

Cost-aware Cascading Bandits

Ruida Zhou¹, Chao Gan², Jing Yang², Cong Shen¹

¹ University of Science and Technology of China

² The Pennsylvania State University

zrd127@mail.ustc.edu.cn, cug203@psu.edu, yangjing@psu.edu, congshen@ustc.edu.cn

Abstract

In this paper, we propose a cost-aware cascading bandits model, a new variant of multi-armed bandits with cascading feedback, by considering the random cost of pulling arms. In each step, the learning agent chooses an *ordered* list of items and examines them sequentially, until certain stopping condition is satisfied. Our objective is then to maximize the expected *net reward* in each step, i.e., the reward obtained in each step minus the total cost incurred in examining the items, by deciding the ordered list of items, as well as when to stop examination. We study both the offline and online settings, depending on whether the state and cost statistics of the items are known beforehand. For the offline setting, we show that the Unit Cost Ranking with Threshold 1 (UCR-T1) policy is optimal. For the online setting, we propose a Cost-aware Cascading Upper Confidence Bound (CC-UCB) algorithm, and show that the cumulative regret scales in $O(\log T)$. We also provide a lower bound for all α -consistent policies, which scales in $\Omega(\log T)$ and matches our upper bound. The performance of the CC-UCB algorithm is evaluated with both synthetic and real-world data.

1 Introduction

In this paper, we introduce a new cost-aware cascading bandits (CCB) model. We consider a set of K items (arms) denoted as $[K] = \{1, 2, \dots, K\}$. Each item $i \in [K]$ has two possible states 0 and 1, which evolve according to an independent and identically distributed (i.i.d.) Bernoulli random variable. The learning agent chooses an *ordered* list of items in each step and examines them sequentially until certain stopping condition is met. The reward that the learning agent receives in a step equals one if one of the examined items in that step has state 1; Otherwise, it equals zero. We associate a random cost for examining each item. The overall reward function, termed as *net reward*, is the reward obtained in each step minus the total cost incurred in examining the items before the learner stops.

The CCB model is a natural but technically non-trivial extension of cascading bandits [Kveton *et al.*, 2015a], and is a

more suitable model in many fields, including the following examples.

- 1) *Opportunistic spectrum access*. In cognitive radio systems, a user is able to probe multiple channels sequentially at the beginning of a transmission session before it decides to use at most one of the channels for data transmission. Assuming the channel states evolve between *busy* and *idle* in time, the user gets a reward if there exists one *idle* channel for transmission. The cost then corresponds to the energy and delay incurred in probing each channel.
- 2) *Dynamic treatment allocation*. In clinic trials, a doctor must assign one of several treatments to a patient in order to cure a disease. The doctor accrues information about the outcome of previous treatments before making the next assignment. Whether a treatment cures the disease can be modeled as a Bernoulli random variable, and the doctor gets a reward if the patient is cured. The doctor may not only be interested in the expected effect of the treatment but also its riskiness, which can be interpreted as the cost of the treatment.

In both examples, the net reward in each step is determined not only by the subset of items included in the list, but also by the *order* that they are pulled. Intuitively, if the costs in examining the items are homogeneous, we would prefer to have the channel with higher probability to be idle, or the more effective treatment ranked higher in the list. Then, the learner would find an available channel or cure the patient after a few attempts and then stop examination, thus saving the cost without decreasing the reward. However, for more general cases where the costs are heterogeneous or even random, the optimal solution is not immediately clear.

In this paper, we consider a general cost model where the costs of pulling arms are heterogeneous and random, and investigate the corresponding solutions. Our main contributions are three-fold:

- 1) We propose a novel CCB model, which has implications in many practical scenarios, such as opportunistic spectrum access, dynamic treatment allocation, etc. The CCB model is fundamentally different from its cost-oblivious counterparts, and admits a unique structure in the corresponding learning strategy.

