls one or more arms and must decide which to pull (or abstain). Players cannot communicate during rounds - they must coordinate only through pre-agreed strategies and environmental feedback.

Problem Structure A combinatorial arm $\vec{a} = (a_1, a_2, \ldots, a_M)$ represents a joint action where each a_m is player m's choice. Individual arm rewards are assumed to be 1-subGaussian. Not all combinations are valid: only $\vec{a} \in \mathcal{F} \subseteq (K+1)^{[M]}$ yield rewards, where $\mathcal{F}$ is the feasible set known to all players. When players select an invalid combination, they receive zero reward. Each valid $\vec{a}$ has an unknown mean reward $\mu_{\vec{a}}$, and we denote $\mu^* = \max_{\vec{a} \in \mathcal{F}} \mu_{\vec{a}}$ as the optimal value.

Information Asymmetry The core difficulty arises from what players can and cannot observe:

- **Problem A (Action Asymmetry):** Players cannot observe others' actions but receive identical rewards. When $\vec{a}$ is pulled, all see reward $X_{\vec{a}}(t)$.
- **Problem B (Reward Asymmetry):** Players observe actions but receive private rewards. When $\vec{a}$ is pulled, player m receives $\{X^j_{a_m}(t)\}_{j=1}^M \sim \mathcal{D}_{a_m}$ [1].
- **Problem C (Full Asymmetry):** Players observe neither actions nor share rewards - each sees only their own arm and reward.

Coordination Mechanism We establish a feedback mechanism for all 3 problems, inspired by Chang et al. (2022), below.

Remark 1 *For Problems A and C (with action asymmetry), we propose the following feedback mechanism: Before play begins, players agree on a fixed predetermined ordering of all arms in the feasible set. When a combinatorial arm $\vec{a}$ is pulled, the realized rewards are reported according to this ordering, so the feedback can be consistently aligned with arm indices. Although players cannot observe which specific arms were actually selected by others, this mechanism allows them to incorporate the ordered reward outcomes into their confidence-interval updates. For Problem B (with reward asymmetry), we add the sabotage rounds upon the predetermined ordering to broadcast the asymmetric reward information.*

Formulations We study two variants:

1. **0-1 Problem:** Each player has one arm ($K = 1$) and decides to pull or abstain
2. **Joint Arm Problem:** Each player selects from K arms or abstains

We measure performance via cumulative regret:

$$R_T = T\mu^* - \sum_{t=1}^{T} \mathbb{E}[X_{\vec{a}_t}],$$

where $X_{\vec{a}_t}$ is the reward at time t. We start with the 0-1 problem.

4 The 0-1 Problem

4.1 Problem A: Information Asymmetry in Actions

In Problem A, we introduce action asymmetry, where players cannot observe others' decisions despite receiving identical rewards. Specifically, when a combinatorial arm $\vec{a} \in \mathcal{F}$ is pulled, all players observes the same

[1] $X^j_{a_m}(t)$ denotes the reward received by player m, observed by player j, at iteration t.

stochastic reward $X_{\vec{a}}(t)$, but cannot infer which individual arm were selected. This creates a fundamental challenge of updating estimates without knowing which arms to update. To address this, we implement a UCB-based approach. For each individual arm a, Let

$$\epsilon_a(t) := \sqrt{4 \log T / n_a(t)}. \tag{1}$$

be the confidence radius, where $n_a(t)$ denotes the number of arm a pulled up to but not including t, and T is the time horizon. We define the UCB and LCB indices for individual arms as

$$\mathrm{UCB}_a(t) = \widehat{\mu}_a(t) + \epsilon_a(t), \quad \mathrm{LCB}_a(t) = \widehat{\mu}_a(t) - \epsilon_a(t), \tag{2}$$

while for combinatorial arms they are defined as

$$\begin{aligned} \mathrm{UCB}_{\vec{a}}(t) &= \sum_{a_m \in \vec{a}} \mathrm{UCB}_{a_m}(t), \\ \mathrm{LCB}_{\vec{a}}(t) &= \sum_{a_m \in \vec{a}} \mathrm{LCB}_{a_m}(t). \end{aligned} \tag{3}$$

As shown in Chang et al. (2022), with high probability, the true mean should fall within

$$I_{\vec{a}}(t) = [\mathrm{LCB}_{\vec{a}}, \mathrm{UCB}_{\vec{a}}]. \tag{4}$$

We define the good event for individual arms, as well as the good event across all arms and time as

$$G = \bigcap_{t=1}^{T} \bigcap_{a=1}^{M} G_a(t), G_a(t) = \left\{ |\widehat{\mu}_a - \mu_a| \leq \sqrt{\frac{4 \log T}{n_a(t-1)}} \right\} \tag{5}$$

respectively. This gives us Lemma 2, the details of which will be deferred to the Appendix.

Lemma 2 *The probability of the bad event G^c in (5) is upper bounded by*

$$P(G^c) \leq M/T.$$

4.1.1 Procedure of Algorithm

As mentioned in Remark 1, the key idea to solve problem A is that players can establish a shared ordering over $\mathcal{F}$ before learning begins. In particular, during execution, each player maintains confidence intervals $[\mathrm{LCB}_{\vec{a}}, \mathrm{UCB}_{\vec{a}}]$ for all $\vec{a} \in \mathcal{F}$ and updates those intervals separately. We put the combinatorial arms into the following sets depending on whether they contain arm a_m:

$$\forall m \in \{1, \ldots, M\} : S_m = \{\vec{a} \in \mathcal{F} : a_m \in \vec{a}\} \tag{6}$$

Following the predetermined order, all players jointly execute the same predetermined joint action $\vec{a}_t$ each round, ensuring synchronized updates. When $\mathrm{UCB}_{\vec{a}} < \mathrm{LCB}_{\vec{a}'}$ for any arms $\vec{a}$ and $\vec{a}'$, $\vec{a}$ is identified as suboptimal and removed from $\vec{a}$ from all S_m such that $\vec{a} \in S_m$. As our order is constructed such that every individual arm a_m is guaranteed to be pulled at least once every M rounds, this ordering ensures that we will have sufficient exploration. In fact, Remark 5 provides a significant argument for this choice of order, and we leave it after the regret bound for sake of clarity and contrast. The implementation is shown to the right; we make the arguments above rigorous with a lemma.

Lemma 3 *For each individual arm $a \in \mathcal{A}$ and for all rounds t, the empirical mean $\widehat{\mu}_a(t)$, pull count $n_a(t)$ and confidence interval remain identical across all players.*

Algorithm 1 `mCombinatorialUCB-A`

Require: Horizon T,
Family of feasible combinatorial arms $\mathcal{F}$,
Sets $S_1, \ldots, S_M$ with fixed internal order.

Initialize:
$m \leftarrow 1$ ▷ index of the current set
for each individual arm a_m **do**
  $n_{a_m}(0) \leftarrow 0$
  $\hat{\mu}_{a_m}(0) \leftarrow 0$
  $\mathrm{UCB}_{a_m}(0) \leftarrow +\infty$
  $\mathrm{LCB}_{a_m}(0) \leftarrow -\infty$
end for
for $t = 1$ **to** T **do**
  Let S_m be the current set and let $\vec{a}$ be the next combinatorial arm in (6)
  $\vec{a}_t \leftarrow \vec{a}$
  Player-wise elimination che:
  while $\exists \vec{a}' \in \mathcal{F} : \mathrm{UCB}_{\vec{a}_t}(t-1) < \mathrm{LCB}_{\vec{a}'}(t-1)$ **do**
    Remove $\vec{a}_t$ from all S_j with $\vec{a}_t \in S_j$
    Set $\vec{a}_t$ to the next arm in S_m under the fixed ordering
  end while
  Update individual-UCB indices (for components pulled in $\vec{a}_t$):
  for all $a_m \in \vec{a}_t$ **do**
    Observe reward $X_{a_m}(t)$ for component a_m
    $n_{a_m}(t) \leftarrow n_{a_m}(t-1) + 1$
    $\hat{\mu}_{a_m}(t) \leftarrow \frac{n_{a_m}(t-1)\hat{\mu}_{a_m}(t-1) + X_{a_m}(t)}{n_{a_m}(t)}$
    $\mathrm{UCB}_{a_m}(t) \leftarrow \hat{\mu}_{a_m}(t) + \sqrt{\frac{4 \log T}{n_{a_m}(t)}}$
    $\mathrm{LCB}_{a_m}(t) \leftarrow \hat{\mu}_{a_m}(t) - \sqrt{\frac{4 \log T}{n_{a_m}(t)}}$
  end for
  Update combinatorial-UCB indices:
  for all $\vec{a} \in S_m$ **do**
    $\mathrm{UCB}_{\vec{a}}(t) \leftarrow \sum_{a_m \in \vec{a}} \mathrm{UCB}_{a_m}(t)$
    $\mathrm{LCB}_{\vec{a}}(t) \leftarrow \sum_{a_m \in \vec{a}} \mathrm{LCB}_{a_m}(t)$
  end for
  $m \leftarrow (m \bmod M) + 1$ ▷ advance to next set cyclically
end for

