Combinatorial bandits for maximum value reward function under maximum value and index feedback

Anonymous Authors[1]

Abstract

We consider a combinatorial multi-armed bandit problem for maximum value reward function under maximum value and index feedback. Our problem has a new feedback structure that is in between semi-bandit and full-bandit feedback. We propose an algorithm and provide a regret bound for problem instances with stochastic outcomes of arms according to arbitrary distributions with finite supports. The key idea is in using a reduction to the case of binary distributions. The regret analysis rests on considering an extended set of arms, associated with values and probabilities of outcomes, and applying a smoothness condition. Our algorithm achieves a $O((k/\Delta)\log(T))$ distribution-dependent and a $\tilde{O}(\sqrt{T})$ distribution-independent regret where k is the number of arms selected in each round, Δ is a distribution-dependent gap and T is the horizon time. We demonstrate the effectiveness of our algorithm empirically in several simulation settings.

1. Introduction

We consider a sequential decision making problem in which an agent selects a set of items of cardinality k from a set of n items in each round, where each selected item independently produces a random value, and the reward of the set is the maximum of these random values. After selecting a set of items in a round, the agent observes the feedback that consists of the maximum value and the identity of the item that achieves this maximum value. We refer to this as the *max value-index feedback*. The values of items are assumed to be stochastic and independent over items and rounds. We first consider binary distributions; and then further extend this to discrete distributions with finite supports. The performance is measured by expected cumulative regret over a time horizon, defined as the difference of the cumulative rewards achieved by selecting a set with maximum expected reward in each round and that achieved by the learning agent. We refer to this sequential decision making problem as k-MAX *bandit with max value-index feedback*.

There are several motivating applications for the k-MAX bandit problem with the max value-index feedback. In crowdsourcing platforms and team formation applications (Kleinberg and Raghu, 2018; Sekar et al., 2021; Lee et al., 2022; Mehta et al., 2020), a team performance may correspond to the best individual performance, which is often dubbed as the strongest-link performance. In this case, observing a team success measure corresponds to observing best individual value. The observation data may also contain information about which individual accomplished the best solution. In project portfolio selection, e.g. R&D projects in pharmaceutical industry (Blau et al., 2004; Jekunen, 2014), projects may fail or succeed and different projects may have different rewards conditional on being successful, including high-risk, high-reward projects. In recommendation systems and information retrieval (Manning et al., 2008; Chapelle and Zhang, 2009), a goal is to recommend a set of items to a user that maximizes the probability of the recommendation set containing at least one relevant item. Not all relevant items need to be equally relevant to the user, as some may be of higher value than other. In selection problems, such as in job search, school admissions, server selection in data centers and various other settings, an agent (e.g. job applicant) may have preferences over a set of items (e.g. employers) and upon selecting a set of items only some may be available (e.g. depending on being offered a job).

Our goal is to maximize the expected cumulative reward for the learning agent over the time horizon. The problem is challenging mainly for two reasons. Firstly, the reward function is the maximum value function, which is nonlinear and thus depends not only on the expected values of the constituent base arms. The uncertainty of binary-valued items makes the problem more challenging under the maximum value reward. As we will show in the numerical section, high-risk high-reward items may outperform stable-value items in this case. The second challenge is due to the limited feedback. The agent only observes the maximum value and the identity of an item that achieves this value. These all make it hard to estimate distributions of values of base arms.

The k-MAX bandit problem has action set and rewards

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

like in some combinatorial bandits, but its max value-index feedback structure is neither semi-bandit nor full-bandit feedback, commonly studied in bandits literature. Assume outcomes of arms are according to distributions of some independent random variables $X_1, \ldots, X_n$. Then, for any set S of arms, semi-bandit feedback consists of all values $\{X_i \mid i \in S\}$, while full-bandit feedback is only maximum value, $\max\{X_i \mid i \in S\}$, under maximum value reward function. On the other hand, the max value-index feedback consists of maximum value $\max\{X_i \mid i \in S\}$ and index $I \in \arg\max\{X_i \mid i \in S\}$. This feedback is between semi-bandit feedback and full-bandit feedback, and only slightly stronger than the full-bandit feedback. Indeed, the only information that can be deduced from the max value-index feedback about arms $j \neq I$, is that their outcome values are smaller than or equal to the maximum value X_I. The index feedback is alike to comparison observations in dueling bandits, but with additional maximum-value feedback.

We present algorithms for the underlying sequential decision making problem and show regret upper bounds for these algorithms, under different assumptions on what information is available about item values. Our results show that despite limited feedback, comparable regret bounds to combinatorial semi-bandits can be achieved for the k-MAX bandit problem with the max value-index feedback.

1.1. Related work

The problem we study has connections with combinatorial multi-armed bandits (CMAB) (Cesa-Bianchi and Lugosi, 2012; Chen et al., 2013; 2016b). Most of the existing work on CMAB problems is focused on semi-bandit feedback setting, e.g. (Chen et al., 2013; Kveton et al., 2015a). The k-MAX problem with the semi-bandit feedback was studied in (Chen et al., 2016a), and the solution is easier than in our paper because the semi-bandit feedback provides much more information.

In most works on full-bandit CMAB, restrictions are placed on the reward function. (Auer et al., 2002) studied the problem under linear reward and provided a linear UCB algorithm. (Dani et al., 2008) fully analyzed the linear UCB algorithm and gave a nearly optimal regret bound. (Rejwan and Mansour, 2020) considered the reward defined as the sum of arm values. Only a few algorithms have been proposed for full-bandit CMAB problem with non-linear rewards. (Katariya et al., 2017) considered minimum value reward function and item values according to Bernoulli distributions. (Gopalan et al., 2014) studied the full-bandit CMAB with general rewards using Thompson sampling method. However, it is computationally hard to compute the posteriors in the algorithm and the regret bound has a large exponential constant. Recent work by (Agarwal et al., 2021) proposed a merge and sort algorithm under assumption that distributions of base arm outcomes obey a stochastic dominance relation. This is a restrictive assumption, which does not generally hold, such as for binary distributions. We thus point out that full-bandit CMAB solutions proposed so far either do not apply to our problem or require exponential computational complexity in the regret bound.

A related work is on combinatorial cascading bandits, e.g. (Kveton et al., 2015b). An agent chooses an ordered subsequence from the set of base arms and the outcomes of base arms are revealed one by one until a stopping criteria is met. (Chen et al., 2016b) generalized the problem to the framework of combinatorial semi-bandits with probabilistically triggered arms (CMAB-T). The main difference with our setting is that CMAB-T assumes more information and is inherently semi-bandit. By revealing the outcome of base arms one by one, the agent is able to observe individual rewards for all arms selected before the one meeting the criteria. Another difference in our work is that we consider more general item value distributions.

We summarize some known results on regret bounds for CMAB problems in Table 1. In the table, Δ denotes the gap between the optimum expected regret of a set and the best suboptimal expected regret of a set. (Agarwal et al., 2021) only provides a distribution-independent regret bound, which is worse than $\tilde{O}(T^{1/2})$ distribution-independent regret bounds that follow from our distribution-dependent regret bounds. The goal in (Sui et al., 2017) is to select the best item instead of the best set of items, and for this reason there is no dependency on k in their bound.

Another related line of work is that on dueling bandits (Ailon et al., 2014) where the agent plays two arms at each time and observes the outcome of the duel. The goal is to find the best arm in the sense of a Condorcet winner under relative feedback of the dueling outcomes. (Sui et al., 2017) extended the setting to multiple dueling bandits problem by simultaneously playing k arms instead of two arms. Compared with this line of work, we assume additional absolute feedback. Our goal is different as we would like to select a best set of items with respect to a non-linear reward function.

It is noteworthy that our work is related to choice models, e.g. (Luce, 1959) (Thurstone, 1927), and sequential learning for choice models (Agarwal et al., 2020). The main difference from previous work is that we consider maximum value and index feedback.

1.2. Summary of contributions

Our results can be summarized in the following points.

- We consider the combinatorial bandit problem for maximum value reward function under max value-index feedback. This is a new problem with feedback in between full-bandit and semi-bandit feedback, and only slightly stronger than the full-bandit feedback. Com-

Table 1. Known regret bounds for CMAB problems under different settings.

	Feedback	Restrictions	Regret
(Chen et al., 2016a)	semi-bandit	general	$O(\frac{nk}{\Delta}\log(T))$
(Rejwan and Mansour, 2020)	full-bandit	linear reward	$O(\frac{nk}{\Delta}\log(T))$
(Katariya et al., 2017)	full-bandit	Bernoulli, K rows L colns	$O(\frac{K+L}{\Delta}\log(T))$
(Agarwal et al., 2021)	full-bandit	stochastic dominance	$\tilde{O}(n^{1/3}k^{1/2}T^{2/3})$
(Kveton et al., 2015b)	cascading-bandit	Bernoulli	$O(\frac{nk}{\Delta}\log(T))$
(Sui et al., 2017)	relative	approx linearity	$O(\frac{n}{\Delta}\log(T))$

pared to the full-bandit setting, we assume additional information of maximum-value index, which is a natural assumption to be made in real-world applications. On the other hand, we do not assume per-item value feedback, which differentiates with the semi-bandit problem. Our work is one step towards solving the full-bandit CMAB problem with non-linear reward functions under mild assumptions.

- We use a reduction to the case of arm outcomes according to binary distributions and, for analysis, use an extended set of base arms associated with values and probabilities of outcomes. In the case when the ordering of values is known within each action, the problem boils down to a standard CMAB-T problem instance. In the case when the ordering of values is unknown, the problem differs from CMAB-T as the triggered subset of the base arm set given an action depends on whether the item values are observed or not. We tackle with this difficulty by introducing the concept of item equivalence, such that we can restore the CMAB-T framework by using replacement items.

- We present a Combinatorial Upper Confidence Bound (CUCB) algorithm to solve the k-MAX problem under the case that the ordering of values is known to the learner. We prove that the algorithm achieves comparable regret bound as standard CMAB problems. The CUCB algorithm does not directly apply to the case of unknown ordering of values. We propose a variant of the CUCB algorithm based on the concept of item equivalence. We show that the modified algorithm achieves comparable regret as the case of the known ordering of values. This means we do not lose much under limited feedback.

Organization of the paper. In Section 2, we formally define the problem. In Section 3, we first prove key properties of the reward function that will be used for the regret analysis. Then we present our algorithms and regret bounds for two settings of the problem for binary distributions. In Section 4, we discuss extension to arbitrary discrete distributions with finite supports. Section 5 contains our numerical results. Finally, we summarize our work in Section 6. Proofs of theorems are provided in Appendix.