- 2) Second, with a priori statistical knowledge of the arm states and the costs, we explicitly identify the special structure of optimal policy (coined as the *offline* policy), which serves as the baseline for the online algorithm we develop. The optimal offline policy, called Unit Cost Ranking with Threshold 1 (UCR-T1), is to *rank* the arms based the statistics of their states and costs, and pull those above certain threshold sequentially until a state 1 is observed.
- 3) Third, we propose a cost-aware cascading Upper Confidence Bound (CC-UCB) algorithm for the scenario when prior arm statistics are unavailable, and show that it is order-optimal by establishing order-matching upper and lower bounds on the regret. Our analysis indicates that the UCB based algorithm performs well for ranking the arms, i.e., the cumulative regret of ranking the desired arms in a wrong order is bounded.

2 Related Literature

There have been some attempts that take the cost of pulling arms and budget constraint into the multi-armed bandit (MAB) framework recently. They can be summarized in two types. In the first type [Audibert and Bubeck, 2010; Bubeck *et al.*, 2009; Guha and Munagala, 2007], pulling each arm in the exploration phase has a unit cost and the goal is to find the best arm given the budget constraint on the total number of exploration arms. This type of problems is also referred to as “best-arm identification” or “pure exploration”. In the second type, pulling an arm is always associated with a cost and constrained by a budget, no matter in the exploration phase or the exploitation phase, and the objective usually is to design an arm pulling algorithm in order to maximize the total reward with given cost or budget constraint. References [Tran-Thanh *et al.*, 2010; Tran-Thanh *et al.*, 2012; Burnetas and Kanavetas, 2012] consider the problem when the cost of pulling each arm is *fixed* and becomes known after the arm is used once. A sample-path cost constraint with *known* bandit dependent cost is considered in [Burnetas *et al.*, 2017]. References [Ding *et al.*, 2013; Xia *et al.*, 2016a; Xia *et al.*, 2015; Xia *et al.*, 2016b] study the budgeted bandit problems with *random* arm pulling costs. Reference [Badani-diyuru *et al.*, 2013] considers the knapsack problem where there can be more than one budget constraints and shows how to construct policies with sub-linear regret.

In the proposed CCB model, the *net reward* function is related to the cost of pulling arms and the learning agent faces a “soft constraint” on the cost instead of a fixed budget constraint. If the learner only pulls one arm in each step, the cost of pulling an arm can be absorbed into the reward of that arm (i.e., net reward). Our model then reduces to a conventional MAB model for this case. In this paper, however, we are interested in the scenario where the learner is allowed to sequentially pull a number of arms in each step, and the reward obtained in each step cannot be decomposed into the summation of the rewards from the pulled arms. Thus, the cost cannot be simply absorbed into the reward of individual arms. The intricate relationship between the cost and the reward, and its implications on the optimal policy require more

sophisticated analysis, and that is the focus of this paper.

MAB with more than one arm to pull in each step has been studied in multiple-play MAB (MP-MAB) models in [Anantharam *et al.*, 1987; Komiyama *et al.*, 2015; Kalathil *et al.*, 2014], cascading bandits (CB) models in [Kveton *et al.*, 2015a; Kveton *et al.*, 2015b; Zong *et al.*, 2016], and ranked bandits (RB) models in [Radlinski *et al.*, 2008; Streeter and Golovin, 2008; Slivkins *et al.*, 2013]. Under MP-MAB, the learner is allowed to pull L out of K arms, and the reward equals to the summation of the rewards from individual arms. Although the proposed CCB model also allows the user to pull multiple arms in each step, the reward is not accumulative, thus leading to different solutions.

The CCB model proposed in this paper is closely related to the CB model investigated in [Kveton *et al.*, 2015a]. Specifically, [Kveton *et al.*, 2015a] considers the scenario where at each step, L out of K items are listed by a learning agent and presented to a user. The user examines the ordered list from the first to the last, until he/she finds the first attractive item and clicks it. The system receives a reward if the user finds at least one of the items to be attractive. Our model has the same reward model as that in the cascading bandits setting. However, there are also important distinctions between them. In the CB model in [Kveton *et al.*, 2015a], the total number of items to be examined each step is fixed, and the cost of pulling individual arms is not considered. As a result, the same subset of items on a list will give the same expected reward, irrespective of their order on the list. However, for the proposed CCB model, the ranking of the items on a list does affect the expected net reward. Therefore, the structure of the optimal offline policy and the online algorithm we develop are fundamentally different from those in [Kveton *et al.*, 2015a].