Proof: We proceed by induction. Initially, all players start with identical $(-\infty, \infty)$ intervals. Assuming synchronization at round t, when arm $\vec{a}_{t+1}$ is pulled at round $t+1$, all players receive identical feedback $X_{\vec{a}_{t+1}}$ aligned under the ordering and update the same individual arms (by the predetermined order) with the same values (by the identical reward received). Arms not in $\vec{a}_{t+1}$ remain unchanged. Therefore, all players maintain identical statistics and confidence intervals for every arm at all rounds. □

The regret bound is as follows:

Theorem 4 *`mCombinatorialUCB-A` achieves regret bound*

$$R_T \leq 8M(M+1)\sqrt{T \log T} + M. \tag{7}$$

Proof deferred to Appendix; here we state only the high-level argument. Our novel development is by extending our work beyond Gai et al. (2012), which achieved a gap-dependent regret of order $O\left(\frac{\log T}{\Delta^2_{\min}}\right)$ where $\Delta_{\min} = \min_{\vec{a}} \Delta_{\vec{a}}$. As converting this to a gap-independent bound is challenging due to potentially large ratios of the form $\frac{\Delta_{\vec{a}}}{\Delta^2_{\min}}$, we developed a new proof technique that directly achieves a gap-independent bound. We can now compare the regret of our approach versus the naive approach:

Remark 5 *Consider an arbitrary ordering where all combinations of individual arms $\{a_1, ..., a_{M-1}\}(\mathcal{P}(\{a_1, ..., a_{M-1}\}))$ are pulled before a_M, which is the combinatorial arm where all players not pull their arms but a_M, and assume a_M is optimal. This would result in a_M not being pulled for 2^{M-1} consecutive rounds, which enlarges the confidence interval significantly and consequently leads to a regret of at least $2^M\sqrt{4 \log T}$.*

As mentioned above, our design choice is *critical* for this exact reason. By ensuring no arm is "starved" for more than M rounds, we avoid the exponential dependence on M and achieve the polynomial regret bound stated in Theorem 4.

4.2 Problem B: Information Asymmetry in Rewards

In Problem B, we introduce reward asymmetry: players can observe each other's actions, but receive i.i.d reward feedback. When a combinatorial arm $\vec{a}_t \in \mathcal{F}$ is selected, for each $a_m \in \vec{a}_t$, each player j observes rewards that conforms to i.i.d. $\{X^j_{a_m}(t)\}_{j=1}^M \sim \mathcal{D}_{a_m}$ [2] from their pulled arm's distribution.

This fundamentally differs from Problem A, which means that synchronized feedback in `mCombinatorialUCB-A` breaks down as reward observations differ across players, even with identical action sequences. Consequently, players disagree on which arms are suboptimal: one player might eliminate a joint action that another still considers viable. This divergence breaks the possibility of maintaining a common feasible set $\mathcal{F}$ or shared confidence intervals.

This also means that while UCB remains valid locally for each player, coordinated elimination becomes impossible, invalidating the assumptions required for sublinear regret bounds.

4.2.1 Solution: Sabotage-Based Coordination

To adapt `mCombinatorialUCB-A` to Problem B, we introduce a minimal coordination mechanism of protocol-based sabotage, which has been covered in Chang and Lu (2024). While players receive distinct rewards, they can observe actions, enabling implicit coordination without sharing reward values. The general idea is as follows:

1. Each player maintains private empirical estimates and confidence intervals[3]:

$$\begin{cases} I^j(t) := (\mathrm{LCB}^j_{a_m}(t), \mathrm{UCB}^j_{a_m}(t)), \\ \mathrm{LCB}^j_{a_m}(t) = \widehat{\mu}^j_{a_m}(t) - \epsilon_{a_m}(t), \\ \mathrm{UCB}^j_{a_m}(t) = \widehat{\mu}^j_{a_m}(t) + \epsilon_{a_m}(t). \end{cases}$$

[2] $X^j_{a_m}(t)$ denotes the reward received by player m, observed by player j, at iteration t.

[3] Note that $\epsilon_{a_m}(t)$ is not player-indexed since it depends only on pull counts, which all players can track.

Algorithm 2 `mCombinatorialUCB-B`

Require: Horizon T,
Family of feasible combinatorial arms $\mathcal{F}$,
Sets $S_1, \ldots, S_M$ with fixed internal order.
Ensure: Eliminate suboptimal combinatorial arms and select from surviving set

Initialize:
$m \leftarrow 1$ ▷ index of the current set
for each individual arm a_m **do**
  $n_{a_m}(0) \leftarrow 0$
  $\hat{\mu}_{a_m}(0) \leftarrow 0$
  $\mathrm{UCB}_{a_m}(0) \leftarrow +\infty$
  $\mathrm{LCB}_{a_m}(0) \leftarrow -\infty$
end for
for $t = 1$ **to** T **do**
  Let S_m be the current set; let $\vec{a}$ be the next combinatorial arm in (6)
  $\vec{a}_t \leftarrow \vec{a}$
  Player-wise elimination check with sabotage:
  while $\exists \vec{a}' \in \mathcal{F} : \mathrm{UCB}_{\vec{a}_t}(t-1) < \mathrm{LCB}_{\vec{a}'}(t-1)$ **do**
    Player j sabotages $\vec{a}_t$
    Remove $\vec{a}_t$ from all S_r with $\vec{a}_t \in S_r$
    Set $\vec{a}_t$ to the next arm in S_m under the fixed ordering
  end while
  Update individual indices (for components pulled in $\vec{a}_t$):
  for all $a_m \in \vec{a}_t$ **do**
    Observe reward $X_{a_m}(t)$ for component a_m
    $n_{a_m}(t) \leftarrow n_{a_m}(t-1) + 1$
    $\hat{\mu}_{a_m}(t) \leftarrow \frac{n_{a_m}(t-1)\hat{\mu}_{a_m}(t-1) + X_{a_m}(t)}{n_{a_m}(t)}$
    $\mathrm{UCB}_{a_m}(t) \leftarrow \hat{\mu}_{a_m}(t) + \sqrt{\frac{4 \log T}{n_{a_m}(t)}}$
    $\mathrm{LCB}_{a_m}(t) \leftarrow \hat{\mu}_{a_m}(t) - \sqrt{\frac{4 \log T}{n_{a_m}(t)}}$
  end for
  Update combinatorial indices (for current set)
  for all $\vec{b} \in S_m$ **do**
    $\mathrm{UCB}_{\vec{b}}(t) \leftarrow \sum_{a_q \in \vec{b}} \mathrm{UCB}_{a_q}(t)$
    $\mathrm{LCB}_{\vec{b}}(t) \leftarrow \sum_{a_q \in \vec{b}} \mathrm{LCB}_{a_q}(t)$
  end for
  $m \leftarrow \big(m \bmod M\big) + 1$ ▷ advance to next set cyclically
end for

2. When player m determines $\vec{a} \in \mathcal{F}$ is suboptimal, they initiate a sabotage round by:
   - Not pulling a_m if $a_m \in \vec{a}$.
   - Pulling a_m if $a_m \notin \vec{a}$.
3. All players eliminate $\vec{a}$ from relevant sets S_j that contain a_m when observing the pulled joint arm is not aligned with the agreed arm to pull in the predetermined order.

The regret bound for this algorithm is

Theorem 6 *`mCombinatorialUCB-Intervals-B` suffers regret:*

$$R_T \leq |\mathcal{F}| + 8M(M+1)\sqrt{T \log T} + M$$

The proof follows immediately from the results in Theorem 4. However, we note that equation 15 only holds for non-sabotaged rounds; otherwise we assume maximum regret every round of 1. Since there are at most $|\mathcal{F}|$ sabotages, our desired regret bound follows. Despite the additional $|\mathcal{F}|$ term, this remains superior to applying UCB on all combinatorial arms (which yields $O(\sqrt{|\mathcal{F}|T})$ regret).