2. Problem formulation

We consider a sequential decision making problem involving an agent and a set of n base arms[1], denoted as $E = [n] = \{1, 2, \ldots, n\}$. For each base arm $i \in E$, the outcomes are independent and identically distributed over time steps, according to distribution of a random variable X_i that has a discrete distribution with a finite support. We let $0 = v_{i,0} < v_{i,1} < \cdots < v_{i,s_i}$ denote values of the support of distribution of X_i, where s_i is a positive integer, and $s_i + 1$ is the support size. Let $p_{i,j} = \Pr[X_i = v_{i,j}]$ for $j \in \{0, 1, \ldots, s_i\}$, with $0 < \sum_{j=1}^{s_i} p_{i,j} \leq 1$. Let $\boldsymbol{v} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$ and $\boldsymbol{p} = (\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n)$ where $\boldsymbol{v}_i = (v_{i,1}, \ldots, v_{i,s_i})$ with $v_{i,j} \in [0, 1]$, and $\boldsymbol{p}_i = (p_{i,1}, \ldots, p_{i,s_i})$. For the special case of binary distributions, we write p_i and v_i in lieu of $p_{i,1}$ and $v_{i,1}$, respectively. Both $\boldsymbol{v}$ and $\boldsymbol{p}$, as well as the s_i's in the general case, are unknown parameters to the learning agent.

We define $\mathcal{F} = \{S \in 2^E \mid |S| = k\}$ as the set of arms of cardinality k. At each time step t, the agent takes an action to play an arm $S_t \in \mathcal{F}$. The agent observes the maximum value of the selected arms and the index of an item achieving the maximum value. The goal is to select a set of random variables with maximum performance according to the expected maximum objective.

When an action is played, the agent obtains a non-negative reward of the maximum value, which is fully determined by the triggered arms. We denote the expected reward as $r_S(\boldsymbol{p}, \boldsymbol{v}) = \mathbb{E}[\max\{X_i \mid i \in S\}]$, which is a function of action S and parameters $\boldsymbol{p}$ and $\boldsymbol{v}$.

The performance of a learning algorithm is measured by the cumulative regret, which is defined as the difference in expected cumulative reward by playing the best action and playing actions suggested by the algorithm. Denote $\mathrm{OPT}(\boldsymbol{p}, \boldsymbol{v}) = \max\{r_S(\boldsymbol{p}, \boldsymbol{v}) \mid S \in \mathcal{F}\}$. An (α, β)-approximation oracle takes $(\boldsymbol{p}', \boldsymbol{v}')$ as input and returns a set S such that $\Pr[r_S(\boldsymbol{p}', \boldsymbol{v}') \geq \alpha\, \mathrm{OPT}(\boldsymbol{p}', \boldsymbol{v}')] \geq \beta$ where α is the approximation ratio and β is the success probability. If the learning agent uses an (α, β)-approximation oracle, then we evaluate its performance by the (α, β)-approximation

[1] We use the terms base arm, arm and item interchangeably.

regret defined as

$$R(T) = T\,\alpha\,\beta\,\mathrm{OPT}(\boldsymbol{p},\boldsymbol{v}) - \mathbb{E}\left[\sum_{t=1}^{T} r_{S_t}(\boldsymbol{p},\boldsymbol{v})\right].$$

The offline k-MAX problem can be solved either by a greedy algorithm to achieve a $(1-1/e)$ approximate solution, or by a polynomial-time approximation scheme (PTAS) to achieve a $(1-\varepsilon)$ approximate solution for any $\varepsilon > 0$ (Chen et al., 2016a). For the special case of binary distributions, an exact solution can be found by using a dynamic programming algorithm (Chen and Du, 2022).

3. Algorithms and regret bounds for binary distributions

In this section we present algorithms and regret bounds for binary distributions. We first show some properties of reward functions which are crucial for our regret analysis. We then present an algorithm for the case when the ordering of $v_1,\ldots,v_n$ values is known. In this case, we will see that the problem can be reduced to a CMAB-T instance solvable using the standard CUCB method. Then we move to the general case when the ordering of $v_1,\ldots,v_n$ is a priori unknown. We present an algorithm and show that this algorithm achieves the same regret bound as when the ordering is known up to constant factors.

For the convenience of exposition, we assume that values $v_1,\ldots,v_n$ are distinct. This ensures that for any action S_t, there is a unique item achieving the maximum value over the items in S_t. This is equivalent to allowing for non-unique values and using a deterministic tie-breaking rule.

We introduce two sets of base arms decomposed from the random variables $X_1,\ldots,X_n$. The first set of base arms $\mathcal{Z}$ consists of n independent Bernoulli random variables $Z_1,\ldots,Z_n$ with mean values $p_1,\ldots,p_n$. The second set of base arms $\mathcal{V} = \{V_1,\ldots,V_n\}$ are deterministic with mean values $v_1,\ldots,v_n$. We also define an extended set of base arms $\mathcal{B}$ containing both sets of base arms. Note that we have $X_i = V_i Z_i$. Each time an action S_t is played, we obtain information on some of the base arms Z_i and V_i in $\mathcal{B}$. We call these arms as being triggered, and we observe their values as feedbacks. We define $T_{i,t}$ as the number of triggering times for Z_i and $\tilde{T}_{i,t}$ as the number of triggering times for V_i up to time step t. For any two vectors $\boldsymbol{x},\boldsymbol{x}' \in \mathbb{R}^n$, we write $\boldsymbol{x} \geq \boldsymbol{x}'$ if $x_i \geq x'_i$ for all $i \in [n]$.

3.1. Properties of reward functions

For the case of binary distributions of base arm outcomes, for any set $S \in \mathcal{F}$, under assumption that base arms are ordered in decreasing order of their values $v_1,\ldots,v_n$, the expected reward can be expressed as

$$r_S(\boldsymbol{p},\boldsymbol{v}) = \sum_{i\in S} v_i p_i \prod_{j\in S: j<i} (1-p_j) \tag{3.1}$$

where a product over an empty set has value 1.

There are two key properties of the regret function that we leverage for analysing regret of algorithms.

Monotonicity The first property is monotonicity.

Lemma 3.1. *$r_S(\boldsymbol{p},\boldsymbol{v})$ is increasing in every p_i and v_i.*

Recall that for a given set of random variables, we can explicitly write $r_S(\boldsymbol{p},\boldsymbol{v})$ as in equation (3.1). It is clear from the expression that $r_S(\boldsymbol{p},\boldsymbol{v})$ is monotonic increasing in v_i. We can prove that it is monotonic increasing in p_i by taking first derivative with respect to p_i and showing that the derivative is greater than zero.

Smoothness The second condition is relative triggering probability modulated (RTPM) smoothness. This is a slight generalization of the condition in (Wang and Chen, 2017), which allows for item-specific weights. Let $q_i^{\boldsymbol{\mu},S}$ denote the triggering probability of a base arm i in a set of base arms $\mathcal{B}$ with expectation $\boldsymbol{\mu}$ for action S.

Definition 3.2 (RTPM smoothness)**.** The problem satisfies the RTPM smoothness condition with respect to the base arm set $\mathcal{B}$ if, for any two distributions with expectation vectors $\boldsymbol{\mu}$ and $\boldsymbol{\mu}'$ and any action S,

$$|r_S(\boldsymbol{\mu}) - r_S(\boldsymbol{\mu}')| \leq \sum_{i\in S} q_i^{\boldsymbol{\mu},S} b_i |\mu_i - \mu'_i|,$$

where b_i is some per-arm weight coefficient. Note that when $r_S(\boldsymbol{\mu})$ is monotonic increasing in $\boldsymbol{\mu}$, and $\boldsymbol{v} \geq \boldsymbol{\mu}'$, we can remove the absolute value notation.

Note that we add the triggering probability $q_i^{\boldsymbol{\mu},S}$ and a relative weight coefficient b_i to *modulate* the standard 1-norm condition By allowing for arm-specific weights, we account for item-specific values as we will see in our next lemmas. The intuition is that we underweight the importance of items with small triggering probability or weight in expected reward. Even if for some item i we cannot estimate its expected value accurately, we lose very little in the expected reward. This will be a very important concept in the regret analysis that follows.

Consider an arbitrary set and without loss of generality assume that arms are ordered in decreasing value. Let the triggering probability of Z_i for action S be $q_i^{\boldsymbol{p},S}$ and the triggering probability of V_i for action S be $\tilde{q}_i^{\boldsymbol{p},S}$. The triggering probability for Z_i for action S is

$$q_i^{\boldsymbol{p},S} = (1-p_1)(1-p_2)\cdots(1-p_{i-1})$$

and the triggering probability for V_i for action S is

$$\tilde{q}_i^{\boldsymbol{p},S} = (1-p_1)(1-p_2)\cdots(1-p_{i-1})p_i.$$

We note that Z_i is triggered when the winner item has value smaller than or equal to v_i, while V_i is triggered when arm i is the winner, thus $\tilde{q}_i^{\boldsymbol{p},S} = q_i^{\boldsymbol{p},S} p_i$.

The following is a key lemma for our regret analysis.

Lemma 3.3. *The expected maximum reward function (3.1) satisfies the RTPM condition (Definition 3.2) with respect to the extended base arm set $\mathcal{B}$: for any S, $\boldsymbol{p}$ and $\boldsymbol{p}'$, and $\boldsymbol{v}$ and $\boldsymbol{v}'$ such that $\boldsymbol{v} \geq \boldsymbol{v}'$, it holds*

$$\begin{aligned} & |r_S(\boldsymbol{p},\boldsymbol{v}) - r_S(\boldsymbol{p}',\boldsymbol{v}')| \\ \leq \; & 2\textstyle\sum_{i\in S} q_i^{\boldsymbol{p},S} v_i' |p_i - p_i'| + \sum_{i\in S} \tilde{q}_i^{\boldsymbol{p},S} |v_i - v_i'|. \end{aligned}$$

Furthermore, if $\boldsymbol{p} \geq \boldsymbol{p}'$, then we can remove the factor 2 in the last inequality.

The lemma can be intuitively explained as follows. When an arm i has small value v_i and the corresponding base arm Z_i is unlikely to be triggered (small q), its importance in regret analysis diminishes. On the other hand, if the arm is unlikely to win (small $\tilde{q}$), it is also not important in our analysis. This concept is important throughout our proof for the main theorem. For items with small values or items whose values are hardly to be observed, we may not be able to estimate their value and probability parameters accurately. The lemma suggests that it is not a serious issue as these items are not important for our analysis.

To prove Lemma 3.3, we consider a sequence of vectors changing from $(\boldsymbol{p},\boldsymbol{v})$ to $(\boldsymbol{p}',\boldsymbol{v}')$ and add up the changes in expected rewards. The full proof is in Appendix B.2.