The proposed CCB model is also related to RB in the sense that the order of the arms to pull in each step matters. A crucial feature of RB is that the click probability for a given item may depend on the item and its position on the list, as well as the items shown above. However, in our case, we assume the state of an arm evolves in an i.i.d. fashion from step to step.

3 System Model and Problem Formulation

Consider a K -armed stochastic bandit system where the state of each arm evolves independently from step to step. Let $X_{i,t}$ be the state of arm $i \in [K]$ in step t . Then, $X_{i,t} \in \{0, 1\}$ evolves according to an i.i.d. Bernoulli distribution with parameter θ_i . Denote $Y_{i,t}$ as the cost of pulling arm i in step t , where $Y_{i,t} \in [0, 1]$ evolves according to an i.i.d. unknown distribution with $\mathbb{E}[Y_{i,t}] = c_i$.

In step t , the learning agent chooses an ordered list of arms from $[K]$ and pull the arms sequentially. Once an arm is pulled, its state and the pulling cost are revealed instantly. Denote the ordered list as $I_t := \{I_t(1), I_t(2), \dots, I_t(|I_t|)\}$, where $I_t(i)$ is the i th arm to be pulled, and $|I_t|$ is the cardinality of I_t . Denote \tilde{I}_t as the list of arms that have been actually pulled in step t . We have $\tilde{I}_t \subseteq I_t$.

The *reward* that the learning agent receives in step t depends on both \tilde{I}_t and $\{X_{i,t}\}_{i \in \tilde{I}_t}$. Specifically, it can be ex-

pressed as $1 - \prod_{i=1}^{|\tilde{I}_t|} (1 - X_{\tilde{I}_t(i),t})$, i.e., the learning agent gets reward one if one of the arms that have been examined in step t has state 1; Otherwise, it equals zero. The *cost* that is incurred at step t also depends on \tilde{I}_t , and it can be expressed as $\sum_{i=1}^{|\tilde{I}_t|} Y_{\tilde{I}_t(i),t}$.

With a given ordered list I_t , \tilde{I}_t is random and its realization depends on the observed $X_{i,t}$, $Y_{i,t}$ and the stopping condition in general. Denote the *net reward* received by the learning agent at step t as $r_t := 1 - \prod_{i=1}^{|\tilde{I}_t|} (1 - X_{\tilde{I}_t(i),t}) - \sum_{i=1}^{|\tilde{I}_t|} Y_{\tilde{I}_t(i),t}$. Define the *per-step regret* $\text{reg}_t := r_t^* - r_t$, where r_t^* is the *net reward* that would be obtained at step t if the statistics of $\{X_{i,t}\}_i$ and $\{Y_{i,t}\}_i$ were known beforehand and the optimal I_t and stopping condition were adopted. Denote the observations up to step $t-1$ as \mathcal{H}^{t-1} . Then, $\mathcal{H}^{t-1} := \cup_{\tau=1}^{t-1} \{X_{i,\tau}, Y_{i,\tau}\}_{i \in \tilde{I}_\tau}$. Without a priori statistics about $\{X_{i,t}\}$ and $\{Y_{i,t}\}$, our goal is to design an online algorithm to decide I_t based on observations obtained in previous steps \mathcal{H}^{t-1} , and \tilde{I}_t based on observed states and costs in step t , so as to minimize the *cumulative regret* $R(T) := \mathbb{E} \left[\sum_{t=1}^T \text{reg}_t \right]$.