4.3 Problem C: Information Asymmetry in Actions and Rewards

In Problem C, we face asymmetry in both actions and rewards: each player m observes only their own arm choice a_m; for each $a_m \in \vec{a}_t$, each player j observes rewards that conforms to i.i.d. $\{X^j_{a_m}(t)\}_{j=1}^M \sim \mathcal{D}_{a_m}$. For problem C, from player j's perspective, the identity of a_m is anonymous, they only observe their own realized reward, while the actions and outcomes of the other players remain hidden. Reward realizations are independent across both time and players. This eliminates both the synchronized updates from Problem A and the sabotage signaling from Problem B, motivating the application of `mDSEE` from Chang et al. (2022). However, direct application of this algorithm would face scalability issues with $|\mathcal{F}| \sim 2^M$.

Our novel solution covers the combinatorial arm space into M ordered subsets as in equation (6). In exploration phases (occurring at powers of 2), players pull representatives from each S_m exactly $f(\lambda)$ times, where $f(\lambda) \to \infty$ as $\lambda \to \infty$, and $f(\lambda)$ is nondecreasing. The function f plays the role of an exploration parameter: if f is larger, players collect more samples per epoch, which improves the accuracy of their reward estimates and increases the likelihood committing to the true best arm in later phases. The trade-off is that a larger f also means more exploration rounds, which adds to the regret. Each player uses only rewards from their own arm a_m to form empirical means of joint arms. During commit phases, players independently select their empirically best combinatorial arm and execute it until the next exploration. Initially, players miscoordinate due to limited samples, but as λ grows and the number of samples increases, their estimates converge, ensuring eventual convergence in coordination. See Algorithm 3 for details.

Theorem 7 *The regret of the `mDSEE` Algorithm is*

$$R_T = \mathcal{O}\left(\lfloor \log_2(T) \rfloor f(\lfloor \log_2(T) \rfloor) M\right).$$

Hidden in this regret bound is the asymptotic dependence on the gaps of the combinatorial arms and the choice of f, which makes our bound non-standard. However, we are still able to show that the regret grows logarithmically with respect to a fixed set of reward gaps. The proof is deferred to the Appendix.

5 The Joint Arm Problem

We now generalize from binary decisions to multi-arm selection: each player m has K arms with means $\mu_m^1, ..., \mu_m^K$ and selects one (or abstains). The combinatorial reward structure remains - only valid combinations $\vec{a} \in \mathcal{F}$ yield rewards - but arm quality now depends on specific selections within each player's set. Consider

Algorithm 3 `mDSEE Algorithm`

Require: Per-player set sizes $(K_1, \ldots, K_M)$; monotone $f(\lambda)$ with $f(\lambda) \to \infty$ as $\lambda \to \infty$
Ensure: Alternating exploration and commit phases

1: $\lambda \leftarrow 1$; $\quad t \leftarrow 1$ ▷ epoch and round counters
2: **while** $t \leq T$ **do**
3: **Exploration phase (for $f(\lambda)$ rounds):**
4: **for** $m = 1$ **to** M (cyclic) **do**
5: Let $\vec{a}$ be the next combinatorial arm in (6)
Observe rewards $r_1, \ldots, r_M$ for $\vec{a}$;
6: **for** $j = 1$ **to** M **do**
7: $i \leftarrow (\vec{a})_j$;
8: $n_i^{(j)} \leftarrow n_i^{(j)} + 1$;
9: $\hat{\mu}_i^{(j)} \leftarrow \hat{\mu}_i^{(j)} + \dfrac{r_j - \hat{\mu}_i^{(j)}}{n_i^{(j)}}$.
10: **end for**
11: $t \leftarrow t + 1$
12: **if** $t > T$ **then**
13: **break**
14: **end if**
15: **end for**
16: **if** $t > T$ **then**
17: **break**
18: **end if**
19: **Estimate empirical means:**
20: **for** each combinatorial arm $\vec{a} \in \mathcal{F}$ **do**
21: **for** $m = 1$ **to** M **do**
22: Compute $\hat{\mu}_{\vec{a}}^m$ as the sample mean of rewards observed for individual arm a by player m
23: **end for**
24: **end for**
25: **Commit phase (until next power-of-2 round):**
26: **for** $m = 1$ **to** M **do**
27: $a_{\text{commit}}^m \leftarrow \arg\max_{\vec{a} \in \mathcal{F}} \hat{\mu}_{\vec{a}}^m[m]$
28: **end for**
29: Find the least integer r with $2^r > t$; set $T_{\text{commit}} \leftarrow 2^r - t$
30: **for** $t = 1$ **to** T_{commit} **do**
31: $\vec{a}_{\text{commit}} \leftarrow (a_{commit}^1, \ldots, a_{commit}^M)$;
32: upon observing $r_1, \ldots, r_M$, update $n_{(\vec{a}_{\text{commit}})_j}^{(j)}$ and $\hat{\mu}_{(\vec{a}_{\text{commit}})_j}^{(j)}$ as in line 6.
33: **if** $t > T$ **then**
34: **break**
35: **end if**
36: **end for**
37: $\lambda \leftarrow \lambda + 1$ ▷ start next exploration epoch
38: **end while**

this reward matrix:

$$\begin{matrix} \emptyset & \emptyset & \cdots & \emptyset \\ \mu_1^1 & \mu_2^1 & \cdots & \mu_M^1 \\ \mu_1^2 & \mu_2^2 & \cdots & \mu_M^2 \\ \vdots & \vdots & \ddots & \vdots \\ \mu_1^K & \mu_2^K & \cdots & \mu_M^K \end{matrix}$$

where $\emptyset$ denotes not pulling. For players $\{1, 2, M\}$, the orange values $\mu_1^1 + \mu_2^2 + \mu_M^1$ might yield a poor combinatorial arm, while the blue values $\mu_1^K + \mu_2^K + \mu_M^K$ might produce an excellent one. As we will show below, this change in structure will leave to higher complexities in regret bounds.

5.1 Problem A: Information Asymmetry in Actions

While it's tempting to naively expand Problem A's approach for $0-1$, with K arms per player, naive cycling through all K^M actions would yield exponential regret. We extend `mCombinatorialUCB-A` by maintaining two types of desired sets: $S_1, ..., S_M$ for combinatorial arms (as before) and $D_1, ..., D_M$ for individual arms, where D_m tracks player m's potentially optimal arms.

$$\forall m \in \{1, \ldots, M\} : D_m = \{\, a_m^k \in [K] : a_m^k \in \vec{a} \,\} \tag{8}$$

The key modification to the $0-1$ case is that when pulling combinatorial arm $\vec{a} \in S_m$, we pull it $|D_m|$ times with player m cycling through their arms in D_m. For combinatorial arms, we define intervals that account for all possible arm selections:

$$I_{\vec{a}}(t) = \left[\sum_{a_m \in \vec{a}} \min_{k \in [K]} \mathrm{LCB}_{a_m^k}(t), \sum_{a_m \in \vec{a}} \max_{k \in [K]} \mathrm{UCB}_{a_m^k}(t) \right],$$

where the arm elimination heuristic extends from the $0-1$ Problem. Players eliminate arms at two levels: (1) individual arms within D_m when another arm in D_m dominates it, and (2) combinatorial arms from S_m when another combinatorial arm globally dominates it. Since all players receive identical rewards and follow the same predetermined ordering, they maintain synchronized confidence intervals and make identical elimination decisions, not observing but knowing each other's actions.

Theorem 8 *`Joint-mCombinatorialUCB-A` suffers*

$$R_T \leq 4K^2M^2\sqrt{T \log T}.$$

The proof of this, as well as the pseudocode, is deferred to the Appendix.

5.2 Problem B: Information Asymmetry in Rewards

The sabotage mechanism from Section 4.2.1 extends naturally to the multi-arm setting with one key modification: we must distinguish between individual and combinatorial arm sabotage to prevent signaling conflicts.

When considering combinatorial arm $\vec{a} \in S_m$ and individual arm $a_m^k \in D_m$:

- Only player m sabotages arms in their own set D_m. This ensures each player controls their own arm elimination based on their private observations.
- Only players $j \neq m$ can sabotage $\vec{a} \in S_m$. This prevents player m from both selecting and sabotaging simultaneously, creating ambiguous signals.

