
COORDINATING FOR CLICKS: LEARNING IN MULTI-AGENT INFORMATION ASYMMETRIC CASCADING BANDITS

William Chang

Department of Applied Mathematics
University of California, Los Angeles
Los Angeles, CA
<https://williamc.me/>

Steve

Department of Applied Mathematics
University of California, Los Angeles
Los Angeles, CA
<https://www.youtube.com/watch?v=dQw4w9WgXcQ>

ABSTRACT

We formulate a decentralized, cooperative multi-agent framework, studied in Chang et al. [2022], applied to the stochastic partial-monitoring cascading bandit problem first introduced in Kveton et al. [2015a]. The reward in each round depends on the joint-ranking "cascade" collectively taken by all learning agents. The objective is shared, but to make the coordination problem more challenging, we contend with three variants of information asymmetry: action asymmetry, where the overall joint-ranking is unobservable to all agents but the feedback received is common; reward asymmetry, where the overall ranking is observable, but feedback received by each agent is i.i.d.; and that with both action and reward asymmetry. For the first setting, we propose `mCascadeUCB`, and for the second setting, we propose `mCascadeUCB-Intervals`, with both algorithms achieving $O(\log T)$ gap-dependent regret within their respective settings. For the last setting, we propose `mCascadeDSEE`, which achieves close to $O(\log T)$ independent regret. We demonstrate our algorithms with experimental results.

<https://arxiv.org/pdf/1502.02763>

1 Introduction

In applications such as search engine optimization, network routing, or user recommendation, the *cascade model* is a prominent tool employed to describe user behavior or instances of partial feedback. User behavior research in the early twenty-first century examined how characteristics of implicit user feedback and engagement—click-through rate, dwell times, query chains, etc.—could be modeled and employed in recommendation or web search algorithms to improve performance by up to thirty-one percent relative to algorithms without this data (Agichtein et al. [2006], Radlinski and Joachims [2005]). Craswell et al. [2008] introduced the first notion of the "cascade model", a user behavior model that sought to account for user "position bias" in search rankings.

In essence, the model assumes a user that examines *from first to last* an ordered ranking or list of items. The user endows each item with an attraction probability, which is the likelihood the user is "attracted" by the item. An important assumption is the user will continue to examine the list until they are attracted by an item or they reach the end of the list. Additionally, if an item has attracted the user (often interpreted as a "click"), the user stops and does not examine the remaining items. Thus, this model incorporates "position bias", in that higher-ranked items are more likely to be observed or clicked by users.

Previous literature has extended this model to a reinforcement learning bandit problem. Kveton et al. [2015a] presented the first extension of the cascade model to a stochastic bandit setting. We rebuild the technical details in section 2.1, but primarily, they consider a learning agent whose goal is to minimize regret by recommending the top K items with cascade "click" feedback. In particular, they develop the first algorithms for the cascading setting, `CascadeUCB1` and `CascadeKL-UCB`, and prove gap-dependent regret upper bounds of $O(\log T)$. Many related works to the cascading setting have since been studied, which we examine further in section 1.

In a different vein, the study of multi-agent multi-armed bandits (MAMAB) has progressed in recent years, motivated by applications in areas like cognitive radio networks (Boursier and Perchet [2022]). In particular, one of the first

frameworks for a decentralized, cooperative multi-agent setting for general multi-armed bandits is presented in Chang et al. [2022] and Chang and Lu [2023]. They introduce the notion of all individual players' actions comprising a collective *joint-action* with its own distinct reward distribution. Thus, the common goal for all players is to minimize collective regret by pulling the most optimal joint-arm.

Furthermore, because of multi-agent dynamics, Chang et al. also introduce forms of information asymmetry between the players into their settings. First, that in which players are unable to observe the actions taken by other players, but receive identical rewards from the joint-action (action asymmetry); second, that in which players receive i.i.d. copies of the reward, but they can observe each other's actions (reward asymmetry); and finally, where players cannot observe each other's actions and receive i.i.d. copies of the reward (both action and reward asymmetry).

Our contribution. We make four contributions. First, we formulate a novel extension of the stochastic cascading bandit introduced in Kveton et al. [2015a] to a multi-agent setting, equipped with *joint-items* and *joint-rankings*. Previous work in this intersection (Yang et al. [2024]) has explored multi-agent systems with communication between players facilitated by a central server, but in our paper, our environment is decentralized and we allow no explicit communication between players during play. The setting and problem setup we study are elaborated further in section 2. Second, we increase the challenge of player coordination by considering three problems involving information asymmetries inspired by Chang et al. [2022], proposing algorithms for each. They are as follows:

- **Problem A Action Asymmetry:** We propose an ordering on the joint-items that allows the players to coordinate even without having to observe the actions of the other players. See Algorithm 1, mCascadeUCB.
- **Problem B: Reward Asymmetry** We propose an interval-based approach that allows players to successively eliminate suboptimal joint-items despite receiving different rewards. See Algorithm 2, mCascadeUCB-Intervals.
- **Problem C Action and Reward Asymmetry** We propose a variant of the explore-then-commit algorithm that achieves nearly optimal regret. See Algorithm 3, mCascadeDSEE.

Further descriptions of the algorithms can be found in section 3. Third, we prove gap-dependent regret upper bounds, notably for mCascadeUCB-Intervals in the reward-asymmetric setting, on the order of $O(\log T)$, and a gap-independent regret upper bound for mCascadeDSEE in the action and reward-asymmetric setting that achieves close to log regret. The details can be found in section 4. Finally, we showcase the performance of our algorithms with several experiments, the results of which can be found in section 5.

Related Works. Various advancements to the canonical single-agent cascading bandit in Kveton et al. [2015a] have been made. Thompson Sampling algorithms such as TS-Cascade (Cheung et al. [2019], Zhong et al. [2021]) and variance-aware confidence sets derived from Bernstein and Chernoff bounds match previous algorithms in the cascading setting (Vial et al. [2022]). Zong et al. [2016] formulate a *linear* cascading bandit, proposing algorithms that improve the regret dependence on L via the assumption that attraction probabilities can be approximated via a parameter vector.

Other popular cascading bandit variants are contextual cascading bandits (Li et al. [2016], Wang [2021], Choi et al. [2024]), where information or context vectors on the items are received; best-arm/top K item identification (Zhong et al. [2020]); non-stationary settings with abruptly changing attraction probabilities (Li and De Rijke [2019]); and even an extension to a cascading Markov decision process, where transitions to more auspicious states must be considered (Du et al. [2024]). As touched upon earlier, Yang et al. [2024] consider a *federated* contextual cascading bandit, where each user is *individually* served by an agent, in contrast to our setting where agents' actions comprise a *joint-ranking* for the user(s). Agent communication is also facilitated through a central server, while our decentralized setting features no communication and various information asymmetries, giving rise to a distinctly complex cooperation problem.

Modifications to the cascade model have been implemented into bandits as well. One is user and topic *clustering*, with frameworks featuring item "topics" and user "interests" with correlated attraction probabilities (Combes et al. [2015]) or even an underlying cluster graph where an edge denotes user similarity which can be learned while incurring $O(\sqrt{T})$ regret with the CLUB-Cascade algorithm (Li and Zhang [2018]). Katariya et al. [2016] extend a similar "Dependent Click Model" (DCM) to a bandit setting, which adds an aspect of "user satisfaction" that just a "click" might not imply. Mansoury et al. [2024] implement "exposure bias"—more "useful" recommendations pushed down by those consistently recommended—by discounting rewards to items clicked higher in the ranking. An item's attraction probability could depend on its variation from previous items', adding "diversity" Li et al. [2020], Becker et al. [2007]). Item "abandonment probabilities" (Cao et al. [2019]) remove the assumption that users will always continue examining. For a final exhibit, Wang et al. [2024] incorporate delayed click feedback.