In the following, we will first identify the structure of the optimal offline policy with a priori arm statistics, and then develop an online algorithm to learn the channel statistics and track the optimal offline policy progressively.

4 Optimal Offline Policy

We first study the optimal offline policy for the non-trivial case where $c_i > 0$. We assume that the arm statistics $\{\theta_i\}_{i=1}^K$ and $\{c_i\}_{i=1}^K$ are known to the learning agent as prior knowledge. However, the instantaneous realization of rewards and costs associated with arms are unknown to the learning agent until they are pulled. Under the assumption that the distributions of arm states and costs are i.i.d. across steps, the optimal offline policy should remain the same for different steps. Thus, in this section, we drop the step index and focus on the policy at an individual step.

According to the definition of the cascading feedback, for any ordered list, the reward in each step will not grow after the learner observes an arm with state 1. Therefore, *to maximize the net reward, the learner should stop examining the rest of the list when a state 1 is observed, in order to save the cost of examination.* Let the ordered list under the optimal offline policy be I^* , then

$$\mathbb{E}[r^*] = \max_I \sum_{i=1}^{|I|} (\theta_{I(i)} - c_{I(i)}) \prod_{j=1}^{i-1} (1 - \theta_{I(j)}). \quad (1)$$

We note that the expected net reward structure is more complex than the standard multi-armed bandits problem or the standard cascading model, and the optimal offline policy is not straightforward. By observing (1), we note that there are both a subtraction part $\theta_{I(i)} - c_{I(i)}$ and a product part $\prod_{j=1}^{i-1} (1 - \theta_{I(j)})$ inside each summation term. On one hand we should choose large θ_i to increase the value of $\theta_i - c_i$, but on the other hand we should not choose an arm with a big

gap between θ_i and c_i , because big θ_i contributes to smaller $1 - \theta_i$. In the extreme case where no cost is assigned to arms, the optimal policy is to pull all arms in *any* order, and the problem becomes trivial.

For simplicity of the analysis, we make the following assumptions:

Assumption 1 1) $\theta_i \neq c_i$, for all $i \in [K]$. 2) There exists a constant $\epsilon > 0$, such that $c_i > \epsilon$ for all $i \in [K]$.

Under Assumptions 1, we present the optimal offline policy in Theorem 1, which is called Unit Cost Ranking with Threshold 1 (UCR-T1), as it ranks the expected reward normalized by the average cost and compares against threshold one.

Theorem 1 Arrange the arm indices such that

$$\frac{\theta_{1^*}}{c_{1^*}} \geq \frac{\theta_{2^*}}{c_{2^*}} \geq \dots \geq \frac{\theta_{L^*}}{c_{L^*}} > 1 > \frac{\theta_{(L+1)^*}}{c_{(L+1)^*}} \geq \dots \geq \frac{\theta_{K^*}}{c_{K^*}}.$$

Then, $I^* = \{1^*, 2^*, \dots, L^*\}$, and the corresponding optimal per-step reward is $\mathbb{E}[r^*] = \sum_{i=1}^L (\theta_{i^*} - c_{i^*}) \prod_{j=1}^{i-1} (1 - \theta_{j^*})$.

Due to space limitation, the proof of Theorem 1 is omitted. Theorem 1 indicates that ranking the arms in a descending order of $\frac{\theta_i}{c_i}$ and only including those with $\frac{\theta_i}{c_i} > 1$ in I^* achieves a balanced tradeoff between the subtraction $\theta_i - c_i$ and the product $\prod (1 - \theta_i)$. This is an important observation which will also be useful in the online policy design.

5 Online Policy

With the optimal offline policy explicitly described in Theorem 1, in this section, we will develop an online algorithm to maximize the cumulative expected net rewards without a priori knowledge of $\{\theta_i\}_{i=1}^K$ and $\{c_i\}_{i=1}^K$.