This separation would allow observers to know whether the sabotage targets an individual arm (when player m deviates) or a combinatorial arm (when others deviate).

Theorem 9 *`Joint-mCombinatorialUCB-B` suffers*

$$R_T \leq MK + |\mathcal{F}| + 4K^2M^2\sqrt{T\log T}.$$

Note the addition of $|\mathcal{F}|$, which is required in order to account for rounds where combinatorial arm elimination occurs. Despite this additional term, the regret bound remains superior to applying UCB on all combinatorial arms (which yields $\mathcal{O}(\sqrt{|\mathcal{F}|T})$ regret). The proof and pseudocode will be deferred to the Appendix.

5.3 Problem C: Information Asymmetry in Actions and Rewards

The `mDSEE` approach from Section 4.3 extends directly to the multi-arm setting with a single modification: during exploration phases, we pull each representative $\vec{a} \in S_m$ exactly K times, allowing player m to cycle through all arms in D_m. This ensures each player accumulates sufficient samples for every arm despite the increased action space.

Theorem 10 *The regret of `Joint-mDSEE` is*

$$R_T = \mathcal{O}\left(\lfloor\log_2(T)\rfloor f(\lfloor\log_2(T)\rfloor)KM\right).$$

The K-fold increase in exploration naturally appears as a multiplicative factor in the regret bound. As in the binary case, the gap dependence is embedded in the choice of $f(\lambda)$, with regret growing logarithmically for fixed reward gaps. The proof is deferred to the Appendix.

6 Experimental Results

We validate our theoretical bounds through experiments on both $0-1$ and joint arm combinatorial bandits, comparing `mCombinatorialUCB-A` (action asymmetry), `mCombinatorialUCB-B` (reward asymmetry with sabotage), and `mDSEE` (full asymmetry).

6.1 Setup

We consider M players where each player has K arms plus a null action (0). The total action space contains $(K+1)^M - 1$ joint actions (excluding the all-zero action). The feasible set $\mathcal{F}$ is a random subset of this space, with $|\mathcal{F}|$ denoting its cardinality. Per-player true arm rewards are drawn i.i.d. from $\text{Unif}[0.2, 0.8]$, and stochastic rewards follow $\text{Beta}(\alpha, \beta)$ distributions with concentration $\kappa = 10$, where $\alpha = \mu\kappa$ and $\beta = (1-\mu)\kappa$ for mean μ.[4] We also set $f(x) = x$. We report averages over 10 independent runs with ± 1 standard deviation bands.

6.2 Binary Combinatorial Bandits

With $M = 5$, $K = 1$, $|\mathcal{F}| = 19$ (60% of 31 possible actions), and $T = 150{,}000$ (Figure 1), `mCombinatorialUCB-A` attains lowest cumulative regret (≈15k), leveraging synchronized confidence bounds. `mCombinatorialUCB-B` shows higher variance due to sabotage events that temporarily eliminate feasible actions, consistent with the additive $|\mathcal{F}|$ term in Theorem 9. `mDSEE` displays periodic exploration phases at each power of 2, creating its characteristic staircase pattern in cumulative regret with 0 regret during later commit phases.

6.3 Multi-Arm Combinatorial Bandits

With $M = 4$, $K = 3$, $|\mathcal{F}| = 51$ (20% of 255 possible actions), and $T = 500{,}000$, cumulative regret increases roughly K^2-fold compared to the binary case (Figure 2). `mCombinatorialUCB-A` maintains its advantage

[4] Note that any value drawn from Beta is in $[0, 1]$ which is $(1/4)$-subGaussian by Hoeffding's Lemma. See detailed proof in Appendix.

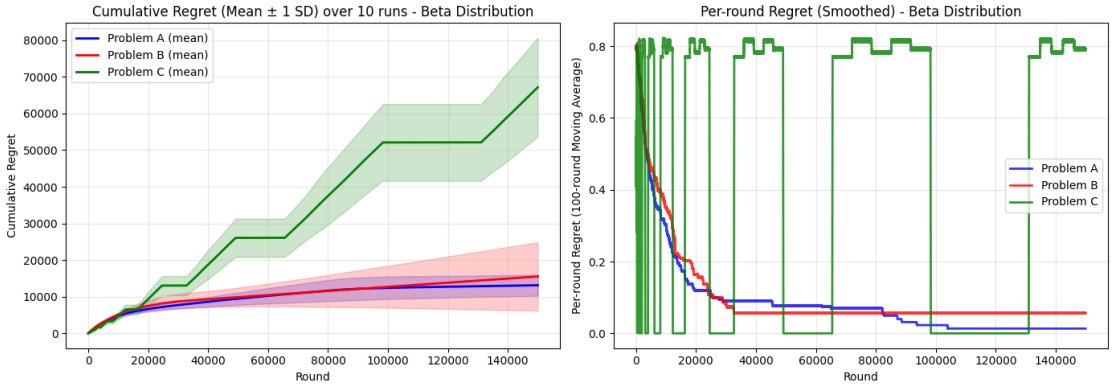


Figure 1: Binary setting: $M = 5$, $K = 1$, $|\mathcal{F}| = 19$, $T = 150{,}000$.

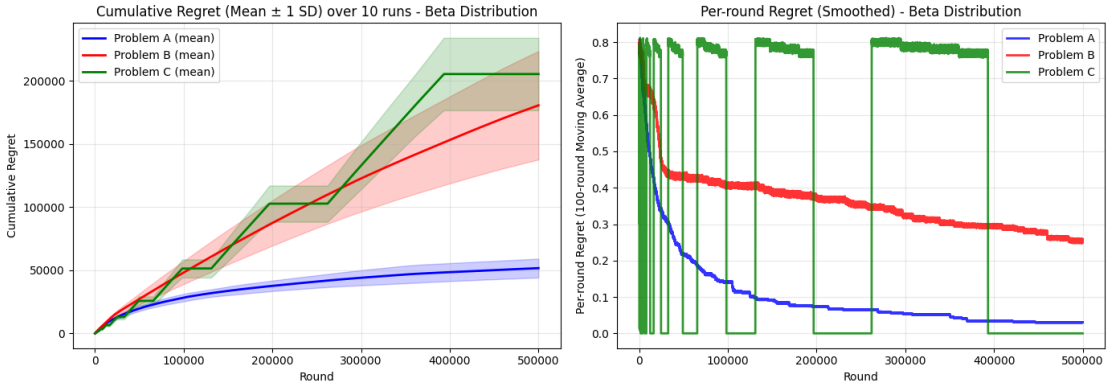


Figure 2: Multi-arm setting: $M = 4$, $K = 3$, $|\mathcal{F}| = 51$, $T = 500{,}000$.

(≈50k regret), while `mCombinatorialUCB-B` reaches ≈180k with increased sabotage complexity and more difficulty eliminating arms. `mDSEE` shows the steepest scaling (≈220k), requiring K times more exploration per epoch as predicted by the KM factor in Theorem 10.

Code Reproducibility Complete implementation and experimental code to reproduce all results are available at `https://anonymous.4open.science/status/combinatorial_bandits_code-FE60`.

7 Conclusions and Future Work

We studied combinatorial bandits under multiplayer information asymmetry, revealing that coordination without communication is achievable with only polynomial overhead in the number of players and arms. Our key insight is that pre-agreed exploration orderings (Problem A) or observable deviations (Problem B) can substitute for explicit communication while maintaining near-optimal $\tilde{\mathcal{O}}(\sqrt{T})$ regret, matching the single-player setting up to problem-dependent factors.

The transition from binary to multi-arm selection ($K = 1$ to $K > 1$) introduces only a multiplicative K^2 factor in regret, suggesting that the fundamental difficulty lies in player coordination rather than arm complexity. Notably, our gap-independent bounds significantly improve upon prior gap-dependent results in combinatorial settings, removing the potentially exponential dependence on $1/\Delta_{\min}^2$.

Several open questions remain:

1. Are our upper bounds tight, particularly considering the M^2 dependence?

2. Can the additive $|\mathcal{F}|$ term in Problem B be removed or is sabotage-based coordination inherently costly?

3. For Problem C, can we achieve gap-independent polylogarithmic regret without the exploration function $f(\cdot)$?

Finally, extending to time-varying feasible sets $\mathcal{F}_t$ or partial observability between Problems B and C presents interesting directions for future work.

References

Baruch Awerbuch and Robert Kleinberg. Competitive collaborative learning. *Journal of Computer and System Sciences*, 74(8):1271–1288, 2008.