Furthermore, our work intersects with the growing literature on the multi-agent or cooperative bandits problem first introduced by Awerbuch and Kleinberg [2008]. In this setting, M players have a collective goal of determining the best

action, and often, a graph represents the communication framework between players. Subsequent algorithms, such as ϵ -greedy variations (Szorenyi et al. [2013], Jin et al. [2023]), gossip UCB (Landgren et al. [2016], Martínez-Rubio et al. [2019]), or leader-follower DPE1 Wang et al. [2020] have been proposed. Other lines of multi-agent works involve players only observing the rewards of players within a neighboring distance Cesa-Bianchi et al. [2016] or realizing heterogeneous rewards while communicating information via a graph network Xu and Klabjan [2024]. Some multi-agent settings involve asynchronous actions, where only a subset of players are active at any time (Bonnefoi et al. [2017], Cesa-Bianchi et al. [2020]), or voting systems to select the best action in a shared network set (Shahrampour et al. [2017]). Dubey et al. [2020] consider a group of communicating agents selecting actions from their *individual* action sets, with the goal of minimizing total group regret.

The decentralized, limited communication, multi-agent setting introduced Chang et al. [2022] has been explored further as well, with cooperative multi-agent reinforcement learning algorithms following (Kao [2022], Mao et al. [2021], Mao et al. [2022]) that do not require communication among players or a central server during learning. Furthermore, information asymmetric RL settings have been explored, such as the leader-follower games in Kao et al. [2022] where only the follower observes the action of the leader while realizing the same reward, or extensions of information asymmetry to multi-agent contextual bandits in Chang and Lu [2024]. To the best of our knowledge, information asymmetry in a multi-agent cascading bandit setting has yet to be researched.

Various areas call for answers that lie within the intersection of cascading models and multi-agent reinforcement learning. The user-interaction and click framework (Craswell et al. [2008], Kveton et al. [2015a]) we adopt in this paper translates to a multi-agent problem when we consider complex, multi-faceted items or recommendations that involve the collaboration of multiple parties (e.g. content, thumbnails, and titles for YouTube videos; scripts and graphic design for advertisements, etc.). Another area that might find utility, particularly from the information asymmetry we study, is network routing. Kveton et al. [2015b] take this framework in single-agent cascading bandits, where the Bernoulli feedback is interpreted points in a chosen network route which are "down". *Multi-agent* reinforcement learning in particular has potential in multi-agent routing (Yamin and Permuter [2024], Zeng et al. [2020], Mammeri [2019]). Physical or cost constraints may hinder a communication or a central server for networks, calling for no-communication solutions. Additionally, a network path's efficiency or latency is highly dependent on the (potentially-unobservable) actions of other stations, and can vary from station to station, analogous to the joint-actions and information asymmetry we study. This leaves stations having to coordinate to maximize overall network performance—a cooperative multi-agent problem.

2 Preliminaries and Problem Statements

2.1 Single-Agent Cascading Bandits

First, we setup the standard cascade model and its inspired bandit. For any round t up to the horizon T , a learning agent plays a *ranking* or action $A = (a_1, \dots, a_K)$ which consists of a ordered list of K items chosen from the agent's ground set of items with cardinality L . A user then examines the ranking from the first to last item. The model (or user) endows each item $a_i \in E$ with an *attraction probability*, $\bar{w}(a_i) \in [0, 1]$, which is the likelihood item a_i "attracts" the user, and the user "clicks". This probability is assumed to be independent of the other items.

As previously introduced, an important assumption of the cascade bandit is that once the user has clicked one item $a_i \in A$, the user stops and does not examine the remaining items in A . In other words, items a_1, \dots, a_{i-1} are considered to be unattractive to the user for that A_t , while the attractiveness of items a_{i+1}, \dots, a_K are *unobserved*. The reward for the agent takes on a value of 1 if the user clicked on any item in the agent's ranking, and 0 if no items were clicked. Ultimately, for any A , a user can make at most one click and the agent's goal is to maximize the likelihood the user clicks on an item, which equates to choosing the $A^* \in \Pi_K(E)$ consisting of the K most attractive items such that $1 - \prod_{i=1}^K (1 - \bar{w}(a_i))$ is maximized.

2.2 Multi-Agent Extension of Cascading Bandits

Here we extend the single-agent cascading bandit problem and formulate the decentralized multi-agent setting by introducing a new framework allowing for *joint actions* by multiple agents.

Let M be the number of players and E^i denote the set of actions player i has access to. At the start of every round, each player picks K actions from their own actions set. This results in a *joint action or ranking* taken by these players which we denote using a tuple $A = (a_1, \dots, a_K)$ of K *joint items*. Each joint item a is some vector (a^1, \dots, a^M) where

$a^i \in E^i$.¹ Additionally, we make the assumption that for all players i , $|E^i| = L$. Thus, the number of distinct joint items by M players is L^M and the number of joint rankings with K distinct items is $\frac{(L^M)!}{(L^M-K)!}$. In standard no-communication multi-agent bandits, the players are allowed to agree on a strategy prior to the learning process; however, once the learning begins they cannot explicitly communicate. Therefore, we have the following modification from the single-agent setting.

Remark 1 *In the single agent setting, at every step, the player recommends K out of L items for the user to click on. For the multiplayer setting, this process will occur via players taking their own actions K times in an ordering of their choice. When the user clicks on an item the players will know the place number of the item, but not the joint item itself. e.g. they will know it's the 1st, 2nd, 3rd, etc. item that was recommended.*

We let $C_t \in \{1, \dots, K, \infty\}$ be this feedback of the item that was clicked by the user, where $C_t = \infty$ is equivalent to no click occurring. This remark is necessary for the information asymmetry in actions (see below problem A) to be meaningful.

A subtle point to mention is with how the bandit is structured, the multiplayer setting now creates a nontrivial possibility of *repeated* items. Recall in the single-player setting, an agent takes an action $A \in \Pi_K(E)$, which eliminates the possibility of item repetition. However, in the multiplayer setting, players can *intend* to play certain permutations of joint-items, but because they only control *their respective item*, this can result in a joint-ranking with joint-item repetition. While repetition ostensibly seems "legal" in the sense of a cascade model, it clashes with the intuition of applications such as a recommendation system. Fortunately, the setup in the following paragraph helps punish this repetition.

Let $B(p)$ denote the Bernoulli distribution with parameter p . Our user assigns a Bernoulli distribution to each of the L^M joint items which remains fixed across time. For a joint item, an outcome of 1 is interpreted as attractive (where the user will click on the item) and 0 as unattractive (where the user will skip over that item). Define $\bar{w}(e)$ to be the true attraction probability for joint item e . For each round t , let $\mathbf{w}_t \in \{0, 1\}^{L^M}$ be drawn from the joint distribution $\prod_e B(\bar{w}(e))$, which encodes the attractiveness of all joint items for the user at time t . In other words, if for some joint action e , $\mathbf{w}_t(e) = 1$, then the user, given that they *observe* item a , would be attracted by or click on item a at time t . Note that while \bar{w} stays constant throughout all rounds, \mathbf{w}_t is instantiated for round t only.