Unlike the previous work on MAB, the net reward structure in our setting is rather complex. One difficulty is that the learner has to rank θ_i/c_i and compare it with threshold 1 for exploitation. A method to deal with this difficulty is using an UCB-type algorithm following the Optimism in Face of Uncertainty (OFU) principle [Auer *et al.*, 2002]. More specifically, we use an UCB-type indexing policy to rank the arms. Though the upper confidence bound is a biased estimation of the statistics, it will converge to the expectation asymptotically.

5.1 Algorithm and Upper Bound

The cost-aware cascading UCB (CC-UCB) algorithm is described in Algorithm 1. The costs are assumed to be random but the learning agent has no knowledge of their distributions. We use $N_{i,t}$ to track the number of steps that arm i has been pulled up to step i , and $\hat{\theta}_{i,t}$, $\hat{c}_{i,t}$ to denote the sample average of θ_i and c_i at step t , respectively. The UCB padding term on the state and cost of arm i at step t is $u_{i,t} := \sqrt{\frac{\alpha \log t}{N_{i,t}}}$, where α is a positive constant no less than 1.5.

CC-UCB adopts the OFU principle to construct an upper bound of the ratio $\frac{\theta_i}{c_i}$. The main technical difficulty and

Algorithm 1 Cost-aware Cascading UCB (CC-UCB)

 Input: ϵ, α .

```

1: Initialization: Pull all arms in  $[K]$  once, and observe
   their states and costs.
2: while  $t$  do
3:   for  $i = 1 : K$  do
4:      $U_{i,t} = \hat{\theta}_{i,t} + u_{i,t}$ ;
5:      $L_{i,t} = \max(\hat{c}_{i,t} - u_{i,t}, \epsilon)$ ;
6:     if  $U_{i,t}/L_{i,t} > 1$  then  $i \rightarrow I_t$ ;
7:   end if
8: end for
9: Rank arms in  $I_t$  in the descending order of  $\frac{U_{i,t}}{L_{i,t}}$ .
10: for  $i = 1 : |I_t|$  do
11:   Pull arm  $I_t(i)$  and observe  $X_{I_t(i),t}, Y_{I_t(i),t}$ ;
12:    $i \rightarrow \tilde{I}_t$ ;
13:   if  $X_{I_t(i),t} = 1$  then break;
14: end if
15: end for
16: Update  $N_{i,t}, \hat{\theta}_{i,t}, \hat{c}_{i,t}$  for all  $i \in \tilde{I}_t$ .
17:  $t = t + 1$ ;
18: end while
    
```

correspondingly our novel contribution, however, lies in the theoretical analysis. This is because we have to deal with two types of regret: the regret caused by pulling “bad” arms ($\theta_i < c_i$); and that caused by pulling “good” arms in a wrong order. To the authors’ best knowledge, the latter component has not been addressed in the bandit literature before, and is technically challenging. The overall regret analysis of CC-UCB is thus complicated and non-trivial.

We have the following main result for the cumulative regret upper bound of Algorithm 1.

Theorem 2 Denote $\Delta_i := c_i - \theta_i$. The accumulative regret under Algorithm 1 is upper bounded as follows:

$$R(T) \leq \sum_{i \in [K] \setminus I^*} c_i \frac{16\alpha \log T}{\Delta_i^2} + O(1),$$

where $[K] \setminus I^*$ includes all items in $[K]$ except those in I^* .

The proof of Theorem 2 is deferred to Section 5.2.

Remark: In Theorem 2, the upper bound depends on $(c_i - \theta_i)^2$, while conventional upper bounds for UCB algorithms usually depend on the gap between the ranking parameters. It is because the regret caused by pulling “good” arms in a wrong order is bounded, as shown in Section 5.2; Thus, the regret is mainly due to pulling “bad” arms, which is determined by arm i itself ($U_{i,t} > L_{i,t}$) and not related to the θ/c gap between the best arm and arm i . When c_i s are known a priori, the upper bound can be reduced by a factor of 4.