Yogev Bar-On and Yishay Mansour. Individual regret in cooperative nonstochastic multi-armed bandits. *Advances in Neural Information Processing Systems*, 32, 2019.

Ilai Bistritz and Amir Leshem. Distributed multi-player bandits-a game of thrones approach. *Advances in Neural Information Processing Systems*, 31, 2018.

Rémi Bonnefoi, Lilian Besson, Christophe Moy, Emilie Kaufmann, and Jacques 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.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 2013.

Simina Brânzei and Yuval Peres. Multiplayer bandit learning, from competition to cooperation. In *Conference on Learning Theory*, pages 679–723. PMLR, 2021.

Mario Bravo, David Leslie, and Panayotis Mertikopoulos. Bandit learning in concave n-person games. *Advances in Neural Information Processing Systems*, 31, 2018.

Sarah H Cen and Devavrat Shah. Regret, stability & fairness in matching markets with bandit learners. In *International Conference on Artificial Intelligence and Statistics*, pages 8938–8968. PMLR, 2022.

Nicolo Cesa-Bianchi and Gábor Lugosi. Combinatorial bandits. *Journal of Computer and System Sciences*, 78(5):1404–1422, 2012.

Nicolò Cesa-Bianchi, Tommaso Cesari, and Claire Monteleoni. Cooperative online learning: Keeping your neighbors updated. In *Algorithmic learning theory*, pages 234–250. PMLR, 2020.

Nicol'o Cesa-Bianchi, Claudio Gentile, Yishay Mansour, and Alberto Minora. Delay and cooperation in nonstochastic bandits. In *Conference on Learning Theory*, pages 605–622. PMLR, 2016.

William Chang and Aditi Karthik. Multiplayer information asymmetric bandits in metric spaces. *arXiv preprint arXiv:2503.08004*, 2025.

William Chang and Yuanhao Lu. Optimal cooperative multiplayer learning bandits with noisy rewards and no communication. In *2024 IEEE 63rd Conference on Decision and Control (CDC)*, pages 823–830. IEEE, 2024.

William Chang and Yuanhao Lu. Multiplayer information asymmetric contextual bandits. *arXiv preprint arXiv:2503.08961*, 2025.

William Chang, Mehdi Jafarnia-Jahromi, and Rahul Jain. Online learning for cooperative multi-player multi-armed bandits. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 7248–7253. IEEE, 2022.

Shouyuan Chen, Tian Lin, Irwin King, Michael R Lyu, and Wei Chen. Combinatorial pure exploration of multi-armed bandits. *Advances in neural information processing systems*, 27, 2014.

Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework and applications. In *International conference on machine learning*, pages 151–159. PMLR, 2013.

Richard Combes, Mohammad Sadegh Talebi Mazraeh Shahi, Alexandre Proutiere, et al. Combinatorial bandits revisited. *Advances in neural information processing systems*, 28, 2015.

Yi Gai, Bhaskar Krishnamachari, and Rahul Jain. Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation. In *2010 IEEE Symposium on New Frontiers in Dynamic Spectrum (DySPAN)*, pages 1–9. IEEE, 2010.

Yi Gai, Bhaskar Krishnamachari, and Rahul Jain. Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE/ACM Transactions on Networking*, 20(5):1466–1478, 2012.

Amélie Heliou, Johanne Cohen, and Panayotis Mertikopoulos. Learning with bandit feedback in potential games. *Advances in Neural Information Processing Systems*, 30, 2017.

Meena Jagadeesan, Alexander Wei, Yixin Wang, Michael Jordan, and Jacob Steinhardt. Learning equilibria in matching markets from bandit feedback. *Advances in Neural Information Processing Systems*, 34:3323–3335, 2021.

Xiaohan Kang, Hong Ri, Mohd Nor Akmal Khalid, and Hiroyuki Iida. Addictive games: Case study on multi-armed bandit game. *Information*, 12(12):521, 2021.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. On distributed cooperative decision-making in multiarmed bandits. In *2016 European Control Conference (ECC)*, pages 243–248. IEEE, 2016.

Lydia T Liu, Horia Mania, and Michael Jordan. Competing bandits in matching markets. In *International Conference on Artificial Intelligence and Statistics*, pages 1618–1628. PMLR, 2020.

Lydia T Liu, Feng Ruan, Horia Mania, and Michael I Jordan. Bandit learning in decentralized matching markets. *Journal of Machine Learning Research*, 22(211):1–34, 2021.

Xutong Liu, Siwei Wang, Jinhang Zuo, Han Zhong, Xuchuang Wang, Zhiyong Wang, Shuai Li, Mohammad Hajiesmaili, John Lui, and Wei Chen. Combinatorial multivariant multi-armed bandits with applications to episodic reinforcement learning and beyond. *arXiv preprint arXiv:2406.01386*, 2024.

Weichao Mao. *Multi-agent reinforcement learning for nonzero-sum Markov games*. PhD thesis, University of Illinois at Urbana-Champaign, 2024.

David Martínez-Rubio, Varun Kanade, and Patrick Rebeschini. Decentralized cooperative stochastic bandits. 2019.

Santiago Ontanón. The combinatorial multi-armed bandit problem and its application to real-time strategy games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 9, pages 58–64, 2013.

Santiago Ontanón. Combinatorial multi-armed bandits for real-time strategy games. *Journal of Artificial Intelligence Research*, 58:665–702, 2017.

Alexandre Proutiere and Po-An Wang. An optimal algorithm for multiplayer multi-armed bandits. *arXiv preprint arXiv:1909.13079*, 2019.

Abishek Sankararaman, Soumya Basu, and Karthik Abinav Sankararaman. Dominate or delete: Decentralized competing bandits in serial dictatorship. In *International Conference on Artificial Intelligence and Statistics*, pages 1252–1260. PMLR, 2021.

Antoine Scheid, Daniil Tiapkin, Etienne Boursier, Aymeric Capitaine, El Mahdi El Mhamdi, Éric Moulines, Michael I Jordan, and Alain Durmus. Incentivized learning in principal-agent bandit games. *arXiv preprint arXiv:2403.03811*, 2024.

Balazs Szorenyi, Róbert Busa-Fekete, István Hegedus, Róbert Ormándi, Márk Jelasity, and Balázs Kégl. Gossip-based distributed stochastic bandit algorithms. In *International Conference on Machine Learning*, pages 19–27. PMLR, 2013.

Po-An Wang, Alexandre Proutiere, Kaito Ariu, Yassir Jedra, and Alessio Russo. Optimal algorithms for multiplayer multi-armed bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 4120–4129. PMLR, 2020.

Siwei Wang and Jun Zhu. Thompson sampling for (combinatorial) pure exploration. In *International Conference on Machine Learning*, pages 23470–23483. PMLR, 2022.

8 Beta Distributions are 1-subGaussian via Hoeffding's Lemma

Definition 11 (σ^2-subGaussian) *A centered random variable X is σ^2-subGaussian if*

$$\mathbb{E}\, e^{\lambda(X-\mathbb{E}X)} \;\le\; \exp\!\Big(\tfrac{\lambda^2\sigma^2}{2}\Big) \quad \textit{for all } \lambda \in \mathbb{R}.$$

Equivalently, X has tails dominated by a Gaussian with variance proxy σ^2.

Theorem 12 (Hoeffding's Lemma) *Let X be any real-valued random variable with $a \le X \le b$ almost surely. Then*

$$\mathbb{E}\, e^{\lambda(X-\mathbb{E}X)} \;\le\; \exp\!\Big(\tfrac{\lambda^2(b-a)^2}{8}\Big) \quad \textit{for all } \lambda \in \mathbb{R}.$$

In particular, $X - \mathbb{E}X$ is sub-Gaussian with variance proxy $\sigma^2 = (b-a)^2/4$. (Boucheron et al., 2013, Ch. 2).

Proposition 13 (Beta is 1-subGaussian) *If $X \sim \mathrm{Beta}(\alpha, \beta)$, then $X - \mathbb{E}X$ is sub-Gaussian with variance proxy $\sigma^2 \le \frac{1}{4}$.*

Proof: If $X \sim \mathrm{Beta}(\alpha, \beta)$ with $\alpha, \beta > 0$, then X is supported on the interval $[0, 1]$.