Remark 2 *The drawing of \mathbf{w}_t for each round t punishes any instance of repeated joint-items, as the probability of the user being attracted to each the repeated items is no longer independent; a user will assign 1 or 0 equally to all instances of that item for the duration of round t . Thus, in maximizing the probability a joint-ranking gets a click, it is in the players' best interests to always have K distinct joint-items.*

Avoiding this repetition is not easy, particularly in "collision-prone" multiplayer problems. Ultimately, we seek to contend with emergent complexities of a multi-agent setting by considering potential forms of *information asymmetry* present. The individual problems we study are expanded upon in the following sections.

Problem A *Asymmetry in Actions.* In this setting, each player is unable to observe the other players' actions, but they observe the same reward. In other words, the overall joint-ranking \mathbf{A}_t is unobservable to all players, but the item weights drawn by the user \mathbf{w}_t apply to all players.

Problem B *Asymmetry in Rewards.* In this setting, each player can observe the other player's actions, but they each obtain their own i.i.d. realization of the reward. They cannot observe other player's realizations.

Problem C *Asymmetry in both Actions and Rewards.* In this setting each player is unable to observe the other player's actions, and they obtain their own i.i.d realization of the reward.

We define the reward function f for some player i on any round t as follows:

$$f(\mathbf{A}_t, \mathbf{w}_t) = 1 - \prod_{k=1}^K (1 - \mathbf{w}_t(a_k)) \quad (1)$$

¹To avoid notation confusion, for a given item, a subscript indicates the *ranking position* of that *joint item* which is recommended, while a superscript indicates the *player* whom an item is from.

We evaluate a learning policy by *all* the players by its *expected regret*, defined as:

$$R_T = \mathbb{E} \left[\sum_{t=1}^T f(\mathbf{A}^*, \mathbf{w}_t) - f(\mathbf{A}_t, \mathbf{w}_t) \right] \quad (2)$$

where \mathbf{A}^* is an optimal joint-ranking consisting of the K items with the highest true attraction probabilities.

Note that from Remark 1 for Problem A, the players *all* receive a reward of 1 if at least one joint-item in \mathbf{A}_t is attractive to the user at time t , as each player observes the same ranking-position of the item clicked, C_t , because \mathbf{w}_t is shared. Also note that because of the reward asymmetry in Problems B and C, each player receives their own reward, but since the rewards are i.i.d., they will experience the same regret as regret is calculated in *expectation*.

3 Main Results and Algorithms

3.1 Problem A: Information Asymmetry in Actions

Recall in Problem A, players cannot observe the exact joint action taken. Incoordination arises easily, as joint-items played can differ from those intended. Consider for a given position, player 1 wants to play joint item (a, b) and player 2 wants to play (c, d) . The resulting joint item is (a, d) . Not only could this lead to repetition of joint-items, decreasing the likelihood of the ranking getting a click, but if this joint-item is observed, both players would record the observation and feedback for the wrong joint-item. Thus, players must somehow coordinate effectively and *infer* the actions of all other players. To solve this challenge of information asymmetry in actions, we introduce the mCascadeUCB algorithm. The psuedocode is in Algorithm 1. mCascadeUCB is a multiplayer adaptation of CascadeUCB1 (Kveton et al., 2015).

Players first estimate the user's attraction probabilities for each joint-item on each round by calculating its upper-confidence bound (UCB). As in CascadeUCB1, for each item e , the UCB at time t is given by:

$$\text{UCB}_t(e) = \begin{cases} \infty & \text{if } n_{t-1}(e) = 0 \\ \widehat{\mathbf{w}}_{n_{t-1}(e)}(e) + c_{n_{t-1}(e)} & \text{otherwise.} \end{cases} \quad (3)$$

where $\widehat{\mathbf{w}}_s(e)$ is the average of s observed weights of joint-item e , $n_t(e)$ is the number of times that item e is observed after t rounds, and:

$$c_s = \sqrt{\frac{1.5 \log T}{s}} \quad (4)$$

is the radius of a confidence interval around $\widehat{\mathbf{w}}_s(e)$ after t steps such that $\bar{w}(e) \in [\widehat{\mathbf{w}}_s(e) - c_s, \widehat{\mathbf{w}}_s(e) + c_s]$ holds with high probability.

Each player then plays their own K items that correspond to the K joint-items with the highest UCBs.² Here arises two concerns: 1) what should a player do when two or more joint-items' UCBs are equal, and 2) whether UCB indices for items differ between players, leading to players intending for different joint-actions. For 1), ties in the single-player UCB algorithm were settled arbitrarily, however, this fails in the multiplayer setting. So to address this issue, we define an *order relation* on \mathbf{E} , of which players can agree upon prior to learning and follow throughout the game. We define the joint items order relation as follows:

Definition 3 Let M be the number of players and let $\mathbf{a} = (a^1, \dots, a^M)$, $\mathbf{b} = (b^1, \dots, b^M)$ be two joint items in \mathbf{E} . We say $\mathbf{a} < \mathbf{b}$ if and only if there exists an $n \in \{1, 2, \dots, M\}$ such that for all $i < n$, $a_i = b_i$ and $a_n < b_n$.

In mCascadeUCB-A, players then settle UCB ties between joint-items choosing the *lesser* joint-items according to this relation. This ensures all players choose the same K items, at least for the first round. This coordination will continue for all rounds, because for Problem A, C_t is the same for all players. Thus, all players will update UCBs over \mathbf{E} identically, resolving 2), and ensuring for any round, all players can infer what joint-ranking will be played.

We show that mCascadeUCB achieves a similar upper-bound on the expected T -step regret as the single-agent CascadeUCB1 algorithm. We first assume the joint-items in \mathbf{E} are sorted by decreasing attraction probabilities, so that $\mathbf{A}^* = (1, 2, \dots, K)$ is the optimal ranking, consisting of the K *optimal joint-items*. Thus, joint-items

²Note that the reward is the same for all permutations of a joint-ranking, but as shown in the psuedocode for mCascadeUCB-A, we arrange these K items from lowest to highest UCB to increase the likelihood of more items being observed. This often is seen as a shortcoming of the cascade model, as intuition would want the most-attractive item to the user ranked first. We discuss this ordering later in our analysis.

Algorithm 1: mCascadeUCB

```

1 Each player  $i$  has a ground set  $E^i$  consisting of  $L$  items. Denote  $e$  to be a joint item and  $\mathbf{E} = \Pi_M(E^i)$  to be the set
  of all joint items by  $M$  players. Players will agree to an ordering of  $\mathbf{E}$  held constant during the entire game.
2  $\forall e \in \mathbf{E}$ , initialize  $\hat{\mathbf{w}}_0(e) = 0$  and  $\mathbf{n}_0(e) = 0$ .
3 for  $t = 1, \dots, T$  do
4    $\forall e \in \mathbf{E}$ , compute  $\text{UCB}_t(e)$  (see 3).
5
6   // Joint ranking selection
7   Each player  $i$  considers the  $K$  joint items with the largest UCBs. If there exist ties, preference goes to the lesser
    joint item(s) indicated by the order relation specified in Definition 3. These  $K$  items are then sorted from
    lowest UCB to highest UCB. Let  $\mathbf{a}_1, \dots, \mathbf{a}_K$  be these  $K$  joint items after this selection and sorting.
8   Each player  $i$  then selects their respective individual  $K$  items from  $E_i$  in the order that makes up this order of
     $K$  joint actions.
9    $(\mathbf{a}_1^t, \dots, \mathbf{a}_K^t) \rightarrow A_t$ 
10  Each player observes same click  $C_t \in \{1, \dots, K, \infty\}$ .
11
12  // Update relevant statistics
13   $\forall e \in \mathbf{E}, \mathbf{n}_t(e) = \mathbf{n}_{t-1}(e)$ 
14  forall  $k \in \{1, 2, \dots, C_t\}$  do
15     $\mathbf{a}_k^t \rightarrow e$ 
16     $\mathbf{n}_t(e) + 1 \rightarrow \mathbf{n}_t(e)$ 
17     $\frac{\hat{\mathbf{w}}_{t-1}(e)\mathbf{n}_{t-1}(e) + \mathbb{1}\{k = C_t\}}{\mathbf{n}_t(e)} \rightarrow \hat{\mathbf{w}}_t(e)$ 
18  end
19 end
    
```

$K + 1, \dots, L^M$, are called *sub-optimal*. The *sub-optimality gap* between an optimal item e^* and a suboptimal item e is defined to be the following:

$$\Delta_{e,e^*} := \bar{w}(e^*) - \bar{w}(e)$$

We make these assumptions again in later proofs of Problems B and C.