5.2 Analysis of the Upper Bound

Define $\mathcal{E}_t := \{\exists i \in [K], |\hat{\theta}_{i,t} - \theta_i| > u_{i,t} \text{ or } |\hat{c}_{i,t} - c_i| > u_{i,t}\}$, i.e. there exists an arm whose sample average of reward or cost lies outside the corresponding confidence interval. Denote $\bar{\mathcal{E}}_t$ as the complement of \mathcal{E}_t . Then, we have the following observations.

Lemma 1 If $\mathbb{1}(\bar{\mathcal{E}}_t) = 1$, then, under Algorithm 1, all arms in I^* will be included in I_t .

Proof: According to Algorithm 1, arm i will be included in I_t if $\frac{U_{i,t}}{L_{i,t}} \geq 1$. When $\mathbb{1}(\mathcal{E}_t) = 0$, we have $|\hat{\theta}_{i,t} - \theta_i| < u_{i,t}$, $|\hat{c}_{i,t} - c_i| < u_{i,t}$. Thus, $\hat{\theta}_{i,t} + u_{i,t} \geq \theta_i$, $\hat{c}_{i,t} - u_{i,t} \leq \max\{\hat{c}_{i,t} - u_{i,t}, \epsilon\} \leq c_i$, which implies that $\frac{U_{i,t}}{L_{i,t}} \geq 1$. ■

Lemma 2 Under Algorithm 1, we have $\sum_{t=1}^T \mathbb{E}[\mathbb{1}(\mathcal{E}_t)] \leq \psi := K \left(1 + \frac{4\pi^2}{3}\right)$.

Then, define $\mathcal{B}_t := \{\exists i^*, j^* \in I^*, i < j, \text{ s.t. } \frac{U_{i^*,t}}{L_{j^*,t}} < \frac{U_{j^*,t}}{L_{i^*,t}}\}$, which represents the event that arms from I^* are not ranked in the correct order. Since those arms are pulled linearly often in order to achieve small regret, intuitively, \mathcal{B}_t happens with small probability.

Lemma 3 $\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}(\bar{\mathcal{E}}_t) \mathbb{1}(\mathcal{B}_t) \right] \leq \zeta = O(1)$.

The proof of Lemma 2 is based on Hoeffding’s inequality. The proof of Lemma 3 is derived based on the observation that the arms in I^* are pulled linearly often if $\bar{\mathcal{E}}_t$ is true, and the corresponding confidence intervals $u_{i,t}$ shrinks fast in time. As $u_{i,t}$ decreases below certain threshold, \mathcal{B}_t happens. The detailed proofs of Lemma 2 and Lemma 3 are omitted due to space limitation.

Lemma 4 Consider an ordered list I_t that includes all arms from I^* with the same relative order as in I^* . Then, under Algorithm 1, $\mathbb{E}[\text{reg}_t \mid \tilde{I}_t] \leq \sum_{i \in \tilde{I}_t \setminus I^*} c_i$.

Proof: First, we point out the difference between I_t and I^* is that in I_t , there may exist arms from $[K] \setminus I^*$ that are inserted between the arms in I^* . Denoted such ordered subset of arms as $I_t \setminus I^*$. Then, depending on the realization of the states of the arm on I_t , a random subset of $I_t \setminus I^*$ will be pulled (i.e., $\tilde{I}_t \setminus I^*$), resulting in a different regret. Denote the index of the last pulled arm in $\tilde{I}_t \setminus I^*$ as \tilde{i} . Then, depending on the state of arm \tilde{i} , there are two possible cases:

i) $X_{\tilde{i},t} = 0$. For this case, the regret comes from the cost of pulling the arms in $\tilde{I}_t \setminus I^*$ only. This is because if I^* were the list of arms to pull, with the same realization of arm states, the learner would only pull the arms in $\tilde{I}_t \cap I^*$ and receive the same reward. Thus, given \tilde{I}_t and $X_{\tilde{i},t} = 0$, we have $\mathbb{E}[r_t^* - r_t \mid \tilde{I}_t, X_{\tilde{i},t} = 0] = \sum_{i \in \tilde{I}_t \setminus I^*} c_i$.