3.2. Algorithm for known ordering of values

We use a similar CUCB algorithm as standard CMAB problem to estimate parameters $\boldsymbol{p}$ and $\boldsymbol{v}$. Estimates of both sets of parameters are initialized to one at the beginning. Each time we observe v_j as the maximum value of the set, we update the corresponding estimates for v_j and the estimates for p_i, for items in the action set ordered before j. The algorithm maintains an upper confidence bound (UCB) for both parameters and feeds the UCB to the approximation oracle to obtain the next action. We note that for this case, our problem can be interpreted as a conjunctive cascading bandit (Kveton et al., 2015b) with binary-valued arms. The ordering of arms within each action enables us to observe values of all arms ordered before the winner, which makes the problem easier to solve.

For each action $S \in \mathcal{F}$, we define the gap $\Delta_S = \max\{\alpha \mathrm{OPT}(\boldsymbol{p},\boldsymbol{v}) - r_S(\boldsymbol{p},\boldsymbol{v}), 0\}$. We call an action *bad* if its gap is positive. For arms that are contained in at least one bad action, we define,

$$\Delta_{\min}^i = \min_{S: i\in S, q_i^{\boldsymbol{p},S}, \tilde{q}_i^{\boldsymbol{p},S} > 0, \Delta_S > 0} \Delta_S,$$

Algorithm 1 CUCB algorithm for known ordering of values

For $i \in E$, $T_i \leftarrow 0$ {Num. of triggering times for Z_i}
For $i \in E$, $\hat{p}_i \leftarrow 1$, $\hat{v}_i \leftarrow 1$ {Initial est. of params}
for $t = 1, 2, \ldots$ **do**
  For $i \in E$, $\rho_i \leftarrow \sqrt{\frac{3\log(t)}{2T_i}}$ {Confidence radius}
  For $i \in E$, $\bar{p}_i \leftarrow \min\{\hat{p}_i + \rho_i, 1\}$, $\bar{v}_i \leftarrow \hat{v}_i$ {UCB}
  $S \leftarrow \mathrm{Oracle}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}})$ {Oracle decides the next action}
  Play S and observe winner index i^* and value v_{i^*}
  Update $\hat{v}_{i^*}$ for winner item i^*: $\hat{v}_{i^*} \leftarrow v_{i^*}$
  For $i \in S$ s. t. $i \leq i^*$: $T_i \leftarrow T_i + 1$
  For $i \in S$ s. t. $i < i^*$: $\hat{p}_i \leftarrow (1 - 1/T_i)\hat{p}_i$
  $\hat{p}_{i^*} \leftarrow (1 - 1/T_{i^*})\hat{p}_{i^*} + 1/T_{i^*}$
end for

$$\Delta_{\max}^i = \max_{S: i\in S, q_i^{\boldsymbol{p},S}, \tilde{q}_i^{\boldsymbol{p},S} > 0, \Delta_S > 0} \Delta_S$$

where $q_i^{\boldsymbol{p},S}, \tilde{q}_i^{\boldsymbol{p},S} > 0$ require that Z_i, V_i are triggered by action S with non-zero probabilities. For other arms, we define $\Delta_{\min}^i = \infty$ and $\Delta_{\max}^i = 0$. Then we define $\Delta_{\min} = \min_{i\in[n]} \Delta_{\min}^i$ and $\Delta_{\max} = \max_{i\in[n]} \Delta_{\max}^i$.

The regret bound for the algorithm is provided as follows.

Theorem 3.4. *Under assumption $\Delta_{\min} > 0$, Algorithm 1 has the following distribution-dependent regret bound,*

$$\begin{aligned} R(T) \leq & c_1 \sum_{i=1}^n \left(\frac{v_i^2 k}{\Delta_{\min}^i} + \log\left(\frac{k}{\Delta_{\min}^i} + 1\right)\right) \log(T) \\ & + c_2 \sum_{i=1}^n \left(\left(\log\left(\frac{v_i k}{\Delta_{\min}^i} + 1\right) + 1\right) \Delta_{\max} + v_i \right), \end{aligned}$$

for some positive constants c_1 and c_2.

This regret bound achieves $O((nk/\Delta)\log(T))$ which is comparable to the regret upper bound for the standard CMAB-T problem (Chen et al., 2016a), which in turn is tight with respect to dependence on T in comparison with the lower bound in (Kveton et al., 2015a). The only term in the regret bound that depends on horizon time T is the first summation term. In this summation term, the summands have two terms, one scaling linearly with $k/\Delta_{\min}^i$ and other scaling logarithmically with $k/\Delta_{\min}^i$, which are due to uncertainty of parameters $\boldsymbol{p}$ and $\boldsymbol{v}$, respectively. Hence, we may argue that the uncertainty about values of parameters $\boldsymbol{p}$ has more effect on regret than uncertainty about values of parameters $\boldsymbol{v}$. The regret bound in Theorem 3.4 implies a $\tilde{O}(\sqrt{T})$ distribution-independent regret bound.

To see how the algorithm can be boiled down into two CMAB-T problems, we consider the contribution of each action to regret, i.e, $\Delta_{S_t} = \max\{\alpha \ \mathrm{OPT}(\boldsymbol{p},\boldsymbol{v}) - r_{S_t}(\boldsymbol{p},\boldsymbol{v}), 0\}$. Let $\mathcal{F}_t$ be the good event $\{r_{S_t}(\bar{\boldsymbol{p}},\bar{\boldsymbol{v}}) \geq \alpha \ \mathrm{OPT}(\boldsymbol{p},\boldsymbol{v})\}$ that the approximation oracle works well.

By the smoothness condition, under $\mathcal{F}_t$ we have

$$\begin{aligned}\Delta_{S_t} &\le r_{S_t}(\bar{\boldsymbol{p}}_t, \bar{\boldsymbol{v}}_t) - r_{S_t}(\boldsymbol{p}, \boldsymbol{v}) \\ &\le \sum_{i \in S_t} q_i^{\boldsymbol{p},S} v_i(\bar{p}_{i,t} - p_i) + \sum_{i \in S_t} \tilde{q}_i^{\boldsymbol{p},S}(\bar{v}_{i,t} - v_i).\end{aligned}$$

Clearly, the first term corresponds to regret from the set of base arms $\mathcal{Z}$, and the second term corresponds to regrets from the set of base arms $\mathcal{V}$. We bound Δ_{S_t} by bounding the two summation terms individually. The first summation term is standard in literature. Note that we need to take extra steps to bound the second summation term, as our estimates for v_i will not be more and more accurate as the number of selected times increase. The UCB of v_i remains at the upper bound value 1 until the base arm i in $\mathcal{V}$ is triggered once and we then know the exact value of v_i. We show the full proof in Appendix B.3.

3.3. Algorithm for unknown ordering of values

In the general case, the agent does not know the ordering of $v_1, \ldots, v_n$ for all actions. This greatly decreases the information that can be deduced from information feedback. To see this, we consider each arm $i \in \mathcal{V}$ in two stages, before and after its value v_i is observed.

In the first stage when v_i is unknown, the corresponding base arm Z_i has not been triggered yet. Note the in the simpler case where the ordering of the values are known, Z_i is triggered whenever arm i is ordered before the winner j. This is because in this case we can deduce that Z_i has to be zero, since otherwise arm i with a higher value v_i would beat arm j and j cannot be the winner. However, since the ordering is unknown in the general case, we can no longer carry out the above deduction and it is unclear whether $Z_i = 0$ or $Z_i = 1$. More specifically, suppose that in round t we play set S_t, and item $j \in S_t$ with value v_j is the winner and value-index pair (v_j, j) is observed. For an arm $i \in S_t$, we have not observed v_i so do not know whether $v_i > v_j$ or $v_i < v_j$. For the first case, arms i could take a non-zero value that is not observed, while it takes zero value for the other case. Importantly, we note that the triggering of Z_i is dependent on whether knowing the value of v_i or not. This is different from the CMAB framework and thus we cannot simply reduce this setting back to an equivalent CMAB setting.

On the other hand, when V_i is triggered once and v_i becomes known, then the corresponding random variable Z_i is triggered whenever the winner value is smaller than v_i. We can immediately conclude that Z_i takes value zero. Thus the analysis for second stage is the same as for the case of known ordering of values.

A naive approach is to adopt the CUCB algorithm for the simpler case and introduce $\tilde{T}_i$ as the triggering time for V_i. We update parameters of item i only when $\tilde{T}_i \neq 0$. However, this approach could fail for items i with large v_i and small p_i. Note that the estimate of p_i will not be updated until v_i is observed. However, the upper bound of 1 for p_i is clearly an overestimate for this type of items, which would cause large regrets during the period when their values are not observed. This will be reflected as an undesirable factor in the regret upper bound.

Algorithm 2 CUCB alg. for unknown ordering of values

For $i \in E$, $T_i \leftarrow 0$, $\tilde{T}_i \leftarrow 0$ {Num. trig. for Z_i and V_i}
For $i \in E$, $\hat{p}_i \leftarrow 1$, $\hat{v}_i \leftarrow 1$ {Initial est. of params}
for $t = 1, 2, \ldots$ **do**
  For $i \in E$, $\rho_i \leftarrow \sqrt{\frac{3\log(t)}{2T_i}}$, $\tilde{\rho}_i \leftarrow \mathbf{1}\{\tilde{T}_i = 0\}$
  For $i \in E$, $\bar{p}_i \leftarrow \min\{\hat{p}_i + \rho_i, 1\}$, $\bar{v}_i \leftarrow \min\{\hat{v}_i + \tilde{\rho}_i, 1\}$ {UCBs}
  $S \leftarrow \text{Oracle}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}})$ {Oracle decides the next action}
  Play S and observe winner index i^* and value v_{i^*}
  if $\tilde{T}_{i^*} = 0$ **then**
    Reset $T_{i^*} \leftarrow 0, \quad \tilde{T}_{i^*} \leftarrow 1, \quad \hat{v}_{i^*} \leftarrow v_{i^*}$
  end if
  For $i \in S$ s. t. $\hat{v}_i \ge v_{i^*}$: $T_i \leftarrow T_i + 1$
  For $i \in S$ s. t. $\hat{v}_i > v_{i^*}$: $\hat{p}_i \leftarrow (1 - 1/T_i)\hat{p}_i$
  For $i \in S$ s. t. $\hat{v}_i = v_{i^*}$: $\hat{p}_i \leftarrow (1 - 1/T_i)\hat{p}_i + 1/T_i$
end for

To remove the extra factor, we propose a variant of the CUCB algorithm (Algorithm 2). In this algorithm, we do not wait to update p_i after observing v_i. We start with optimistic initial estimates such that every item has high chance of being a winner. We use the estimates $\hat{v}_i$ and pretend that Z_i is triggered and takes value zero when v_i is not observed. In this way, the true winners will gradually stand out, while we are still giving chances to those items whose values are not observed yet. This intuitively makes sense as even if v_i takes value 1, the above-mentioned type of items will not be important to our regret analysis due to their small probability parameter estimates.