Theorem 4 *If each player uses mCascadeUCB-A in the setting of Problem A, then the expected T -step regret of mCascadeUCB-A is bounded as:*

$$R_T \leq \sum_{e=K+1}^{L^M} \frac{12}{\Delta_{e,K}} \log T + \frac{\pi^2}{3} L^M.$$

This upper bound is a direct corollary from Theorem 2 as proved in Kveton et al. [2015a]. As using mCascadeUCB-A essentially removes all action information asymmetry between the players and the players' UCB indices are identical throughout the play, Problem A becomes analogous to a single player cascading bandit, with the main distinction being there are now L^M total joint-items and K optimal joint-items with the highest attraction probabilities comprising the optimal ranking.

3.2 Problem B: Information Asymmetry in Rewards

In Problem A, since all players observe the same reward outcome (i.e., the same click), they maintain identical updates for each joint item's empirical mean and confidence interval over time. In other words, each player has the same ordering of the joint items at every step. A fixed ordering relation of joint items, used only to break ties in UCB values, thus suffices to ensure that all players choose the same top K items.

In Problem B, however, each player sees their own i.i.d. reward realization for any recommended item and will consequently form different empirical means for the same joint action. As a result, even for the same joint action, players will no longer share a common set of UCB values. Applying the UCB algorithm from Problem A could give rise to miscoordination — where one player aims for one joint arm, but another player aims for another joint arm, thus resulting in taking a completely different joint arm (see Figure 1 in Chang et al. [2022] for an illustration in the standard MAB problem that using UCB indices alone results in linear regret for Problem B).

Algorithm 2: mCascadeUCB-Intervals

```

1  Players will agree to an ordering of  $E$  held constant during the entire game.
2  For each player  $i$ , initialize the desired set  $D = E$ , and  $\forall e \in E$ , initialize  $\hat{w}_0(e) = 0$  and  $n_0(e) = 0$ .
3  for  $t = 1, \dots, T$  do
4      for each player  $i \in \{1, \dots, M\}$  do
5          // Compute UCB and LCB
6           $UCB_t(e) = \hat{w}_{n_{t-1}(e)} + c_{t-1, n_{t-1}(e)}$ 
7           $LCB_t(e) = \hat{w}_{n_{t-1}(e)} - c_{t-1, n_{t-1}(e)}$ 
8          Consider the next arm  $a$  inside  $D$  in accordance to the order (see step 22)
9          // Update desired set
10         for each recommendation  $rec \in \text{recommendations}$  do
11             for each  $e \in D^i$  do
12                 if Player  $i$  observes  $K$  other joint arms  $e$  satisfying  $UCB_t(rec) < LCB_t(e)$  then
13                     | Player  $i$  pulls the arm not equal to  $a[i]$ 
14                 end
15                 else
16                     | Player  $i$  pulls the arm equal to  $a[i]$ 
17                 end
18             end
19         end
20         for  $k = 1, \dots, K$  do
21             Observe  $k$ th joint actions  $a_t^1, \dots, a_t^K$  if  $a_t \neq a^k$  then
22                 | remove  $a^k$  from  $D$ . For the next round consider the arm after  $a$  (not  $a_t$ ).
23             end
24         end
25         Each player observe their own click  $C_t \in \{1, \dots, K, \infty\}$ .
26         // Update relevant statistics
27          $\forall e \in E, n_t(e) = n_{t-1}(e)$ 
28         forall  $k \in \{1, 2, \dots, C_t\}$  do
29              $a_k^t \rightarrow e$ 
30              $n_t(e) + 1 \rightarrow n_t(e)$ 
31              $\frac{\hat{w}_{n_{t-1}(e)} n_{t-1}(e) + \mathbb{1}\{k = C_t\}}{n_t(e)} \rightarrow \hat{w}_t(e)$ 
32         end
33     end
34 end

```

Thus for Problem B, we propose an intervals-based algorithm, mCascadeUCB-Intervals, the pseudocode of which is in 2. Recall the familiar UCB index ensures the true mean of an item lies below it with high probability, but by also subtracting the error term, $c_{n_{t-1}}(e)$, from the empirical mean, $\hat{w}_{n_{t-1}(e)}(e)$, we obtain the *lower confidence bound* (LCB), which with high probability, lies below the true mean for a joint-item. Together, the *UCB interval* contains the true attraction probability for a joint-item with high probability.

UCB intervals are powerful, as if two intervals do not overlap (i.e. are disjoint), then with high probability, we can say the item corresponding to the lower interval has a worse expected click probability than the item corresponding to the higher interval. Also note that by the error term $c_{n_{t-1}}(e)$ calculation, these intervals will be monotonically shrinking as items get observed more and more. Thus, in the case of cascading bandits specifically, where players need to recommend the top K joint-items (out of L^M), players can instead work to *eliminate* suboptimal items that are *not* in the top K , adopting a quasi-"innocent until proven guilty" strategy.

Concretely, in mCascadeUCB-Intervals, each player keeps track of their own *desired set*, initialized to contain all joint-items. Having agreed upon an ordering of the items before play, the players then *cycle* through the items *within their desired sets* for each round, i.e. $A_1 = (e_1, \dots, e_K)$, $A_2 = (e_2, \dots, e_{K+1})$, \dots , $A_{LM} = (e_{LM}, e_1, \dots)$, \dots . This idea of disjoint confidence intervals is crucial, as at any time, if *any* player observes an item whose UCB interval is disjoint and below K other items, once that joint-item is meant to be played in any joint-ranking, they *sabotage*. Instead of following the agreed upon order, they play a different individual item, which leads to a different overall joint-item. As players are able to observe each others' *actions*, this signals (without explicit communication) for all

players to remove this item from their desired sets. Thus, using intervals is essential to avoid incoordination, as players can reliably identify and *agree* upon the best K actions, even when their individual reward observations differ.

We show that using `mCascadeUCB-Intervals` for Problem B achieves $O(\log T)$ gap-dependent regret.

Theorem 5 *If each player uses `mCascadeUCB-Intervals` in the setting of Problem B, then the expected T -step regret of `mCascadeUCB-Intervals` is bounded as:*

$$R_T \leq \sum_{e=K+1}^{L^M} \frac{12 + 48K + 48\sqrt{K}}{\Delta_{e,K}} \log T + \left(\frac{\pi^2}{3} M + 2 \right) L^M.$$

The full proof is in Section 4. We first decompose the regret using the event where all items true means lie within their intervals. We then bound the number of observations of suboptimal items, however, using UCB intervals means resulting inequalities include error terms for both the suboptimal item and an optimal item. This issue is alleviated by Lemma 8, which lower bounds the number of observations for any optimal item, limiting the radius of its UCB interval.