Applying Theorem 12 with $(a, b) = (0, 1)$ gives

$$\mathbb{E}\, e^{\lambda(X-\mathbb{E}X)} \;\le\; \exp\!\Big(\tfrac{\lambda^2}{8}\Big) = \exp\!\Big(\tfrac{\lambda^2}{2} \cdot \tfrac{1}{4}\Big),$$

so $X - \mathbb{E}X$ is $\sigma^2 = \frac{1}{4}$-subGaussian.

Since being σ^2-subGaussian is monotone in σ^2, $X - \mathbb{E}X$ is also 1-subGaussian (a looser but valid proxy). □

9 Important Lemmas

We present the proof of Lemma 2.

Proof: By De Morgan's Law, the complement of the good event G^c is:

$$G^c = \left(\bigcap_{t=1}^{T} \bigcap_{a=1}^{M} G_a(t)\right)^c = \bigcup_{t=1}^{T} \bigcup_{a=1}^{M} G_a(t)^c \tag{9}$$

which also means at least one arm at some time t violates the UCB bound:

$$G_a(t)^c = \{\mu_a > \mathrm{UCB}_a(t-1, \delta)\}$$

Setting $\delta = \frac{1}{T^2}$

$$\mathrm{UCB}_a(t-1, \delta) = \widehat{\mu}_a(t-1) + \sqrt{\frac{4 \log T}{n_a(t-1)}}$$

Using the previous proof:

$$\Pr\left(G_a(t)\right) = \Pr\left(\mu_a \le \mathrm{UCB}_a(t-1, \delta)\right) \ge 1 - \delta$$

$$\Pr\left(G_a(t)^c\right) = \Pr\left(\mu_a > \mathrm{UCB}_a(t-1, \delta)\right) \le \delta = \frac{1}{T^2}$$

Then, by using union bound:

$$\Pr(G^c) = \Pr\left(\bigcup_{t=1}^{T} \bigcup_{a=1}^{M} G_a(t)^c\right) \le \sum_{t=1}^{T} \sum_{a=1}^{M} \Pr(G_a(t)^c)$$

$$\Pr(G^c) \leq T \cdot M \cdot \frac{1}{T^2} = \frac{M}{T} \tag{10}$$

Thus, the upper bound of $Pr(G^c)$ is

$$\min\left(\frac{M}{T}, 1\right).$$

□

10 Proofs for the 0-1 Problem

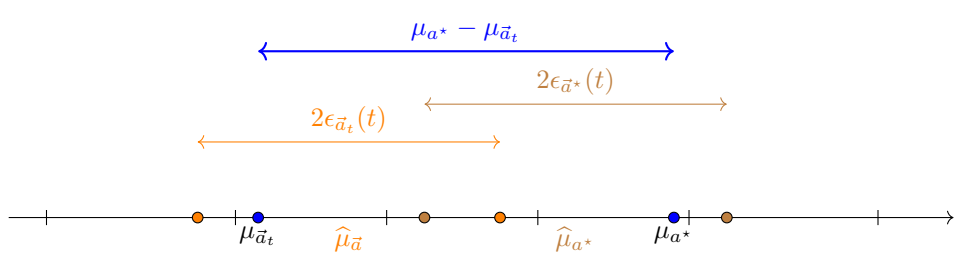
10.1 The Interval Approach for the 0-1 Problem (Problem A and Problem B)

We present the proof of Theorem 4.

Proof: Let $\mu^\star$ be the reward of the most optimal arm to pull, $X_{\vec{a}_t}$ be the arm that is pulled in round t by all players collectively.

$$\begin{aligned}
R_T &\leq \sum_{t=1}^{T} \mathbb{E}[\mu^\star - X_{\vec{a}_t}] && (11)\\
&= \sum_{t=1}^{T} \mathbb{E}[\mu^\star - X_{\vec{a}_{n_t}}|G]P(G) + \sum_{t=1}^{T} \mathbb{E}[\mu^\star - X_{\vec{a}_t}|G^c]P(G^c) && (12)\\
&= P(G)\sum_{t=1}^{T} \mathbb{E}[\mu^\star - X_{\vec{a}_t}|G] + P(G^c)\cdot\sum_{t=1}^{T} \mathbb{E}[\mu^\star - X_{\vec{a}_t}|G^c] && (13)\\
&\leq \sum_{t=1}^{T} \mathbb{E}[\mu^\star - \mu_{\vec{a}_t}|G] + M && (14)
\end{aligned}$$

where G refers to the "good event", where the true mean that for reach player i, their individual arm $\mu_{a_i} \in [\text{LCB}_{a_i}, \text{UCB}_{a_i}]$. The equality at Eq(7) comes from the Total Law of Expectation, and the one at Eq(9) relies on the facts that 1) $P(G) < 1$, 2) $P(G^c) \leq \frac{M}{T}$ by lemma 1, and 3) $\mu^\star - \vec{a}_t \leq 1$, since in the combinatorial bandit setting, the reward of any $\vec{a} \in [0, 1]$, so $\sum_{t=1}^{T} \mathbb{E}[\mu^\star - \vec{a}_t|G^c] \leq T$.



Denote the interval intended to locate $\mu_{\vec{a}}(t)$ as $I_{\vec{a}}(t)$, and the interval intended to locate $\mu_{\vec{a}^\star(t)}$, the estimated mean of the optimal action (which is unknown), as $I_{\vec{a^\star(t)}}$. At each round t, the players pull $\vec{a_t}$; based on the algorithm, $\vec{a}$ must not be "dominated" by $\vec{a^\star}$, i.e., the LCB of $\vec{a_t}$ cannot be lower than the UCB of $\vec{a_\star}$, and $I_{\vec{a}}$ and $I_{\vec{a}^\star}$ should not be disjoint. As conditioning on the good event G,

$$\mu_{\vec{a}}(t) = \mathbb{E}[X_{\vec{a}_t}] \in I_{\vec{a}}, \quad \mu^\star = \mathbb{E}[X_{\vec{a}^\star}] \in I_{\vec{a^\star}}$$

Therefore, for each t, the expected difference between both mean between $\vec{\mu^\star}$ and $X_{\vec{a}_t}$ is less than the sum of both estimated intervals, conditioning on the good event. Hence,

$$\mathbb{E}[\mu^\star - X_{\vec{a}_t}|G] \leq 2\epsilon_{\vec{a}_t}(t) + 2\epsilon_{\vec{a}^*}(t) \tag{15}$$

Therefore, summing the above by T we obtain for the first term,

$$\sum_{t=1}^{T} 2\epsilon_{\vec{a}_t}(t) = \sum_{t=1}^{T}\sum_{a_i\in\vec{a}} 2\epsilon_{a_i}(t) \leq \sum_{j=1}^{M} 2\sqrt{4\log T}\sum_{t=1}^{T}\frac{1}{\sqrt{t}} = 2M\sqrt{4\log T}\sum_{t=1}^{T}\frac{1}{\sqrt{t}}$$

The first inequality follows from the fact that for each individual arm, the definition of $\epsilon_{a_i}(t)$ is $\sqrt{\frac{4\log T}{n_{a_i}(t)}}$. Note that the numerator is constant for all rounds and the denominator increments by 1 each time it is pulled. Therefore we are essentially summing up $\sqrt{\frac{4\log T}{1}} + \sqrt{\frac{4\log T}{2}} + \cdots$. Since each arm is pulled at most T times, this sum terminates before reaching the term $\sqrt{\frac{4\log T}{T}}$. On the other hand, for the second term we obtain

$$\sum_{t=1}^{T} 2\epsilon_{a_*}(t) \leq 2\sum_{i=1}^{M} M\sum_{t=1}^{T}\sqrt{\frac{4\log T}{t}} = 2M^2\sqrt{4\log T}\sum_{t=1}^{T}\sqrt{\frac{1}{t}}.$$

The first equality follows from the fact that since we are cycling through the desired sets $S_1, ..., S_M$ this means that we are pulling the players arm in $\vec{a}^\star$ at least once every M rounds. This means that when summing $\epsilon_{\vec{a}^\star}(t)$ across t, we are adding $\sqrt{\frac{4\log T}{n}}$ at most M times before incrementing n by 1.

To obtain an upper bound of the RHS above, we approximate the discrete sum by the integral of a decreasing function. Since $\frac{1}{t}$ is decreasing, the integral over $[0, T]$ dominates the corresponding Riemann sum.