Let $\Delta^i_{\min}$ be the gaps as in the case of known ordering of values. The regret bound for the algorithm for the case of unknown ordering of values is provided as follows.

Theorem 3.5. *Under assumption $\Delta_{min} > 0$, Algorithm 2 has the following regret bound,*

$$\begin{aligned}R(T) \le & c_1 \sum_{i=1}^{n} \left(\frac{k}{\Delta^i_{\min}} + \log\left(\frac{k}{\Delta^i_{\min}} + 1 \right) \right) \log(T) \\ & + c_2 \sum_{i=1}^{n} \left(\left(\log\left(\frac{v_i k}{\Delta^i_{\min}} + 1 \right) + 1 \right) \Delta_{\max} + 1 \right),\end{aligned}$$

for some positive constants c_1 and c_2.

This regret bound in Theorem 3.5 achieves $O((nk/\Delta)\log(T))$ and agrees with the bound for

the simpler case in Theorem 3.4 up to constant factors. The regret bound in Theorem 3.5 implies a $\tilde{O}(\sqrt{T})$ distribution-independent regret bound.

We note that our problem does not fit into the standard CMAB-T framework. As discussed above, we are *pretending* that Z_i is triggered and takes value zero. This may not be the ground truth in the case when v_i is actually less than the winner value. Therefore, using the observation $Z_i^{(t)} = 0$ will make the estimate biased, not following the standard CMAB-T framework. In particular, for items with small v_i and large p_i values, we clearly underestimate their p_i values, since this type of items could take non-zero value but not observed due to small v_i. On the other hand, intuitively these items are not important due to small value of v_i. This means we cannot simply apply the regret result of CMAB-T or follow its analysis to reach our result.

To tackle this difficulty in our analysis, we introduce the concept of *item equivalence*. In each round t, for every item i with parameters (p_i, v_i) and $\tilde{T}_{i,t} = 0$, we replace it with equivalent item i' with parameters (p'_i, v'_i) where $p'_i = p_i v_i$ and $v'_i = 1$. Note that items with small v_i and large p_i are mapped to equivalent items with large v_i and small p_i, for which our improved algorithm can estimate accurately. We formally justify this equivalence in the regret analysis.

Proof sketch In the following we give a sketch of the proof of Theorem 3.5. The full proof can be found in Appendix B.5.

We use a similar framework for regret analysis as the CUCB method. However, note that one of the key assumptions for CUCB algorithm fails to hold in our setting, i.e., we do not always have upper confidence bounds for parameters p_i. Thus, we use new technical steps to account for this.

Firstly, we notice the following fact when replacing item i with parameters (p_i, v_i) by its equivalent item i' with parameters (p'_i, v'_i).

Lemma 3.6. *For any set $S \in \mathcal{F}$, if $(\boldsymbol{p}', \boldsymbol{v}')$ is the equivalent form of $(\boldsymbol{p}, \boldsymbol{v})$ as defined above, then $r_S(\boldsymbol{p}, \boldsymbol{v}) \le r_S(\boldsymbol{p}', \boldsymbol{v}')$.*

Then we consider the contribution of each action to regret Δ_t. Under the good event $\mathcal{F}_t$ that the approximation oracle works well, i.e. $r_{S_t}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}}) \ge \alpha\ \mathrm{OPT}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}})$. By Lemma 3.6, for each t such that $1 \le t \le T$ we have,

$$\alpha\ \mathrm{OPT}(\boldsymbol{p}'_t, \boldsymbol{v}'_t) \ge \alpha\ \mathrm{OPT}(\boldsymbol{p}, \boldsymbol{v}). \tag{3.2}$$

Thus,

$$\begin{aligned}\Delta_{S_t} &\le \alpha\ \mathrm{OPT}(\boldsymbol{p}'_t, \boldsymbol{v}'_t) - r_{S_t}(\boldsymbol{p}, \boldsymbol{v}) \\ &\le \alpha\ \mathrm{OPT}(\boldsymbol{p}'_t, \boldsymbol{v}'_t) - r_{S_t}(\boldsymbol{p}, \boldsymbol{v}) + r_{S_t}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}}) - \alpha\ \mathrm{OPT}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}}) \\ &\le r_{S_t}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}}) - r_{S_t}(\boldsymbol{p}, \boldsymbol{v}) \\ &= (r_{S_t}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}}) - r_{S_t}(\boldsymbol{p}'_t, \boldsymbol{v}'_t)) + (r_{S_t}(\boldsymbol{p}'_t, \boldsymbol{v}'_t) - r_{S_t}(\boldsymbol{p}, \boldsymbol{v})),\end{aligned}$$

where the first inequality is due to condition (3.2), the second inequality is due to the approximation oracle, and the third inequality is due to monotonicity of r_S in $\boldsymbol{p}$ and $\boldsymbol{v}$. We call the term inside first bracket as the regret caused by estimation error $\Delta^e_{S_t}$, and the term inside the second bracket as the regret caused by replacement error $\Delta^r_{S_t}$. To obtain a tight regret upper bound, we require that the regret caused by replacement error over the time horizon T is not greater than the that by estimation error, i.e, under a series of good events, $\sum_{t=1}^T \Delta^r_{S_t} \le \sum_{t=1}^T \Delta^e_{S_t}$. This would justify the intuition of using replacement items. Now we look closely at these two terms separately.

By Lemma 3.3, we have

$$\Delta^e_{S_t} \le \sum_{i \in S_t} q_i^{\boldsymbol{p},S} v'_{i,t}(\bar{p}_{i,t} - p'_{i,t}).$$

Note that we do not need to include the v_i term as $v'_{i,t} = \bar{v}_i = 1$ for all i when v_i is not observed, and $v'_{i,t} = \bar{v}_i = v_i$ after v_i is observed. In both cases, there is no estimation error for v_i.

We also apply Lemma 3.3 to the second summation term to obtain

$$\Delta^r_{S_t} \le 2 \sum_{i \in S_t} q_i^{\boldsymbol{p},S} v_i (p_i - p'_{i,t}) + \sum_{i \in S_t} \tilde{q}_i^{\boldsymbol{p},S} (v'_{i,t} - v_i).$$

To sum up, we have

$$\begin{aligned}\Delta_{S_t} &\le \textstyle\sum_{i \in S_t} q_i^{\boldsymbol{p},S} v'_{i,t}(\bar{p}_{i,t} - p'_{i,t}) \\ &+ 2 \textstyle\sum_{i \in S_t} q_i^{\boldsymbol{p},S} v_i (p_i - p'_{i,t}) + \sum_{i \in S_t} \tilde{q}_i^{\boldsymbol{p},S} (v'_{i,t} - v_i).\end{aligned}$$

We bound Δ_{S_t} by bounding these error terms in different cases. We can bound the first term by following the proof of the regret bound for the standard CMAB-T problem, stated in Theorem A.1 for completeness. To see this, recall that we have reset the counts T_i and the estimates p_i at the time v_i is observed. This is because $p'_{i,t} = p_i v_i$ when v_i is unknown and $p'_{i,t} = p_i$ afterwards. However, for both stages our estimates are accurate in the sense that p'_i always lies within the confidence interval which decreases as the counter number increases.

For the second term, we note that $p'_{i,t} = p_i v_i$ in the first stage, and $p'_i = p_i$ after v_i is observed. Therefore, the contribution to regret by the second term is zero in the second stage. For the first stage, this term can be analyzed in the similar way as the last term. The key observation is that $p_i - p'_{i,t} = p_i(1 - v_i) \le p_i$. This will be the key for removing the extra factor.

Finally, we note that the analysis for the last summation term is the same as the simpler case, since there is no change to the triggering process of V_i's. Summing up the bounds over time horizon T, we can prove the main theorem.

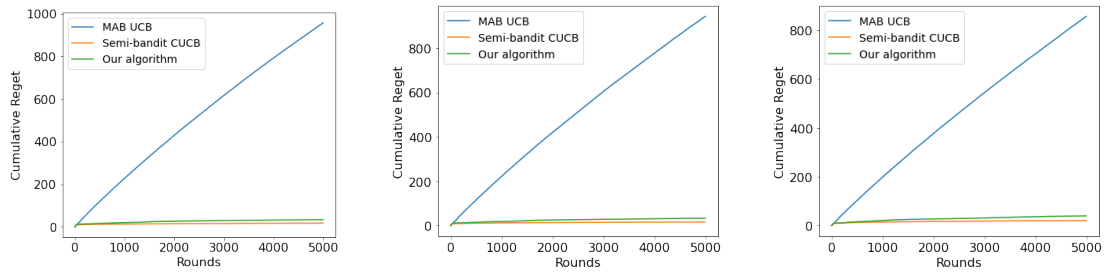

Figure 1. Cumulative regrets for Algorithm 2 for different distributions of arm outcomes as defined in Appendix D.

4. Arbitrary distributions with finite supports

We discuss the case of arbitrary discrete distributions with finite supports. We show that it is possible to turn a set of multi-valued variables equivalently into a set of binary variables. This implies that our algorithms and regret bounds apply to the general case of discrete distributions with finite supports.

To see this, let X_i be a random variable with an arbitrary discrete distribution with finite support as defined in Section 2. Recall that $\Pr[X_i = v_{i,j}] = p_{i,j}$, for $j \in \{0, 1, \ldots, s_i\}$, where $v_{i,j} \in [0, 1]$ and $v_{i,0} = 0$. Let $X_{i,1}, \ldots, X_{i,s_i}$ be independent binary random variables such that $X_{i,j}$ takes value $v_{i,j}$ with probability $\tilde{p}_{i,j}$, and value 0 otherwise. We consider $\max\{X_{i,j} \mid j \in [s_i]\}$, which takes values in $\{0, v_{i,1}, \ldots, v_{i,s_i}\}$. Let

$$\tilde{p}_{i,j} = \begin{cases} \frac{p_{i,j}}{1-\sum_{l=j+1}^{s_i} p_{i,l}} & \text{if } 1 \le j < s_i \\ p_{i,s_i} & \text{if } j = s_i. \end{cases} \tag{4.1}$$

Then note that $\max\{X_{i,j} \mid j \in [s_i]\}$ has the same distribution as the original random variable X_i. In this way, we establish the equivalence between binary variables and any discrete variables with finite support in terms of the $\max$ operator. This means that we can use our algorithm to solve the k-MAX bandit for any such finite-valued variables.