3.3 Problem C

In Problem B, players can observe each other's actions, so despite receiving i.i.d. rewards, players can maintain the same desired set by following a fixed joint ordering relation and eliminating disjoint intervals when at least one player observes interval disjointness. However, the algorithm `mCascadeUCB-B` fails in Problem C because players can not observe other players' actions or rewards, making it impossible to coordinate in the same way.

Since each player must rely solely on their own i.i.d. observations in Problem C, we propose the `mCascadeDSEE` algorithm, which follows a structured exploration and exploitation schedule. The players decide on an ordering and choose a monotonic function $K(\lambda)$. In the λ -th exploration phase (starting from $\lambda = 1$), each joint-item e is ranked first $K(\lambda)$ times, to ensure feedback for all items in the cascading setting. Afterwards, each player commits to the top K items (arranged by the ordering) with the highest empirical means until the next power of 2. The function $K(\lambda)$ is chosen to tend to infinity so that more samples are collected in later epochs as the exploitation phases grow exponentially in length.

Although players may initially converge to different M -tuple optimal arms due to i.i.d. rewards, the probability of mistakes decreases rapidly as λ increases. Thus, with high probability, each player eventually identifies and commits to the true top K joint-items, forming the M -tuple optimal recommendation, \mathbf{A}^* . Additionally, because new exploration phases occur at powers of 2, the overall regret becomes upper bounded by $O(K(T) \log(T))$. The rigorous justification for this procedure is provided in the proof of Theorem 6.

Algorithm 3: `mCascadeDSEE`

- 1 Players will agree to an ordering of \mathbf{E} held constant during the entire game.
 - 2 Pick a monotonic function $K(\lambda) : \mathbb{N} \rightarrow \mathbb{N}$ such that $\lim_{t \rightarrow \infty} K(\lambda) = \infty$. First, let $\lambda = 1$.
 - 3 **for each joint-item $e \in \mathbf{E}$ do**
 - 4 For $K(\lambda)$ rounds, each player recommends their individual items that make up the joint-action \mathbf{A}_t starting with e followed by the next $K - 1$ joint-items after, according to the ordering.
 - 5 **end**
 - 6 **for each player $i \in \{1, \dots, M\}$ do**
 - 7 Player i calculates the empirical attraction probability for each joint-item e .
 - 8 Player i selects the top K joint-items with the highest-attraction probabilities, and in the event of a tie, selects arbitrarily. She arranges them into A_t by the agreed upon order, and commits to her corresponding items up until the next power of 2.
 - 9 **end**
 - 10 When $t = 2^n$ for some $n \geq \lceil \log_2(K(1)L^M) \rceil + 1$, go back to step (3) and start a new exploration phase, incrementing λ by 1.
-

Theorem 6 *If the players follow `mCascadeDSEE` in Algorithm 3 in the setting of Problem C, then we have the following regret bound:*

$$R_T \leq O(K(T) \log(T)) \quad (5)$$

The full proof is in section 4.2. Notice the lack of dependence on the gap $\Delta_{e,K}$ in the above. This is because there is a constant $\sum_{t=1}^{\infty} t^{-2K_0(t)\epsilon^2}$ whose order depends on $\epsilon < \frac{1}{2} \min_{e,e^*} \Delta_{e,e^*} = \frac{1}{2} \Delta_{K+1,K}$.

4 Regret Analysis

4.1 Problem B

Here we prepare the proof of the upper bound in Theorem 5. We first cite some important auxiliary definitions and results from previous literature, particularly Kveton et al. [2015a], attempting to parallel notation when possible.

- For any joint ranking \mathbf{A}_t , define the *permutation of optimal joint-items*, π_t as such: for $k = 1, \dots, K$, if the k -th joint item in \mathbf{A}_t is optimal, set $\pi_t(k) = \mathbf{a}_k^t$. The remaining joint items in \mathbf{A}_t are positioned arbitrarily.
- For any optimal joint-item e^* and sub-optimal joint-item e , let $G_{e,e^*,t}$ be the event e is chosen instead of item e^* at time t , and that e is observed. That is:

$$G_{e,e^*,t} = \{\exists 1 \leq k \leq K \text{ s.t. } \mathbf{a}_k^t = e, \pi_t(k) = e^*, \text{ and } \mathbf{w}_t(\mathbf{a}_1^t), \dots, \mathbf{w}_t(\mathbf{a}_{k-1}^t) = 0\}$$

These definitions set up the use of Theorem 1 from Kveton et al. [2015a], which allows us to decompose the expected regret on round t by looking at the sub-optimality gaps and the G event for all suboptimal items.

Lemma 7 (Theorem 1, Kveton et al. [2015a], adapted for joint-items and joint-rankings)

$$\mathbb{E}_t [R(\mathbf{A}_t, \mathbf{w}_t)] \leq \sum_{e=K+1}^L \sum_{e^*=1}^K \Delta_{e,e^*} \mathbb{E}_t [\mathbb{1}\{G_{e,e^*,t}\}]$$

We now prove Theorem 5.

Proof: (Theorem 5) Define $\mathcal{E}_t^i = \{\exists e \in \mathbf{E} \text{ s.t. } |\bar{w}(e) - \hat{\mathbf{w}}_{n_{t-1}^i(e)}(e)| \geq c_{n_{t-1}^i(e)}\}$. Thus, $\bigcup_i \mathcal{E}_t^i$ is the event that for some player i , there exists some joint-item e where the $\bar{w}(e)$ is *not* within the player i 's UCB interval around $\hat{\mathbf{w}}_{n_{t-1}^i(e)}(e)$. Let $\bigcap_i \bar{\mathcal{E}}_t^i$ be the complement; that is, for all players and for all joint items e , $\bar{w}(e)$ lies within each player's respective UCB interval for e . Decompose the regret of mCascadeUCB-Intervals as

$$R_T = \mathbb{E} \left[\sum_{t=1}^T \mathbb{I}\left\{\bigcup_i \mathcal{E}_t^i\right\} R(\mathbf{A}_t, \mathbf{w}_t) \right] + \mathbb{E} \left[\sum_{t=1}^T \mathbb{I}\left\{\bigcap_i \bar{\mathcal{E}}_t^i\right\} R(\mathbf{A}_t, \mathbf{w}_t) \right] \quad (6)$$

where $R(\mathbf{A}_t, \mathbf{w}_t)$ is the regret incurred on time t .

For the first term of 6, note:

$$\mathbb{E} \left[\sum_{t=1}^T \mathbb{I}\left\{\bigcup_i \mathcal{E}_t^i\right\} R(\mathbf{A}_t, \mathbf{w}_t) \right] \leq \mathbb{E} \left[\sum_{t=1}^T \mathbb{I}\left\{\bigcup_i \mathcal{E}_t^i\right\} \right] \leq \sum_{i=1}^M \mathbb{E} \left[\sum_{t=1}^T \mathbb{I}\{\mathcal{E}_t^i\} \right] \leq \frac{\pi^2}{3} M L^M$$

where the second inequality follows from the fact that 1) Theorem 2 from Kveton et al. [2015a] which uses the fact that our UCB intervals across all L^M joint-items were constructed to hold with high probability (Hoeffding's Inequality) and 2) the aforementioned upper bound holds for each player.