We have

$$\sum_{t=1}^{T}\sqrt{\frac{1}{t}} \leq \int_0^T \frac{1}{\sqrt{t}}\,dt = 2\sqrt{T}.$$

Therefore,

$$R_T \leq 2M\sqrt{4\log T}\cdot 2\sqrt{T} + 2M^2\sqrt{4\log T}\cdot 2\sqrt{T} + M = 8M(M+1)\sqrt{T\log T} + M.$$

□

10.2 mDSEE for Problem C

We present the proof of Theorem 7

Proof: We divide the regret into the exploration phase and the committing phase. In the exploration phase, the regret is upper bounded by the total number of pulls during this phase. Since we are pulling M arms, we have a total regret of

$$R_{\text{explore}} \leq \lfloor \log_2(T)\rfloor f(\lfloor \log_2(T)\rfloor)M$$

Define the following good event for each combinatorial arm $a \in \mathcal{F}$:

$$G_a(t) = \{|\widehat{\mu}_a - \mu_a| < \epsilon\}$$

Choose ϵ so that

$$\epsilon = \frac{1}{2}\min_{\vec{a}\in\mathcal{F}} \Delta_{\vec{a}}$$

where $\Delta_a = \mu^* - \mu_a$ is the suboptimality gap for arm a.

For the regret of the committing phase we have

$$
\begin{aligned}
R_{\text{committ}} &\leq \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{I}\left\{\left(\bigcap_{a\in\mathcal{F}} G_{a_i}(t)\right)^c\right\}\right] && (16)\\
&= \sum_{t=1}^{T} P\left(\bigcup_{a\in\mathcal{F}} G_{a_i}(t)^c\right) && (17)\\
&\leq \sum_{t=1}^{T}\sum_{i=1}^{M} P(G_{a_i}(t)^c) && (18)\\
&= \sum_{t=1}^{T}\sum_{i=1}^{M} P(\{|\widehat{\mu}_a - \mu_a| > \epsilon\}) && (19)\\
&\leq \sum_{t=1}^{T}\sum_{a\in\mathcal{F}} e^{-\frac{n_a(T)\epsilon^2}{2}} && (20)\\
&\leq \sum_{t=1}^{T} M e^{-\frac{M_0(t)\log_2(t)\epsilon^2}{2}} && (21)\\
&= \sum_{t=1}^{T} M t^{-\frac{M_0(t)\epsilon^2}{2\log 2}} && (22)
\end{aligned}
$$

where $M_0(t)$ is a lower bound on the number of times each base arm in the support of any $a \in \mathcal{F}$ has been sampled (assuming uniform exploration). Since $\log_2(t)$ grows with t, this quantity decays super-polynomially.

Thus, the committing phase regret is bounded as:

$$
R_{\text{commit}} \leq \sum_{t=1}^{T} M \cdot t^{-c},
$$

for some constant $c > 0$, depending on $\epsilon^2 M_0(t)$. This sum converges and is at most $O(1)$.

Therefore, the total regret is:

$$
\begin{aligned}
R_T &\leq R_{\text{explore}} + R_{\text{commit}}\\
&\leq \lfloor \log_2(T) \rfloor f(\lfloor \log_2(T) \rfloor) M + O(1)\\
&= O\left(\lfloor \log_2(T) \rfloor f(\lfloor \log_2(T) \rfloor) M\right)
\end{aligned}
$$

□

11 The Joint Arm Problem Pseudocode

The pseudocode for Joint arm Problem A is given in Algorithm 4.

Algorithm 4 `mCombinatorialUCB-Joint-A`

Require: Horizon T; player sets $S_1, \ldots, S_M$ with fixed internal orders; per-player choice lists $D_1, \ldots, D_M$; feasible family $\mathcal{S}_i$ for each i

Ensure: Cyclic elimination and UCB updates for individual and combinatorial arms

Initialize: For each individual arm a_i^j: $n_{a_i^j}(0) \leftarrow 0$, $\hat{\mu}_{a_i^j}(0) \leftarrow 0$, $\mathrm{UCB}_{a_i^j}(0) \leftarrow +\infty$, $\mathrm{LCB}_{a_i^j}(0) \leftarrow -\infty$
$i \leftarrow 1$ ▷ current player / set index
for $t = 1$ **to** T **do**
  Let S_i be the next desired set and let $\vec{a}$ be the next combinatorial arm in (6)
  Let a_i^k be the arm currently considered in D_i
  $\vec{a}_t \leftarrow \vec{a}$

  Filter individual choice a_i^k using UCB/LCB
  while exists $a' \in D_i$ **such that** $\mathrm{UCB}_{a_i^k}(t-1) < \mathrm{LCB}_{a'}(t-1)$ **do**
    Remove a_i^k from D_i
    Set a_i^k to the next element of D_i (by the fixed order)
  end while

  Filter current combinatorial arm within S_i
  while exists $\vec{a}' \in \mathcal{S}_i$ **such that** $\mathrm{UCB}_{\vec{a}_t}(t-1) < \mathrm{LCB}_{\vec{a}'}(t-1)$ **do**
    Remove $\vec{a}_t$ from all S_j that contain it
    Set $\vec{a}_t$ to the next arm in S_i (by the fixed order)
  end while

  Update individual UCB indices (for pulled components)
  for all $a_i^j \in \vec{a}_t$ **do**
    Observe reward $X_{a_i^j}(t)$
    $n_{a_i^j}(t) \leftarrow n_{a_i^j}(t-1) + 1$
    $\hat{\mu}_{a_i^j}(t) \leftarrow \big(n_{a_i^j}(t-1)\hat{\mu}_{a_i^j}(t-1) + X_{a_i^j}(t)\big) \,/\, n_{a_i^j}(t)$
    $\mathrm{UCB}_{a_i^j}(t) \leftarrow \hat{\mu}_{a_i^j}(t) + \sqrt{4 \log T \,/\, n_{a_i^j}(t)}$
    $\mathrm{LCB}_{a_i^j}(t) \leftarrow \hat{\mu}_{a_i^j}(t) - \sqrt{4 \log T \,/\, n_{a_i^j}(t)}$
  end for

  Update combinatorial UCB indices (for current set)
  for all $\vec{b} \in S_i$ **do**
    $\mathrm{UCB}_{\vec{b}}(t) \leftarrow \sum_{a_q \in \vec{b}} \mathrm{UCB}_{a_q}(t)$
    $\mathrm{LCB}_{\vec{b}}(t) \leftarrow \sum_{a_q \in \vec{b}} \mathrm{LCB}_{a_q}(t)$
  end for
  $i \leftarrow \big(i \bmod M\big) + 1$ ▷ advance to next set cyclically
end for

The pseudocode for Joint Arm Problem B is given in Algorithm 5.

Algorithm 5 `mCombinatorialUCB-Joint-B`

Require: Horizon T; player sets $S_1, \ldots, S_M$ with fixed internal orders; per-player choice lists $D_1, \ldots, D_M$; feasible family $\mathcal{S}_i$ for each i

Ensure: Joint elimination via player-triggered sabotage and UCB updates

Initialize: For each individual arm a_i^j: $n_{a_i^j}(0) \leftarrow 0$, $\hat{\mu}_{a_i^j}(0) \leftarrow 0$, $\mathrm{UCB}_{a_i^j}(0) \leftarrow +\infty$, $\mathrm{LCB}_{a_i^j}(0) \leftarrow -\infty$
$i \leftarrow 1$ ▷ current player / set index
for $t = 1$ **to** T **do**
  Let S_i be the next desired set and let $\vec{a}$ be the next combinatorial arm in (6); let a_i^k be the element currently considered in D_i
  $\vec{a}_t \leftarrow \vec{a}$
  Player-i local check (individual elimination)
  if exists $a' \in D_i$ **such that** $\mathrm{UCB}_{a_i^k}(t-1) < \mathrm{LCB}_{a'}(t-1)$ **then**
    Player i pulls any arm different from a_i^k ▷ sabotage
    All players remove a_i^k from D_i
    Set a_i^k to the next element of D_i (by fixed order)
  end if

  Other-player check (combinatorial elimination)
  if exists player $j \neq i$ and **exists** $\vec{a}' \in \mathcal{S}_i$ **such that** $\mathrm{UCB}_{\vec{a}_t}(t-1) < \mathrm{LCB}_{\vec{a}'}(t-1)$ **then**
    Player j chooses an action so that the resulting joint action is not $\vec{a}_t$
    All players remove $\vec{a}_t$ from every S_r that contains it
    Set $\vec{a}_t$ to the next arm in S_i (by fixed order)
  end if