For example, if we know all the possible values, we can order them, and then use the algorithm with known value orders. For the case we do not know the possible values, we can use the algorithm with unknown value orders. We make slight modifications to relax the requirement of knowing item distribution support sizes. The key is that we introduce a counter $\sigma(i)$ to denote the number of observed values of X_i and dynamically maintain a list of values for X_i. We use a fictitious arm with value 1 as a placeholder for those base arms whose values are not yet observed. The UCBs of binary base arm parameters are mapped back to the multi-valued form using (4.1). We refer to the k-MAX PTAS in (Chen et al., 2016a) as our offline approximation oracle. For space reasons, more details are included in Appendix C.

5. Numerical results

We perform experiments to evaluate performance of Algorithm 2 on some specific problem instances. We compare our algorithm with two baseline methods: the well-known UCB algorithm treating each set of size k as one arm, and standard CUCB algorithm for semi-bandit CMAB setting. We use the greedy algorithm as the offline oracle.

We consider settings with $n = 9$ arms and sets of cardinality $k = 3$. We tested on three different arm distributions representing different scenarios. We expect the algorithm to perform well for all three cases. For space reasons, we show the detailed setup and regret plots in Appendix D. We run each experiment for horizon time $T = 5000$. In each round, we select arms according to the offline oracle and sample their values for updates. We compare the reward to that of optimal set S^*. We repeat the experiments for 20 times and average over the cumulative regrets.

Results Figure 1 shows the regrets of Algorithm 2 and two baseline methods for the three cases. We can see that our algorithm achieve much lower regrets compared to the UCB algorithm. Our regret curve is close to that of CUCB method under semi-bandit CMAB setting, which confirms that we do not lose much when we have much fewer feedback.

6. Conclusion

We studied the k-MAX combinatorial multi-armed bandit problem under maximum value-index feedback. This feedback lies in between semi-bandit and full-bandit feedback, and provides much less information than under semi-bandit feedback. We proposed a CUCB algorithm for the case when the ordering of values is known. For the case when the value ordering is uknown, we proposed a new algorithm based on the concept of item equivalence. We showed that algorithms guarantee a regret upper bound that is matching that under semi-bandit feedback up to constant factors.

Future work may consider whether the same regret bound can be achieved for the k-MAX problem under full-bandit feedback, and consider CMAB problems under feedback that lies in between semi-bandit and full-bandit feedback.

References

A. Agarwal, N. Johnson, and S. Agarwal. Choice bandits. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18399–18410. Curran Associates, Inc., 2020.

M. Agarwal, V. Aggarwal, C. J. Quinn, and A. K. Umrawal. Stochastic top-k subset bandits with linear space and non-linear feedback. In *Algorithmic Learning Theory*, pages 306–339. PMLR, 2021.

N. Ailon, Z. Karnin, and T. Joachims. Reducing dueling bandits to cardinal bandits. In *International Conference on Machine Learning*, pages 856–864. PMLR, 2014.

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

G. E. Blau, J. F. Pekny, V. A. Varma, and P. R. Bunch. Managing a portfolio of interdependent new product candidates in the pharmaceutical industry. *Journal of Product Innovation Management*, 21(4):227–245, 2004.

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

O. Chapelle and Y. Zhang. A dynamic bayesian network click model for web search ranking. In *Proceedings of the 18th international conference on World wide web*, pages 1–10, 2009.

W. Chen and Y. Du, 2022. private communication.

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

W. Chen, W. Hu, F. Li, J. Li, Y. Liu, and P. Lu. Combinatorial multi-armed bandit with general reward functions. *Advances in Neural Information Processing Systems*, 29, 2016a.

W. Chen, Y. Wang, Y. Yuan, and Q. Wang. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *Journal of Machine Learning Research*, 17(50):1–33, 2016b.

V. Dani, T. P. Hayes, and S. M. Kakade. Stochastic linear optimization under bandit feedback. 2008.

A. Gopalan, S. Mannor, and Y. Mansour. Thompson sampling for complex online problems. In *International conference on machine learning*, pages 100–108. PMLR, 2014.

A. Jekunen. Decision-making in product portfolios of pharmaceutical research and development–managing streams of innovation in highly regulated markets. *Drug Des Devel Ther.*, 2014.

S. Katariya, B. Kveton, C. Szepesvari, C. Vernade, and Z. Wen. Stochastic rank-1 bandits. In *Artificial Intelligence and Statistics*, pages 392–401. PMLR, 2017.

J. Kleinberg and M. Raghu. Team performance with test scores. *ACM Trans. Econ. Comput.*, 6(3–4), Oct. 2018. ISSN 2167-8375.

B. Kveton, Z. Wen, A. Ashkan, and C. Szepesvari. Tight regret bounds for stochastic combinatorial semi-bandits. In *Artificial Intelligence and Statistics*, pages 535–543. PMLR, 2015a.

B. Kveton, Z. Wen, A. Ashkan, and C. Szepesvari. Combinatorial cascading bandits. *Advances in Neural Information Processing Systems*, 28, 2015b.

D. Lee, M. Vojnovic, and S.-Y. Yun. Test score algorithms for budgeted stochastic utility maximization. *INFORMS Journal on Optimization*, 0(0):null, 2022.

R. D. Luce. *Individual Choice Behavior: A Theoretical analysis*. Wiley, New York, NY, USA, 1959.

C. D. Manning, P. Raghavan, and H. Schutze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

A. Mehta, U. Nadav, A. Psomas, and A. Rubinstein. Hitting the high notes: Subset selection for maximizing expected order statistics. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 15800–15810. Curran Associates, Inc., 2020.

I. Rejwan and Y. Mansour. Top-k combinatorial bandits with full-bandit feedback. In *Algorithmic Learning Theory*, pages 752–776. PMLR, 2020.

S. Sekar, M. Vojnovic, and S. Yun. A test score-based approach to stochastic submodular optimization. *Manag. Sci.*, 67(2):1075–1092, 2021.

Y. Sui, V. Zhuang, J. W. Burdick, and Y. Yue. Multi-dueling bandits with dependent arms. *arXiv preprint arXiv:1705.00253*, 2017.

L. L. Thurstone. A law of comparative judgment. *Psychological Review*, 34(4):273–286, 1927. doi: 10.1037/h0070288.

Q. Wang and W. Chen. Improving regret bounds for combinatorial semi-bandits with probabilistically triggered arms and its applications. *Advances in Neural Information Processing Systems*, 30, 2017.

A. CMAB-T framework and additional notation

We review the framework and results for the classical CMAB problem with triggered arms considered by (Wang and Chen, 2017). In this problem, the expected reward is a function of action S and expectation vector $\boldsymbol{\mu}$ of base arms. Denote the probability that action S triggers arm i as $p_i^{\boldsymbol{\mu},S}$. It is assumed that in each round the value of triggered arms are observed by the agent. The CUCB algorithm (Chen et al., 2013) is used to estimate the expectation vector $\boldsymbol{\mu}$ directly from samples.

The following is Theorem 1 in (Wang and Chen, 2017) for the standard CMAB problem with triggered arms (CMAB-T). It is assumed that the CMAB-T problem instance satisfies monotonicity and 1-norm TPM bounded smoothness (Definition 3.2 with $b_i = B$ for all $i \in [n]$). We will use some proof steps and result of this theorem in proofs of our results.

Theorem A.1. *For the CUCB algorithm on a CMAB-T problem instance satisfying monotonicity and 1-norm TPM bounded smoothness, we have the following distribution-dependent regret bound,*

$$R(T) \leq 576B^2k\left(\sum_{i=1}^{n}\frac{1}{\Delta_{\min}^i}\right)\log(T) + \left(\sum_{i=1}^{n}\left(\log\left(\frac{2Bk}{\Delta_{\min}^i}+1\right)+2\right)\right)\frac{\pi^2}{6}\Delta_{\max} + 4Bn$$

where $\Delta_{\min} = \inf_{S:i\in S, p_i^{\boldsymbol{\mu},S}>0, \Delta_S>0}\Delta_S$.

We next give various definitions used in our analysis. The definitions are given specifically to binary distributions.

We define two sets of *triggering probability (TP) groups*. Let j be a positive integer. For the set $\mathcal{Z}$ of base arms we define the triggering probability group $\mathcal{S}_{i,j}$ as

$$\mathcal{S}_{i,j} = \{S \in \mathcal{F} \mid 2^{-j} < q_i^{\boldsymbol{p},S} \leq 2^{-j+1}\}.$$

We define the triggering probability group $\tilde{\mathcal{S}}_{i,j}$ for the set $\mathcal{V}$ of base arms similarly. We note that the triggering probability groups divide actions that trigger arm i into separated groups such that the actions in the same group contribute similarly to the regret bound.

Let $N_{i,j,t}$ be the counter of the cumulative number of times i in TP group $S_{i,j}$ is selected at the end of round t. Under clear context, we also use $N_{i,j,t}$ to denote the counter of the cumulative number of times i in TP group $\tilde{S}_{i,j}$ is selected at the end of round t.

We define the *event-filtered regret* for a sequence of events $\{\mathcal{E}_t\}$ as

$$R(T, \{\mathcal{E}_t\}) = T\,\alpha\,\mathrm{OPT}(\boldsymbol{p}, \boldsymbol{v}) - \mathbb{E}\left[\sum_{t=1}^{T}\mathbf{1}(\mathcal{E}_t)r_{S_t}(\boldsymbol{p}, \boldsymbol{v})\right]$$

which means that the regret is accounted for in round t only if event $\mathcal{E}_t$ occurs in round t.

We next define four good events as follows:

E1 The approximation oracle works well, i.e.

$$\mathcal{F}_t = \{r_{S_t}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}}) \geq \alpha\,\mathrm{OPT}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{v}})\}.$$

E2 The parameter vector $\boldsymbol{p}$ is estimated well, i.e. for every $i \in [n]$ and $t \geq 1$,

$$\mathcal{N}_t = \{|\hat{p}_{i,t-1} - p_i| < \rho_{i,t}\}$$

where $\hat{p}_{i,t-1}$ is the estimator of p_i at round t and $\rho_{i,t} := \sqrt{3\log(t)/(2T_{i,t-1})}$.

E3 Triggering is nice for $\mathcal{Z}$ given a set of integers $\{j_{\max}^i\}_{i\in[n]}$, i.e. for every TP group $\mathcal{S}_{i,j}$ defined by arm i and $1 \leq j \leq j_{\max}^i$, under the condition $\sqrt{6\log(t)/(N_{i,j,t-1}/3)2^{-j}} \leq 1$, there is $T_{i,t-1} \geq \frac{1}{3}N_{i,j,t-1}2^{-j}$. We denote this event with $\mathcal{N}_t'$.