Next, we bound the number of observations of a suboptimal joint-item e under the good event $\bigcap_i \bar{\mathcal{E}}_t^i$ by any player. Since the players' clicks/observations are i.i.d. and regret is calculated in expectation, consider the regret incurred by an arbitrary player i and let $n_{t-1}^i(e)$ denote the number of observations for item e for player i up to time t . We first decompose the second term of 6 into the first $2L^M$ rounds, where $R(\mathbf{A}_t, \mathbf{w}_t) \leq 1$, and the remaining $T - 2L^M$ rounds. Invoking Lemma 7 for the latter, the second term of 6 is thus upper bounded as:

$$\mathbb{E} \left[\sum_{t=1}^T \mathbb{I}\left\{\bigcap_i \bar{\mathcal{E}}_t^i\right\} R(\mathbf{A}_t, \mathbf{w}_t) \right] \leq \sum_{e=K+1}^{L^M} \mathbb{E} \left[\sum_{e^*=1}^K \sum_{t=2L^M+1}^T \Delta_{e,e^*} \mathbb{I}\left\{\bigcap_i \bar{\mathcal{E}}_t^i, G_{e,e^*,t}\right\} \right] + 2L^M \quad (7)$$

where $G_{e,e^*,t}$ is now the event that e is chosen instead of e^* at time t , and that e is observed for player i . Select any optimal joint-item e^* . For any time $t > 2L^M$, under $G_{e,e^*,t}$, this implies e and e^* 's UCB intervals are *not* disjoint for all players (otherwise e would've been eliminated), which means $\hat{\mathbf{w}}_{n_{t-1}^i(e)}(e) + c_{n_{t-1}^i(e)} \geq \hat{\mathbf{w}}_{n_{t-1}^i(e^*)}(e^*) - c_{n_{t-1}^i(e^*)}$.

Additionally, under $\bigcap_i \bar{\mathcal{E}}_t$, we know the true attractions probabilities lie within each players' respective UCB intervals, i.e. $|\bar{w}(e) - \hat{w}_{n_{t-1}^i(e)}(e)| < c_{n_{t-1}^i(e)}$ and $|\bar{w}(e^*) - \hat{w}_{n_{t-1}^i(e^*)}(e^*)| < c_{n_{t-1}^i(e^*)}$. With these facts together, it holds that:

$$\bar{w}(e) + 2c_{n_{t-1}^i(e)} \geq \bar{w}(e^*) - 2c_{n_{t-1}^i(e^*)}$$

which implies:

$$\begin{aligned} \Delta_{e,e^*} &\leq 2(c_{n_{t-1}^i(e)} + c_{n_{t-1}^i(e^*)}) \\ &\leq 2 \left(\sqrt{\frac{1.5 \log T}{n_{t-1}^i(e)}} + \sqrt{\frac{1.5 \log T}{n_{t-1}^i(e^*)}} \right) \end{aligned} \quad (\text{By definition of } c)$$

By Lemma 8, for any $t > 2L^M$, we have $n_{t-1}^i(e^*) \geq \frac{n_{t-1}^i(e)}{4K}$. Thus,

$$\begin{aligned} \Delta_{e,e^*} &\leq 2 \left(\sqrt{\frac{1.5 \log T}{n_{t-1}^i(e)}} + \sqrt{\frac{6K \log T}{n_{t-1}^i(e)}} \right) \quad (\text{by Lemma 8}) \\ \implies n_{t-1}^i(e) &\leq \frac{(6 + 24K + 24\sqrt{K}) \log T}{\Delta_{e,e^*}^2} \end{aligned}$$

Let $\tau_{e,e^*} = \frac{(6+24K+24\sqrt{K}) \log T}{\Delta_{e,e^*}^2}$. Therefore:

$$\sum_{e^*=1}^K \sum_{t=2L^M+1}^T \Delta_{e,e^*} \mathbb{I}\left\{\bigcap_i \bar{\mathcal{E}}_t, G_{e,e^*,t}\right\} \leq \sum_{e^*=1}^K \Delta_{e,e^*} \sum_{t=2L^M+1}^T \mathbb{I}\left\{\bigcap_i \{n_{t-1}^i(e) \leq \tau_{e,e^*}\}, G_{e,e^*,t}\right\}. \quad (8)$$

Let:

$$\mathbf{M}_{e,e^*} = \sum_{t=2L^M+1}^T \mathbb{I}\left\{\bigcap_i \{n_{t-1}^i(e) \leq \tau_{e,e^*}\}, G_{e,e^*,t}\right\}$$

be the inner sum in 8. Now note that 1) across all players i , the number of observations $n_{t-1}^i(e)$ of item e increases by one when the event $G_{e,e^*,t}$ happens for that player for any optimal item e^* , 2) for any given player, the event $G_{e,e^*,t}$ happens for at most one optimal e^* at any time t , and 3) $\tau_{e,1} \leq \dots \leq \tau_{e,K}$.

From these facts, we can bound the number of times the indicator event is true over $t = 2L^M, \dots, T$ as $\mathbf{M}_{e,e^*} \leq \tau_{e,e^*}$, and moreover, $\sum_{e^*=1}^K \mathbf{M}_{e,e^*} \leq \tau_{e,K}$. Thus, the right-hand side of 8 is bounded above by:

$$\max \left\{ \sum_{e^*=1}^K \Delta_{e,e^*} m_{e,e^*} : 0 \leq m_{e,e^*} \leq \tau_{e,e^*}, \sum_{e^*=1}^K m_{e,e^*} \leq \tau_{e,K} \right\}.$$

Since the gaps are decreasing, $\Delta_{e,1} \geq \dots \geq \Delta_{e,K}$, we maximize the quantity above by making m_{e,e^*} the largest possible for lower-numbered optimal joint-items, i.e. $m_{e,1}^* = \tau_{e,1}, m_{e,2}^* = \tau_{e,2} - \tau_{e,1}, \dots, m_{e,K}^* = \tau_{e,K} - \tau_{e,K-1}$ (where the first constraint is satisfied by fact 3). Substituting in for τ_{e,e^*} , we see 8 is bounded above by:

$$\left[\Delta_{e,1} \frac{1}{\Delta_{e,1}^2} + \sum_{e^*=2}^K \Delta_{e,e^*} \left(\frac{1}{\Delta_{e,e^*}^2} - \frac{1}{\Delta_{e,e^*-1}^2} \right) \right] (6 + 24K + 24\sqrt{K}) \log T. \quad (9)$$

By Lemma 3 of Kveton et al. [2014], the term inside the square brackets of 9 is bounded above by $\frac{2}{\Delta_{e,K}}$. Thus, summing over all suboptimal joint-items e , we get that the first term of the right-hand side of 7 is bounded above as:

$$\sum_{e=K+1}^{L^M} \frac{12 + 48K + 48\sqrt{K}}{\Delta_{e,K}} \log T$$

Chaining all inequalities finishes the proof. \square

4.2 Problem C

Proof: (Theorem 6) Decompose the total regret $R_T = R_{T,E} + R_{T,C}$, where $R_{T,E}$ is the regret incurred during the exploration sequence spaced at powers of 2, while $R_{T,C}$ is the regret incurred during the commitment phase when committing to the ranking of the top K items with the highest estimated means.