  Update individual indices (for components pulled in $\vec{a}_t$)
  for all $a_i^j \in \vec{a}_t$ **do**
    Observe reward $X_{a_i^j}(t)$
    $n_{a_i^j}(t) \leftarrow n_{a_i^j}(t-1) + 1$
    $\hat{\mu}_{a_i^j}(t) \leftarrow \big(n_{a_i^j}(t-1)\hat{\mu}_{a_i^j}(t-1) + X_{a_i^j}(t)\big) \,/\, n_{a_i^j}(t)$
    $\mathrm{UCB}_{a_i^j}(t) \leftarrow \hat{\mu}_{a_i^j}(t) + \sqrt{4 \log T \,/\, n_{a_i^j}(t)}$
    $\mathrm{LCB}_{a_i^j}(t) \leftarrow \hat{\mu}_{a_i^j}(t) - \sqrt{4 \log T \,/\, n_{a_i^j}(t)}$
  end for

  Update combinatorial indices (for current set S_i)
  for all $\vec{b} \in S_i$ **do**
    $\mathrm{UCB}_{\vec{b}}(t) \leftarrow \sum_{a_q \in \vec{b}} \mathrm{UCB}_{a_q}(t)$
    $\mathrm{LCB}_{\vec{b}}(t) \leftarrow \sum_{a_q \in \vec{b}} \mathrm{LCB}_{a_q}(t)$
  end for
  $i \leftarrow \big(i \bmod M\big) + 1$ ▷ advance to next set cyclically
end for

The pseudocode of the Joint Arm Problem C in Algorithm 6

Algorithm 6 `JointArm-mDSEE Algorithm for Problem C`

Require: Number of players M, number of individual arms K, combinatorial set $\mathcal{F}$, monotone function $f(\lambda)$ with $f(\lambda) \to \infty$

Ensure: Identification of high-reward joint arms

```
t ← 1, λ ← 1
while t ≤ T do
    Exploration phase:
    for m = 1 to M do
        for k = 1 to K do
            for p = 1 to M do
                Pull the representative joint arm a⃗ ∈ S_p
                if p = m then
                    Pull a_k^p from D_p
                else
                    Pull the individual arm in D_p with highest μ̂_{a_p^k}
                end if
                Observe reward X^{(p)}_{a_p^k}(t)
            end for
            Compute empirical means μ̂^{(m)}(a_m^k)
        end for
    end for
    Commit phase:
    for m = 1 to M do
        Compute empirical means μ̂_{a⃗} for all a⃗ ∈ F based on local data
        a⃗^m_commit ← arg max_{a⃗∈F} μ̂^m_{a⃗}
        if m is included in a⃗^m_commit then
            Player m pulls the best arm in D_m according to local estimates
        end if
    end for
    λ ← λ + 1
    t ← t + 1
end while
```

11.1 Proofs for the Interval Approach for Problem A and Problem B

We present the proof of Theorem 8

Proof: Recall that the upper end of the interval of the combinatorial arm is given by some maximum over the individual actions for the players contribution to $\vec{a}_t$. Suppose the arg maximum is given by $\vec{a}_t^\star$. Then this exceeds the lower end of the interval of the optimal arm $\vec{a}^\star$. Furthermore, each individual arm for $\vec{a}_t$ has an interval that overlaps with the individual arms of $\vec{a}_t^\star$. Under the good event this implies

$$\mathbb{E}[\mu^\star - X_{a_{n_t}}|G] \leq 2\sum_{a\in\vec{a}_t}\epsilon_a(t) + \sum_{a\in\vec{a}_t^\star}\epsilon_a(t) + \sum_{a\in\vec{a}^\star}\epsilon_a(t) \tag{23}$$

For the first term, summing across $t = 1$ to $t = T$, we obtain

$$2\sum_{t=1}^{T}\sum_{a\in\vec{a}_t}\epsilon_a(t) = 2\sum_{k=1}^{K}\sum_{m=1}^{M}\sum_{s=1}^{n_{a_m^k}(T)}\sqrt{\frac{4\log T}{s}} = 2\sum_{k=1}^{K}\sum_{m=1}^{M}2\sqrt{\log T}\sum_{s=1}^{T}\frac{1}{\sqrt{s}} = 4KM\sqrt{T\log T}$$

For the second term, we have

$$2\sum_{t=1}^{T}\sum_{a\in\vec{a}_t^\star}\epsilon_a(t) = 2K\sum_{k=1}^{K}\sum_{m=1}^{M}\sum_{s=1}^{n_{a_m^k}(T)}\sqrt{\frac{4\log T}{s}} = 2K\sum_{k=1}^{K}\sum_{m=1}^{M}2\sqrt{\log T}\sum_{s=1}^{T}\frac{1}{\sqrt{s}} = 4K^2M\sqrt{T\log T}$$

The first equality follows that for $a \in \vec{a}_t^*$ the corresponding $\sqrt{\frac{4\log T}{s}}$ term is added at most K times for the same s. This is because for a fixed combinatorial arm in S_m we cycle through each of player m's actions once. Finally, for the third term, we have

$$2\sum_{t=1}^{T}\sum_{a\in\vec{a}_t^\star}\epsilon_a(t) = 2KM\sum_{k=1}^{K}\sum_{m=1}^{M}\sum_{s=1}^{n_{a_m^k}(T)}\sqrt{\frac{4\log T}{s}} = 2KM\sum_{k=1}^{K}\sum_{m=1}^{M}2\sqrt{\log T}\sum_{s=1}^{T}\frac{1}{\sqrt{s}} = 4K^2M^2\sqrt{T\log T}$$

The first equality follows that for $a \in \vec{a}^*$ the corresponding $\sqrt{\frac{4\log T}{s}}$ term is added at most MK times for the same s. This is because a fixed combinatorial arm in S_m we cycle through each of player m's actions once, which takes K rounds, and there are M such S_m's. □

11.2 Joint mDSEE alrogithm proof

We present the proof of Theorem 10

Proof: We divide the regret into phase; the exploration phase and the committing phase. In the exploration it is upper bounded by the total number of pulls during this phase. Since we are pulling M arms, we have a total of

$$R_{\text{explore}} \leq \lfloor\log_2(T)\rfloor f(\lfloor\log_2(T)\rfloor)M$$

Define the following good event for each combinatorial arm $a \in \mathcal{F}$

$$G_a(t) = \{|\widehat{\mu}_a - \mu_a| < \epsilon\}$$

Choose ϵ so that

$$\epsilon = \frac{1}{2}\min_{\vec{a}\in\mathcal{F}}\Delta_{\vec{a}}$$

where $\Delta_a = \mu^* - \mu_a$ is the suboptimality gap for arm a.

For the regret of the committing phase we have

$$R_{\text{committ}} \leq \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{I}\left\{\left(\bigcap_{a\in\mathcal{F}} G_{a_m}(t)\right)^c\right\}\right] \tag{24}$$

$$= \sum_{t=1}^{T} P\left(\bigcup_{a\in\mathcal{F}} G_{a_m}(t)^c\right) \tag{25}$$

$$\leq \sum_{t=1}^{T}\sum_{m=1}^{M} P(G_{a_m}(t)^c) \tag{26}$$

$$= \sum_{t=1}^{T}\sum_{m=1}^{M} P(\{|\widehat{\mu}_a - \mu_a| > \epsilon\}) \tag{27}$$

$$\leq \sum_{t=1}^{T}\sum_{a\in\mathcal{F}} e^{-\frac{n_a(T)\epsilon^2}{2}} \tag{28}$$

$$\leq \sum_{t=1}^{T} M e^{-\frac{M_0(t)\log_2(t)\epsilon^2}{2}} \tag{29}$$

where $M_0(t)$ is a lower bound on the number of times each base arm in the support of any $a \in \mathcal{F}$ has been sampled (assuming uniform exploration). Since $\log_2(t)$ grows with t, this quantity decays super-polynomially.

Thus, the committing phase regret is bounded as:

$$R_{\text{committ}} \leq \sum_{t=1}^{T} M \cdot t^{-c},$$

for some constant $c > 0$, depending on $\epsilon^2 M_0(t)$. This sum converges and is at most $O(1)$.

Therefore, the total regret is:

$$\begin{aligned} R_T &\leq R_{\text{explore}} + R_{\text{committ}} \\ &\leq \lfloor \log_2(T) \rfloor f(\lfloor \log_2(T) \rfloor) M + O(1) \\ &= O\left(\lfloor \log_2(T) \rfloor f(\lfloor \log_2(T) \rfloor) M\right) \end{aligned}$$

□