E4 Triggering is nice for $\mathcal{V}$, i.e. for every arm $i \in E$, under the condition $N_{i,j,t-1} \geq 3p_i^{-1}\log(t)2^j$, there is $\tilde{T}_{i,t-1} \neq 0$. Equivalently, we can define this event in terms of TP group $\tilde{S}_{i,j}$ by removing the factor of p_i^{-1}. We denote this event with $\tilde{\mathcal{N}}_t'$.

Note that events E1, E2 and E3 are also used in (Wang and Chen, 2017), and event E4 is newly defined for our problem. We can easily show the following bound for the probability of the event E4 using the Hoeffding's inequality.

Lemma A.2. *For each round $t \geq 1$, we have* $\Pr(\neg\tilde{\mathcal{N}}'_t) \leq n/t^2$.

B. Proofs

B.1. Proof of Lemma 3.1

Without loss of generality, consider $S = \{1, \ldots, k\}$ under assumption $v_1 \geq \cdots \geq v_k$. The expected regret can be written as

$$r_S(\boldsymbol{p}, \boldsymbol{v}) = p_1 v_1 + (1 - p_1) p_2 v_2 + \cdots + (1 - p_1) \cdots (1 - p_{k-1}) p_k v_k.$$

It is clear from the expression that $r_S(\boldsymbol{p}, \boldsymbol{v})$ is monotonic increasing in v_i.

Let us consider $r_S(\boldsymbol{p}, \boldsymbol{v})$ for arbitrarily fixed $\boldsymbol{v}$. Taking derivative with respect to p_i, we have

$$\frac{d}{dp_i} r_S(\boldsymbol{p}, \boldsymbol{v}) = (1 - p_1) \cdots (1 - p_{i-1}) \left[v_i - p_{i+1} v_{i+1} - \left(\sum_{j>i} (1 - p_{i+1}) \cdots (1 - p_j) p_{j+1} v_{j+1} \right) \right].$$

We claim that the term inside the bracket is greater than zero. This can be shown as follows,

$$\begin{aligned}
& v_i - p_{i+1} v_{i+1} - \left(\sum_{j>i} (1 - p_{i+1}) \cdots (1 - p_j) p_{j+1} v_{j+1} \right) \\
\geq\ & v_i (1 - p_{i+1}) - (1 - p_{i+1}) p_{i+2} v_{i+2} - (1 - p_{i+1})(1 - p_{i+2}) p_{i+3} v_{i+3} - \cdots \\
\geq\ & (1 - p_{i+1})(1 - p_{i+2}) \cdots (1 - p_{k-1})(v_i - p_k v_k) \\
\geq\ & (1 - p_{i+1})(1 - p_{i+2}) \cdots (1 - p_k) v_i \\
\geq\ & 0
\end{aligned}$$

Thus, the reward function is monotonic increasing in p_i.

B.2. Proof of Lemma 3.3

For the purpose of this proof only, we assume that the items are ordered in decreasing order of values for both $\boldsymbol{v}$ and $\boldsymbol{v}'$.

Let $\boldsymbol{p} = (p_1, \ldots, p_k)$ and $\boldsymbol{p}' = (p'_1, \ldots, p'_k)$, and $\boldsymbol{p}^{(0)} = \boldsymbol{p}'$, $\boldsymbol{p}^{(j)} = (p_1, \ldots, p_j, p'_{j+1}, \ldots, p'_k)$, for $1 \leq j < k$, and $\boldsymbol{p}^{(k)} = \boldsymbol{p}$. Similarly, let $\boldsymbol{v} = (v_1, \ldots, v_k)$ and $\boldsymbol{v}' = (v'_1, \ldots, v'_k)$, and $\boldsymbol{v}^{(0)} = \boldsymbol{v}'$, $\boldsymbol{v}^{(j)} = (v_1, \ldots, v_j, v'_{j+1}, \ldots, v'_k)$, for $1 \leq j < k$, and $\boldsymbol{v}^{(k)} = \boldsymbol{v}$.

Since $\boldsymbol{v} \geq \boldsymbol{v}'$, the item ordering is preserved, i.e., $v_1 \geq v_2 \geq \ldots \geq v'_{j+1} \geq v'_k$ and we have,

$$r_S(\boldsymbol{p}^{(j)}, \boldsymbol{v}^{(j)}) = p_1 v_1 + \cdots + (1 - p_1) \cdots (1 - p_{j-1}) p_j v_j + (1 - p_1) \cdots (1 - p_j) p'_{j+1} v'_{j+1} + \cdots$$

and

$$r_S(\boldsymbol{p}^{(j-1)}, \boldsymbol{v}^{(j-1)}) = p_1 v_1 + \cdots + (1 - p_1) \cdots (1 - p_{j-1}) p'_j v'_j + (1 - p_1) \cdots (1 - p'_j) p'_{j+1} v'_{j+1} + \cdots$$

Note that the only difference is caused by position j. By definition of triggering probabilities $q_i^{\boldsymbol{p},S}$ and $\tilde{q}_i^{\boldsymbol{p},S}$ we can write,

$$\begin{aligned}
|r_S(\boldsymbol{p}^{(j)}, \boldsymbol{v}^{(j)}) - r_S(\boldsymbol{p}^{(j-1)}, \boldsymbol{v}^{(j-1)})| =& |q_j^{\boldsymbol{p},S} (p_j v_j - p'_j v'_j - \sum_{i>j} (1 - p'_{j+1}) \ldots (1 - p'_{i-1}) p'_i v'_i (p_j - p'_j)| \\
\leq& q_j^{\boldsymbol{p},S} p_j |v_j - v'_j| + q_j^{\boldsymbol{p},S} v'_j |p_j - p'_j| \\
&+ q_j^{\boldsymbol{p},S} (p'_{j+1} v'_{j+1} + (1 - p'_{j+1}) v'_{j+2} + \ldots) |p_j - p'_j| \\
\leq& 2 q_j^{\boldsymbol{p},S} v'_j |p_j - p'_j| + \tilde{q}_j^{\boldsymbol{p},S} |v_j - v'_j|
\end{aligned}$$

where the first inequality is due to the triangle inequality and the second inequality is due to the monotonicity property. Note that if we have $\boldsymbol{p} \geq \boldsymbol{p}'$, we only need to consider the first two terms in the second line.

Summing up over j we can obtain the statement of the lemma.

B.3. Proof of Theorem 3.4

We consider the contribution of each action to regret Δ_t, for every $t \geq 1$. Let $M_i := \Delta^i_{\min}$. Recall that $\Delta_S = \max\{\alpha \cdot \mathrm{OPT}_{\boldsymbol{p},\boldsymbol{v}} - r_S(\boldsymbol{p},\boldsymbol{v}), 0\}$. Assume that $\Delta_{S_t} \geq M_{S_t}$ where $M_S = \max_{i \in S} M_i$. Note that if $\Delta_{S_t} < M_{S_t}$ then $\Delta_{S_t} = 0$, since we have either an empty set, or $\Delta_{S_t} < M_{S_t} < M_i$ for some $i \in S_t$.

By the smoothness condition, we have

$$\Delta_{S_t} \leq r_{S_t}(\bar{\boldsymbol{p}}_t, \bar{\boldsymbol{v}}_t) - r_{S_t}(\boldsymbol{p},\boldsymbol{v}) \leq \sum_{i \in S_t} q_i^{\boldsymbol{p},S} v_i(\bar{p}_{i,t} - p_i) + \sum_{i \in S_t} \tilde{q}_i^{\boldsymbol{p},S}(\bar{v}_{i,t} - v_i).$$

Since $\Delta_{S_t} \geq M_{S_t}$, we add and subtract M_{S_t} from the last expression and we have,

$$\begin{aligned}\Delta_{S_t} &\leq -M_{S_t} + 2\left(\sum_{i \in S_t} q_i^{\boldsymbol{p},S} v_i(\bar{p}_{i,t} - p_i) + \sum_{i \in S_t} \tilde{q}_i^{\boldsymbol{p},S}(\bar{v}_{i,t} - v_i)\right) \\ &\leq 2\left(\sum_{i \in S_t} q_i^{\boldsymbol{p},S} v_i(\bar{p}_{i,t} - p_i) - \frac{M_i}{4k}\right) + 2\left(\sum_{i \in S_t} \tilde{q}_i^{\boldsymbol{p},S}(\bar{v}_{i,t} - v_i) - \frac{M_i}{4k}\right).\end{aligned}$$

Let us call the first term $\Delta^p_{S_t}$ and the second term $\Delta^v_{S_t}$. We bound Δ_{S_t} by bounding the two summation terms individually.

Note that for $\Delta^p_{S_t}$ we can bound following the same procedure as in the proof for Theorem A.1. However, we cannot use the same procedure for $\Delta^v_{S_t}$. The key difference is that our estimate for v_i will not be more and more accurate as the number of selections of the item i increases. We know the exact value of v_i as soon as it is triggered once. We assume that arm i is in TP group $\tilde{S}_{i,j}$. Let j_i be the index of the TP group with $S_t \in \tilde{S}_{i,j_i}$. We take $j^i_{\max} = \lceil \log(4k/M_i + 1) \rceil$.

- Case 1: $1 \leq j_i \leq j^i_{\max}$. In this case, $\tilde{q}_i^{\boldsymbol{p},S} \leq 2 \cdot 2^{j_i}$,

  $$\tilde{q}_i^{\boldsymbol{p},S}(\bar{v}_{i,t} - v_i) \leq 2 \cdot 2^{-j_i}\, \mathbb{1}\{\tilde{T}_{i,t} = 0\}$$

  Under the good event $\tilde{\mathcal{N}}^t_t$, when $N_{i,j_i,t-1} \geq 3\log(t) \cdot 2^j$, the contribution to regret is zero. Otherwise, it is bounded as

  $$\tilde{q}_i^{\boldsymbol{p},S}(\bar{v}_{i,t} - v_i) \leq 2 \cdot 2^{-j_i}.$$

- Case 2: $j_i > j^i_{\max}$. In this case,

  $$\tilde{q}_i^{\boldsymbol{p},S}(\bar{v}_{i,t} - v_i) \leq 2 \cdot 2^{-j_i} \leq \frac{M_i}{4k}.$$

  Thus, the term does not contribute to regret in this case.