During the λ -th exploration phase, each joint-item e is ranked first $K(\lambda)$ times. Let $\nu_t(e)$ denote the total number of times joint-item e is ranked first during exploration up to time t . The number of exploration phases is approximately $\log_2(T)$, since exploration occurs at intervals based on powers of 2. It follows that $\nu_t(e) \leq K(\lfloor \log_2(t) \rfloor) \lceil \log_2(t) \rceil$. [kenny: is this right? sorry for changing to whoever did this first]

As $R_t(\mathbf{A}_t, \mathbf{w}_t) \leq 1$, regret incurred during exploration is thus bounded by the total number of exploration rounds, i.e.:

$$R_{T,E} \leq \sum_{e \in \mathbf{E}} \nu_t(e) \leq L^M \cdot K(\lfloor \log_2(T) \rfloor) \cdot \lceil \log_2(T) \rceil.$$

Next, we show $R_{T,C}$ is bounded by a constant less than ∞ . Let $\epsilon < \frac{1}{2} \Delta_{K+1,K}$, where K is the optimal item with the lowest true attraction probability and $K+1$ is the suboptimal item with the highest true attraction probability (thus, $\Delta_{K+1,K}$ is the minimum sub-optimality gap). Consider the good event $G_t^i(e)$ where for player i at time t , the empirical probability for item e ($\hat{\mathbf{w}}_{n_{t-1}^i(e)}(e)$) is within ϵ of the true probability for item e ($\bar{w}(e)$), i.e.:

$$G_t^i(e) = \{|\hat{\mathbf{w}}_{n_{t-1}^i(e)}(e) - \bar{w}(e)| < \epsilon\}$$

It follows that when $G_t = \bigcap_{e,i} G_t^i(e)$ all players collectively choose $\mathbf{A}^* = (\mathbf{1}, \dots, \mathbf{K})$ and no regret is incurred. Using the law of total expectation, our commit-phase regret can thus be bounded as:

$$\begin{aligned} R_{T,C} &= \sum_{t \in C} \mathbb{E}[R_t \mid G_t] P(G_t) + \mathbb{E}[R_t \mid G_t^c] P(G_t^c) \\ &\leq \sum_{t=1}^T P(G_t^c) \quad (\text{since } \mathbb{E}[R_t \mid G_t] \leq 1 \text{ and } \mathbb{E}[R_t \mid G_t] = 0) \\ &\leq \sum_{t=1}^T \sum_{i=1}^M \sum_{e \in \mathbf{E}} P(G_t^i(e)^c) \quad (\text{applying De Morgan's Law and union bound}) \\ &= M \sum_{t=1}^T \sum_{e \in \mathbf{E}} P(|\hat{\mathbf{w}}_{n_{t-1}(e)}(e) - \bar{w}(e)| \geq \epsilon) \quad (\text{by i.i.d. rewards}) \\ &\leq 2M \sum_{t=1}^T \sum_{e \in \mathbf{E}} \exp(-2n_{t-1}(e)\epsilon^2) \quad (\text{by Hoeffding's inequality}) \\ &\leq 2M \sum_{t=1}^T \sum_{e \in \mathbf{E}} \exp(-2K_0(t) \log_2(t) \epsilon^2) \quad (\text{using } n_{t-1}(e) \geq K_0(t) \log_2(t), \text{ Claim 2, Theorem 5 in Chang et al. [2022]}) \\ &\leq 2ML^M \sum_{t=1}^T t^{-2K_0(t)\epsilon^2} \\ &\leq 2ML^M \sum_{t=1}^{\infty} t^{-2K_0(t)\epsilon^2} \end{aligned}$$

The function $f(t) = t^{-2K_0(t)\epsilon^2}$ is monotonically decreasing for $t \geq 1$ since $K_0(t)$ is a non-decreasing function that tends to infinity as $t \rightarrow \infty$. Thus, the sum $\sum_{t=1}^{\infty} t^{-2K_0(t)\epsilon^2}$ can be bounded by:

$$\sum_{t=1}^{\infty} t^{-2K_0(t)\epsilon^2} \leq 1 + \int_1^{\infty} t^{-2K_0(t)\epsilon^2} dt$$

Since $K_0(t)$ tends to infinity, there exists an integer N such that for all $t > N$, $2K_0(t)\epsilon^2 \geq 3$ so that $t^{-2K_0(t)\epsilon^2} \leq t^{-3}$. As $\gamma > 1$ ensures that the integral $\int_1^{\infty} t^{-\gamma} dt$ converges, it follows that $\int_N^{\infty} t^{-2K_0(t)\epsilon^2} dt < \infty$. Therefore, $R_{T,C}$ is

bounded by a constant, leading to the total regret bound:

$$R_T \leq O(K(T) \log(T)).$$

□

5 Experiments

References

- E. Agichtein, E. Brill, and S. Dumais. Improving web search ranking by incorporating user behavior information. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 19–26, 2006.
- B. Awerbuch and R. Kleinberg. Competitive collaborative learning. *Journal of Computer and System Sciences*, 74(8): 1271–1288, 2008.
- H. Becker, C. Meek, and D. M. Chickering. Modeling contextual factors of click rates. In *AAAI*, volume 7, pages 1310–1315, 2007.
- R. Bonnefoi, L. Besson, C. Moy, E. Kaufmann, and J. Palicot. Multi-armed bandit learning in iot networks: Learning helps even in non-stationary settings. In *International Conference on Cognitive Radio Oriented Wireless Networks*, pages 173–185. Springer, 2017.
- E. Boursier and V. Perchet. A survey on multi-player bandits. *arXiv preprint arXiv:2211.16275*, 2022.
- J. Cao, W. Sun, and Z.-J. M. Shen. Doubly adaptive cascading bandits with user abandonment. *Available at SSRN 3355211*, 2019.
- N. Cesa-Bianchi, C. Gentile, Y. Mansour, and A. Minora. Delay and cooperation in nonstochastic bandits. In *Conference on Learning Theory*, pages 605–622. PMLR, 2016.
- N. Cesa-Bianchi, T. Cesari, and C. Monteleoni. Cooperative online learning: Keeping your neighbors updated. In *Algorithmic learning theory*, pages 234–250. PMLR, 2020.
- W. Chang and Y. Lu. Optimal cooperative multiplayer learning bandits with noisy rewards and no communication. *arXiv preprint arXiv:2311.06210*, 2023.
- W. Chang and Y. Lu. Linucb in multiplayer information asymmetric contextual bandits. *Submitted to Transactions on Machine Learning Research*, 2024. URL <https://openreview.net/forum?id=nMCJ8bFq4B>. Under review.
- W. Chang, M. Jafarnia-Jahromi, and R. 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.
- W. C. Cheung, V. Tan, and Z. Zhong. A thompson sampling algorithm for cascading bandits. In K. Chaudhuri and M. Sugiyama, editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 438–447. PMLR, 16–18 Apr 2019. URL <https://proceedings.mlr.press/v89/cheung19a.html>.
- H.-j. Choi, R. Udawani, and M.-h. Oh. Cascading contextual assortment bandits. *Advances in Neural Information Processing Systems*, 36, 2024.
- R. Combes, S. Magureanu, A. Proutiere, and C. Laroche. Learning to rank: Regret lower bounds and efficient algorithms. In *Proceedings of the 2015 ACM SIGMETRICS international conference on measurement and modeling of computer systems*, pages 231–244, 2015.
- N. Craswell, O. Zoeter, M. Taylor, and B. Ramsey. An experimental comparison of click position-bias models. In *Proceedings of the 2008 international conference on web search and data mining*, pages 87–94, 2008.
- Y. Du, R. Srikant, and W. Chen. Cascading reinforcement learning. *arXiv preprint arXiv:2401.08961*, 2024.
- A. Dubey et al. Cooperative multi-agent bandits with heavy tails. In *International conference on machine learning*, pages 2730–2739. PMLR, 2020.
- T. Jin, H.-L. Hsu, W. Chang, and P. Xu. Finite-time frequentist regret bounds of multi-agent thompson sampling on sparse hypergraphs, 2023. URL <https://arxiv.org/abs/2312.15549>.
- H. Kao. *Efficient Methods for Optimizing Decentralized Multi-Agent Systems*. PhD thesis, 2022.
- H. Kao, C.-Y. Wei, and V. Subramanian. Decentralized cooperative reinforcement learning with hierarchical information structure. In *International Conference on Algorithmic Learning Theory*, pages 573–605. PMLR, 2022.
- S. Katariya, B. Kveton, C. Szepesvari, and Z. Wen. Dcm bandits: Learning to rank with multiple clicks. In *International Conference on Machine Learning*, pages 1215–1224. PMLR, 2016.
- B. Kveton, Z. Wen, A. Ashkan, H. Eydgahi, and B. Eriksson. Matroid bandits: Fast combinatorial optimization with learning. *arXiv preprint arXiv:1403.5045*, 2014.
- B. Kveton, C. Szepesvari, Z. Wen, and A. Ashkan. Cascading bandits: Learning to rank in the cascade model. In *International conference on machine learning*, pages 767–776. PMLR, 2015a.

- B. Kveton, Z. Wen, A. Ashkan, and C. Szepesvari. Combinatorial cascading bandits. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015b. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/1f50893f80d6830d62765ffad7721742-Paper.pdf.
- P. Landgren, V. Srivastava, and N. E. Leonard. On distributed cooperative decision-making in multiarmed bandits. In *2016 European Control Conference (ECC)*, pages 243–248. IEEE, 2016.
- C. Li and M. De Rijke. Cascading non-stationary bandits: Online learning to rank in the non-stationary cascade model. *arXiv preprint arXiv:1905.12370*, 2019.
- C. Li, H. Feng, and M. d. Rijke. Cascading hybrid bandits: Online learning to rank for relevance and diversity. In *Proceedings of the 14th ACM Conference on Recommender Systems*, pages 33–42, 2020.
- S. Li and S. Zhang. Online clustering of contextual cascading bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.
- S. Li, B. Wang, S. Zhang, and W. Chen. Contextual combinatorial cascading bandits. In M. F. Balcan and K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1245–1253, New York, New York, USA, 20–22 Jun 2016. PMLR. URL <https://proceedings.mlr.press/v48/lif16.html>.
- Z. Mammeri. Reinforcement learning based routing in networks: Review and classification of approaches. *Ieee Access*, 7:55916–55950, 2019.
- M. Mansoury, B. Mobasher, and H. van Hoof. Mitigating exposure bias in online learning to rank recommendation: A novel reward model for cascading bandits. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 1638–1648, 2024.
- W. Mao, T. Basar, L. Yang, and K. Zhang. Decentralized cooperative multi-agent reinforcement learning with exploration. 2021.
- W. Mao, L. Yang, K. Zhang, and T. Basar. On improving model-free algorithms for decentralized multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 15007–15049. PMLR, 2022.
- D. Martínez-Rubio, V. Kanade, and P. Rebeschini. Decentralized cooperative stochastic bandits. *Advances in Neural Information Processing Systems*, 32, 2019.
- F. Radlinski and T. Joachims. Query chains: learning to rank from implicit feedback. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 239–248, 2005.
- S. Shahrampour, A. Rakhlin, and A. Jadbabaie. Multi-armed bandits in multi-agent networks. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2786–2790. IEEE, 2017.
- B. Szorenyi, R. Busa-Fekete, I. Hegedus, R. Ormándi, M. Jelasity, and B. Kégl. Gossip-based distributed stochastic bandit algorithms. In *International conference on machine learning*, pages 19–27. PMLR, 2013.
- D. Vial, S. Sanghavi, S. Shakkottai, and R. Srikant. Minimax regret for cascading bandits. *Advances in Neural Information Processing Systems*, 35:29126–29138, 2022.
- D. Wang, J. Cao, Y. Zhang, and W. Qi. Cascading bandits: optimizing recommendation frequency in delayed feedback environments. *Advances in Neural Information Processing Systems*, 36, 2024.
- K. Wang. Conservative contextual combinatorial cascading bandit. *IEEE Access*, 9:151434–151443, 2021.
- P.-A. Wang, A. Proutiere, K. Ariu, Y. Jedra, and A. Russo. Optimal algorithms for multiplayer multi-armed bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 4120–4129. PMLR, 2020.
- M. Xu and D. Klabjan. Decentralized randomly distributed multi-agent multi-armed bandit with heterogeneous rewards. *Advances in Neural Information Processing Systems*, 36, 2024.
- S. Yamin and H. H. Permuter. Multi-agent reinforcement learning for network routing in integrated access backhaul networks. *Ad Hoc Networks*, 153:103347, 2024.
- H. Yang, X. Liu, Z. Wang, H. Xie, J. C. Lui, D. Lian, and E. Chen. Federated contextual cascading bandits with asynchronous communication and heterogeneous users. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 20596–20603, 2024.
- S. Zeng, X. Xu, and Y. Chen. Multi-agent reinforcement learning for adaptive routing: A hybrid method using eligibility traces. In *2020 IEEE 16th International Conference on Control & Automation (ICCA)*, pages 1332–1339. IEEE, 2020.
- Z. Zhong, W. C. Cheung, and V. Tan. Best arm identification for cascading bandits in the fixed confidence setting. In *International Conference on Machine Learning*, pages 11481–11491. PMLR, 2020.

- Z. Zhong, W. C. Chueng, and V. Y. Tan. Thompson sampling algorithms for cascading bandits. *Journal of Machine Learning Research*, 22(218):1–66, 2021.
- S. Zong, H. Ni, K. Sung, N. R. Ke, Z. Wen, and B. Kveton. Cascading bandits for large-scale recommendation problems. *arXiv preprint arXiv:1603.05359*, 2016.

6 Appendix

Lemma 8 *Suppose the players are following mCascadeUCB-Intervals. Let e be any suboptimal item and e^* be any optimal item in a ground set \mathbf{E} with cardinality L . Then for any $t > 2L^M$, if $\bigcap_i \bar{\mathcal{E}}_t$ happens, the number of observations by player i of optimal item e^* is lower bounded as:*

$$n_{t-1}^i(e^*) \geq \frac{1}{4K} n_{t-1}^i(e)$$

Proof: As we are in $\bigcap_i \bar{\mathcal{E}}_t$, we can consider the case where both e^* and e are both still within the desired set at time t ; the case when e has been eliminated follows as a result, because e no longer will be observed and thus the lower bound is unchanging. Without loss of generality, assume the size of the desired set at time t is L^M . Partition the t rounds into groups of size L^M , namely:

$$\{1, \dots, t\} = \{1, \dots, L^M\} \cup \{L^M + 1, \dots, 2L^M\} \cup \dots \cup \{\lfloor \frac{t}{L^M} \rfloor L^M + 1, \dots, t\}.$$

In the event the desired set is smaller than L^M , simply partition the t rounds by each loop over the current desired set. Since by mCascadeUCB-Intervals, the arms are recommended cyclically, in each group of rounds (e.g. $\{nL^M + 1, \dots, (n+1)L^M\}$), for exactly one round, e^* must be the first joint-item to be recommended, and therefore will be observed by all the users for that round. Therefore, $n_{t-1}(e) \geq 1$ (which is true for $t > L^M$),

$$n_{t-1}^i(e^*) \geq \lfloor \frac{t}{L^M} \rfloor \geq \frac{t}{2L^M}$$

where the last inequality holds as $\frac{t}{L^M} \geq 2$. Furthermore, by mCascadeUCB-Intervals, e is recommended (and therefore observed) at most K times over every round. Therefore, we have whenever $n_{t-1}(e) \geq 1$ (again true for $t > L^M$),

$$n_{t-1}^i(e) \leq K \lceil \frac{t}{L^M} \rceil \leq 2K \frac{t}{L^M}$$

where the last inequality holds as $\frac{t}{L^M} \geq 1$. Combining both inequalities above, we have,

$$n_{t-1}^i(e^*) \geq \frac{1}{4K} n_{t-1}^i(e).$$

□