We next calculate the filtered regret under the good events mentioned above and the event that $\Delta_{S_t} \geq M_{S_t}$. Note that

$$R(T, \{\Delta_{S_t} \geq M_{S_t}\}, \mathcal{F}_t, \mathcal{N}^s_t, \mathcal{N}^t_t, \tilde{\mathcal{N}}^t_t) \leq \sum_{t=1}^T \Delta^p_{S_t} + \sum_{t=1}^T \Delta^v_{S_t}.$$

For the event E3 we set $j^i_{\max} = \log(4v_i k/M_i + 1)$. By Theorem A.1 in Appendix A, we know that under good events E1, E2 and E3, the first term is bounded by

$$\sum_{t=1}^T \Delta^p_{S_t} \leq 1152k\left(\sum_{i=1}^k \frac{v_i^2}{M_i}\right)\log(T) + 4\left(\sum_{i=1}^n v_i\right)$$

and the corresponding filtered regrets for the case where events E1, E2 or E3 fail to hold are bounded by,

$$\begin{aligned}&R(T, \neg\mathcal{F}_t) \leq (1-\beta)T \cdot \Delta_{max} \\ &R(T, \neg\mathcal{N}^s_t) \leq \frac{\pi^2}{3} n \Delta_{\max} \\ &R(T, \neg\mathcal{N}^t_t) \leq \frac{\pi^2}{6}\left(\sum_{i=1}^n \log\left(\frac{4v_i k}{M_i} + 1\right)\right)\Delta_{\max}.\end{aligned}$$

Now we focus on bounding the contribution of the second term to regret. Note that under event E4,

$$\sum_{t=1}^{T} \Delta^{v}_{S_t} = \sum_{i=1}^{n} \sum_{j=1}^{\infty} \sum_{s=0}^{N_{i,j,T-1}} \kappa_{j_i,T}(M_i, s)$$

where

$$\kappa_{j,T}(M,s) = \begin{cases} 2 \cdot 2^{-j} & \text{if } s < 3\log(T) \cdot 2^{j} \\ 0 & \text{if } s \geq 3\log(T) \cdot 2^{j} \end{cases}.$$

For every arm i and $j \geq 1$, we have

$$\sum_{s=0}^{N_{i,j,T-1}} \kappa_{j_i,T}(M_i, s) \leq \sum_{s=0}^{3\log(T)\cdot 2^{j_i}} \kappa_{j_i,T}(M_i, s) = 6\log(T).$$

Hence, the contribution of the second term to regret is bounded by

$$\sum_{t=1}^{T} \sum_{i \in \tilde{S}_t} \tilde{\kappa}_{j_i,T}(M_i, N_{i,j_i,t-1}) \leq 6\left(\sum_{i=1}^{n} \log\left(\frac{4k}{M_i} + 1\right)\right)\log(T).$$

The filtered regrets for the case when event E4 fails to hold is bounded by,

$$R(T, \neg\tilde{\mathcal{N}}'_t) \leq \sum_{i=1}^{T} \Pr(\neg\tilde{\mathcal{N}}'_t)\Delta_{max} \leq \frac{\pi^2}{6} n\Delta_{\max}.$$

We obtain the distribution-dependent regret by adding up the filtered regrets calculated above. The corresponding distribution-independent regret is implied by taking $M_i = \sqrt{16nk/T}$ for every $i \in [n]$.

B.4. Proof of Lemma 3.6

Without loss of generality assume that $S = [k]$ and $v_1 \geq v_2 \geq \cdots \geq v_k$. Recall that we can write

$$r_S(\boldsymbol{p}, \boldsymbol{v}) = p_1 v_1 + (1-p_1)p_2 v_2 + \ldots + (1-p_1)\ldots(1-p_{k-1})p_k v_k.$$

Now for $\boldsymbol{p} = (p_1, \ldots, p_k)$ and $\boldsymbol{p}' = (p'_1, \ldots, p'_k)$, let

$$\boldsymbol{p}^{(j)} = (p'_1, \ldots, p'_j, p_{j+1}, \ldots, p_k)$$

and define similarly $\boldsymbol{v}^{(j)}$ for $\boldsymbol{v}$ and $\boldsymbol{v}'$. After changing p_1 to $p'_1 = p_1 v_1$ and v_1 to $v'_1 = 1$,

$$r_S(\boldsymbol{p}^{(1)}, \boldsymbol{v}^{(1)}) = p_1 v_1 + (1-p_1 v_1)p_2 v_2 + \cdots + (1-p_1 v_1)\cdots(1-p_{k-1})p_k v_k.$$

Clearly we have $r_S(\boldsymbol{p}^{(1)}, \boldsymbol{v}^{(1)}) \geq r_S(\boldsymbol{p}, \boldsymbol{v})$. Following the same argument, we can see that $r_S(\boldsymbol{p}^{(2)}, \boldsymbol{v}^{(2)}) \geq r_S(\boldsymbol{p}^{(1)}, \boldsymbol{v}^{(1)})$. Continuing this way to $r_S(\boldsymbol{p}^{(k)}, \boldsymbol{v}^{(k)})$ we can prove the lemma.

B.5. Proof of Theorem 3.5

By the general smoothness condition, we have

$$\Delta_{S_t} \leq \sum_{i \in S_t} q_i^{\boldsymbol{p},S} v'_{i,t}(\bar{p}_{i,t} - p'_{i,t}) + 2\sum_{i \in S_t} q_i^{\boldsymbol{p},S} v_i(p_i - p'_i) + \sum_{i \in S_t} \tilde{q}_i^{\boldsymbol{p},S}(v'_{i,t} - v_i).$$

Key step: bounding the contribution of each action to regret Let $M_i = \Delta^{i}_{\min}$. Assume that $\Delta_{S_t} \geq M_{S_t}$ where $M_S = \max_{i \in S} M_i$. As in the known value ordering case, we can bound Δ_{S_t} such that,

$$\begin{aligned}\Delta_{S_t} &\leq -M_{S_t} + 2\left(\sum_{i \in S_t} q_i^{\boldsymbol{p},S} v'_{i,t}(\bar{p}_{i,t} - p'_{i,t}) + 2\sum_{i \in S_t} q_i^{\boldsymbol{p},S} v_i(p_i - p'_i) + \sum_{i \in S_t} \tilde{q}_i^{\boldsymbol{p},S}(v'_{i,t} - v_i)\right) \\ &\leq 2\left[\left(\sum_{i \in S_t} q_i^{\boldsymbol{p},S} v'_{i,t}(\bar{p}_{i,t} - p'_{i,t}) - \frac{M_i}{8k}\right) + 2\left(\sum_{i \in S_t} q_i^{\boldsymbol{p},S} v_i(p_i - p'_i) - \frac{M_i}{8k}\right) + \left(\sum_{i \in S_t} \tilde{q}_i^{\boldsymbol{p},S}(v'_{i,t} - v_i) - \frac{M_i}{4k}\right)\right].\end{aligned} \tag{B.1}$$

Let j_i be the index of the TP group of S_t such that $S_t \in S_{i,j_i}$. We bound Δ_{S_t} by bounding the three summation terms in (B.1) separately.

- Bounding the first term. Recall that we reset the $T_{i,t}$ and $N_{i,j,t}$ at the time we observe v_i. This is because $p'_{i,t} = p_i v_i$ and $v'_{i,t} = 1$ when v_i is unknown (first stage) and $p'_{i,t} = p_i$ and $v'_{i,t} = v_i$ after v_i is observed (second stage). A key observation is that within both stages our estimates are accurate in the sense that under event E2, the approximation error decreases as the counter number increases in the following way.

$$\bar{p}_{i,t} - p'_{i,t} \leq 2\rho_i = 2\sqrt{\frac{3\log(t)}{2T_{i,t-1}}}.$$

  We note that for the second stage where v_i is observed, this term is similar to the $\Delta^p_{S_t}$ term in the known value ordering case. Specifically, under event E3,

$$\bar{p}_{i,t} - p'_{i,t} \leq \min\left\{\sqrt{\frac{18\log(t)}{2^{-j_i} \cdot N_{i,j_i,t-1}}}, 1\right\}$$

  and

$$q_i^{\boldsymbol{p},S} v'_{i,t}(\bar{p}_{i,t} - p'_{i,t}) \leq \min\left\{\sqrt{\frac{72 \cdot 2^{-j_i} v_i^2 \log(T)}{N_{i,j_i,t-1}}}, 2 \cdot 2^{-j_i} v_i\right\}.$$

  For the event E3, let $j^i_{\max} = \lceil \log(8v_i k/M_i + 1) \rceil$. In the case $j_i \geq j^i_{\max} + 1$, this term does not contribute to regret as we have,

$$q_i^S(\bar{p}_{i,t} - p'_{i,t}) \leq 2 \cdot 2^{-j_i} \leq \frac{M_i}{8k}$$

  Similarly, for $1 \leq j_i \leq j^i_{\max}$, there is no contribution to regret if $N_{i,j,t-1} \geq l_{j_i,T}(M_i)$ where

$$l_{j,T}(M) = \left\lfloor \frac{4608 \cdot 2^{-j} v_i^2 k^2 \log(T)}{M^2} \right\rfloor.$$

  For the first stage, we note that $T_{i,t-1} \geq N_{i,j,t-1}$. This does not require the event E3 to hold, as triggering is always nice for Z_i in the first stage. Thus for the first stage we have,

$$\bar{p}_{i,t} - p'_{i,t} \leq \min\left\{\sqrt{\frac{6\log(t)}{N_{i,j_i,t-1}}}, 1\right\}$$

  and

$$q_i^{\boldsymbol{p},S} v'_{i,t}(\bar{p}_{i,t} - p'_{i,t}) \leq \min\left\{\sqrt{\frac{24 \cdot (2^{-j_i})^2 \log(T)}{N_{i,j_i,t-1}}}, 2 \cdot 2^{-j_i}\right\}$$

  Now we take $j^i_{\max} = \log(8k/M_i + 1)$. We redefine $j^i_{\max}$ to consider different subsets of TP groups. Again, there is no contribution to regret in the case $j_i \geq j^i_{\max} + 1$. For $1 \leq j_i \leq j^i_{\max}$, this term does not contribute to regret if $N_{i,j,t-1} \geq l'_{j_i,T}(M_i)$ where

$$l'_{j,T}(M) = \left\lfloor \frac{1536 \cdot (2^{-j})^2 k^2 \log(T)}{M^2} \right\rfloor.$$

- Bounding the second term. Take $j^i_{\max} = \log(8k/M_i + 1)$. Similarly as the first term, in the case $j_i \geq j^i_{\max} + 1$, the contribution to regret is non-positive. For $1 \leq j_i \leq j^i_{\max}$, as $p'_{i,t} = p_i v_i$, we have

$$q_i^{\boldsymbol{p},S} v_i(p_i - p'_{i,t}) \leq 2 \cdot 2^{-j_i} p_i v_i (1 - v_i).$$

  Under the event E4, we know that the contribution to regret is zero if $N_{i,j_i,t-1} \geq 3p_i^{-1}\log(T) \cdot 2^j$. Otherwise, it is upper bounded by $2 \cdot 2^{-j_i} p_i$.

- Bounding the third term. Take $j^i_{\max} = \log(8k/M_i + 1)$. In the case $j_i \geq j^i_{\max} + 1$, the contribution to regret is non-positive. For $1 \leq j_i \leq j^i_{\max}$, we have $\tilde{q}_i^{\boldsymbol{p},S} \leq 2 \cdot 2^{-j_i} p_i$, thus

$$\tilde{q}_i^{\boldsymbol{p},S}(\bar{v}_{i,t} - v_i) \leq 2 \cdot 2^{-j_i} p_i \tilde{\rho}_i = 2 \cdot 2^{-j_i} p_i \cdot \mathbb{1}\{\tilde{T}_{i,t} = 0\}.$$

  Under the event E4, we know that the contribution to regret is zero if $N_{i,j_i,t-1} \geq 3p_i^{-1}\log(T) \cdot 2^j$. Otherwise, it is upper bounded by

$$\tilde{q}_i^{\boldsymbol{p},S}(\bar{v}_{i,t} - v_i) \leq 2 \cdot 2^{-j_i} p_i$$

  Note that this bound is the same as the bound for the second term.

Summing over the time horizon Next, we sum up Δ_{S_t} over time T and calculate the filtered regret under the above mentioned good events and the event that $\Delta_{S_t} \geq M_{S_t}$, i.e, $R(\{\Delta_{S_t} \geq M_{S_t}\}, \mathcal{F}_t, \mathcal{N}_t, \mathcal{N}_t^t, \tilde{\mathcal{N}}_t^t)$.

By equation (B.1), we know that the filtered regret can be upper bounded by sum of three terms over the time horizon T. By Theorem A.1 in Appendix A, we know under good events E1, E2 and E3, the second stage of the first term is bounded by

$$2\left(\sum_{i\in S_t} q_i^{\boldsymbol{p},S} v_i(\bar{p}_{i,t} - p'_{i,t}) - \frac{M_i}{8k}\right) \leq 2304k\left(\sum_{i=1}^n \frac{v_i^2}{M_i}\right)\log(T) + 4\sum_{i=1}^n v_i$$

and the corresponding filtered regrets for the case when E1, E2 or E3 fail to hold are bounded by,

$$\begin{aligned}
R(T, \neg\mathcal{F}_t) &\leq (1-\beta)T\Delta_{\max} \\
R(T, \neg\mathcal{N}_t^s) &\leq \frac{\pi^2}{3} n\Delta_{\max} \\
R(T, \neg\mathcal{N}_t^t) &\leq \frac{\pi^2}{6}\sum_{i=1}^n \log\left(\frac{8v_i k}{M_i} + 1\right)\Delta_{\max}.
\end{aligned}$$

To bound the first term, we also need to derive a bound for the first stage when the value is not observed. This is bounded by,

$$\begin{aligned}
2\left(\sum_{i\in S_t} q_i^{\boldsymbol{p},S}(\bar{p}_{i,t} - p'_{i,t}) - \frac{M_i}{8k}\right) &\leq \sum_{i=1}^n\sum_{j=1}^\infty \left(4\cdot 2^{-j} + \sum_{N_{i,j_i,t-1}=1}^{l'_{j,T}(M_i)} 2\sqrt{\frac{24\cdot(2^{-j_i})^2\log(T)}{N_{i,j_i,t-1}}}\right) \\
&\leq \sum_{i=1}^n\sum_{j=1}^\infty 4\cdot 2^{-j} + 4\sqrt{l'_{j,T}(M_i)}\sqrt{24\cdot(2^{-j_i})^2\log(T)} \\
&\leq 256k\left(\sum_{i=1}^n \frac{1}{M_i}\right)\log(T) + 4n.
\end{aligned}$$

By the analysis for the Algorithm 3.4, we know that

$$2\sum_{t=1}^T\left(\sum_{i\in S_t} q_i^{\boldsymbol{p},S}(v'_{i,t} - v_i) - \frac{M_i}{4k}\right) \leq 12\sum_{i=1}^n \log\left(\frac{8k}{M_i} + 1\right)\log(T).$$

Similarly we can bound

$$4\sum_{t=1}^T\left(\sum_{i\in S_t} q_i^{\boldsymbol{p},S} v_i(p_i - p'_{i,t}) - \frac{M_i}{8k}\right) \leq 24\sum_{i=1}^n \log\left(\frac{8k}{M_i} + 1\right)\log(T).$$

The filtered regret for the case where event (E4) fails to hold is bounded by,

$$R(T, \neg\tilde{\mathcal{N}}'_t) \leq \sum_{i=1}^T \Pr(\neg\tilde{\mathcal{N}}'_t)\Delta_{max} \leq \frac{\pi^2}{6} n\Delta_{\max}.$$

We obtain the distribution-dependent regret by adding up the filtered regrets calculated above. Similarly as before, the distribution-dependent regret is implied by taking $M_i = \sqrt{64nk/T}$ for every $i \in [n]$.

Algorithm 3 CUCB algorithm for arbitrary distributions with finite supports

For $i \in E$, $\sigma(i) \leftarrow 0$ {Number of known values for arm i}
For $i \in E$, $T_{i,0} \leftarrow 0$, $\tilde{T}_{i,0} \leftarrow 0$ {Number of triggering times for the fictitious arm}
For $i \in E$, $\hat{p}_{i,0} \leftarrow 1$, $\hat{v}_{i,0} \leftarrow 1$ {Initial estimates of parameters for the fictitious arm}
for $t = 1, 2, \ldots$ **do**
  For $i \in E$ and $j \in [\sigma(i)]$, $\rho_{i,j} \leftarrow \sqrt{\frac{3\log(t)}{2T_{i,j}}}$, $\tilde{\rho}_{i,j} \leftarrow \mathbf{1}\{\tilde{T}_{i,j} = 0\}$ {Confidence radius of parameters}
  For $i \in E$ and $j \in [\sigma(i)]$, $\bar{p}_{i,j} \leftarrow \min\{\hat{p}_{i,j} + \rho_{i,j}, 1\}$, $\bar{v}_{i,j} \leftarrow \min\{\hat{v}_{i,j} + \tilde{\rho}_{i,j}, 1\}$ {UCB of parameters}
  Transform $\bar{\boldsymbol{p}}$ to $\tilde{\boldsymbol{p}}$ using Eq. (4.1)
  $S \leftarrow \text{Oracle}(\tilde{\boldsymbol{p}}, \bar{\boldsymbol{v}})$ {Oracle k-MAX PTAS decides the next action}
  Play S and observe winner index i^* and value v
  if $v \notin \{\hat{v}_{i^*,j}, j \in [\sigma(i^*)]\}$ **then**
    $\sigma(i^*) \leftarrow \sigma(i^*) + 1, \quad T_{i^*,\sigma(i^*)} \leftarrow 0, \quad \tilde{T}_{i^*,\sigma(i^*)} \leftarrow 1, \quad \hat{v}_{i^*,\sigma(i^*)} \leftarrow v$
  end if
  For $i \in S$ and $j \in [\sigma(i)]$ such that $\hat{v}_{i,j} \geq v$: $T_{i,j} \leftarrow T_{i,j} + 1$
  For $i \in S$ and $j \in [\sigma(i)]$ such that $\hat{v}_{i,j} > v$: $\hat{p}_{i,j} \leftarrow (1 - 1/T_{i,j})\hat{p}_{i,j}$
  For $i \in S$ and $j \in [\sigma(i)]$ such that $\hat{v}_{i,j} = v$: $\hat{p}_{i,j} \leftarrow (1 - 1/T_{i,j})\hat{p}_i + 1/T_{i,j}$
end for

C. CUCB algorithm for arbitrary distributions with finite supports

In this section, we present our Algorithm 3 for the general discrete distributions with finite support. The algorithm is an extension of Algorithm 2, with slight modifications that allow us to relax the assumption of knowing the support sizes of item distributions.

Recall that we work with binary base arms $\{X_{i,j}, i \in [n], j \in [s_i]\}$ where s_i is the support size of X_i. We use a counter $\sigma(i)$ to denote the number of observed values for X_i. We increase this counter and reset the triggering times and probability estimates for the $\sigma(i)$th base arm whenever we observe a new value for X_i. On the other hand, we use a fictitious arm with value 1 as placeholder for those base arms whose values remain unobserved. Since we have no information on the support size, we always keep this fictitious arm and update its probability estimates whenever X_i is selected in an action.

Note that we convert UCBs of the binary base arms to multi-valued forms according to relationship 4.1 and use the k-MAX PTAS in (Chen et al., 2016a) as the offline oracle. We give further explanations justifying this usage as follows. In the equivalent binary form, we would need an oracle such that for each binary arm $X_{i,j}$, if $X_{i,j} \in S$, then all $X_{i,j'}, j' \in [s_i]$ must also be selected. Because of the fact that $X_i = \max\{X_{i,j} \mid j \in [s_i]\}$, we just need to convert $\bar{\boldsymbol{p}}$ to $\bar{\boldsymbol{q}}$ and use the k-MAX PTAS as the offline oracle for the equivalent binary case.

D. Supplementary information for simulations

In this section, we provide supplementary information for Section 5 on numerical results.

Setup We consider settings with $n = 9$ arms and sets of cardinality $k = 3$ for the following distributions of arms.

D1 $\boldsymbol{v} = (0.1, 0.2, \ldots, 0.9)^\top$. $\boldsymbol{p}$ is such that $p_i = 0.3$, for $1 \leq i \leq 6$, and $p_i = 0.5$, otherwise.

D2 Compared to D1, we introduce an arm i with small v_i and large p_i. Specifically, for arm 1 we redefine $p_1 = 0.9$ and keep v_1 unchanged.

D3 Compared to D1, we introduce an arm i with large v_i and small p_i. Specifically, for the last arm we redefine $p_9 = 0.2$ and keep v_9 unchanged. Note that arm 9 has the same expected value as arm 6, but arm 9 is in the optimal set.

Note that the optimal super arm is $S^* = \{7, 8, 9\}$ in all cases. Distributions D1, D2 and D3 represent different scenarios. D1 is the base case. In D2, there is a stable arm with low value, while in D3 there is a high-risk high-reward arm. Both are not easy to observe and cause challenges for our algorithm design, especially the latter type of arms, which can outperform less-risky arms under the maximum value reward function.

Regret plots We show the regrets of Algorithm 2 and two baseline methods in Figure 1. We plot the 1-approximation regrets instead of $(1 - 1/e)$-approximation regret as the offline greedy oracle performs much better than $(1 - 1/e)$-approximation

in this case. We can see that our algorithm performs well, achieving much lower regrets in all cases.