ii) $X_{\tilde{i},t} = 1$. This indicates that \tilde{i} is the last arm on \tilde{I}_t due to the stopping condition. For this case, the learner spends costs on pulling the arms in $\tilde{I}_t \setminus I^*$ but also receives the full reward one. If I^* were the list of arms to pull, with the same realization of arm states, the learner would first pull all arms in $\tilde{I}_t \cap I^*$. Since the states of such arms should be 0, she would then continue pulling the remaining arms in I^* if there is any. Denote the net reward obtained from the remaining

pullings as $r(I^* \setminus \tilde{I}_t)$. Then, $\mathbb{E}[r(I^* \setminus \tilde{I}_t)] \leq 1$. Therefore, given \tilde{I}_t and $X_{\tilde{i},t} = 1$, we have $\mathbb{E}[r_t^* - r_t | \tilde{I}_t, X_{\tilde{i},t} = 1] = \mathbb{E}[r(I^* \setminus \tilde{I}_t)] - \left(1 - \sum_{i \in \tilde{I}_t \setminus I^*} c_i\right) \leq \sum_{i \in \tilde{I}_t \setminus I^*} c_i$.

Combining both cases, we have $\mathbb{E}[\text{reg}_t | \tilde{I}_t] = \mathbb{E}[r_t^* - r_t | \tilde{I}_t] \leq \sum_{i \in \tilde{I}_t \setminus I^*} c_i$, which completes the proof. ■

Next, we consider the regret resulted from including arms outside I^* in the list I_t . We focus on the scenario when both \mathcal{E}_t and \mathcal{B}_t are false. Notice that the condition of Lemma 4 is satisfied in this scenario. Thus,

$$\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}(\bar{\mathcal{E}}_t) \mathbb{1}(\bar{\mathcal{B}}_t) \mathbb{E}[\text{reg}_t | \tilde{I}_t] \right] \leq \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}(\bar{\mathcal{E}}_t) \mathbb{1}(\bar{\mathcal{B}}_t) \left(\sum_{i \in \tilde{I}_t \setminus I^*} c_i \right) \right] \quad (2)$$

$$= \mathbb{E} \left[\sum_{i \in [K] \setminus I^*} \sum_{t=1}^T \mathbb{1}(\bar{\mathcal{E}}_t) \mathbb{1}(\bar{\mathcal{B}}_t) \mathbb{1}(i \in \tilde{I}_t) c_i \right] \quad (3)$$

$$\leq \mathbb{E} \left[\sum_{i \in [K] \setminus I^*} \sum_{t=1}^T \mathbb{1}(\bar{\mathcal{E}}_t) \mathbb{1} \left(\frac{U_{i,t}}{L_{i,t}} > 1 \right) \mathbb{1}(i \in \tilde{I}_t) c_i \right] \quad (4)$$

$$\leq \sum_{i \in [K] \setminus I^*} c_i \frac{16\alpha \log T}{\Delta_i^2}, \quad (5)$$

where (2) is based on Lemma 4; (5) is due to the fact that $\mathbb{1}(\bar{\mathcal{E}}_t) \mathbb{1} \left(\frac{U_{i,t}}{L_{i,t}} > 1 \right)$ is always less than or equal to $\mathbb{1}(\theta_i + 2u_{i,t} > c_i - 2u_{i,t})$, which is equivalent to $\mathbb{1} \left(N_{i,t} < \frac{16\alpha \log t}{\Delta_i^2} \right)$.

Denote δ^* as the largest possible per-step regret, which is bounded by $\sum_{i \in [K]} c_i$, corresponding to the worst case scenario that the learner pulls all arms but does not receive reward one. Then, combining the results from above, we have

$$\begin{aligned} R(T) &= \mathbb{E} \left[\sum_{t=1}^T [\mathbb{1}(\mathcal{E}_t) + \mathbb{1}(\bar{\mathcal{E}}_t)] \text{reg}_t \right] \\ &\leq \delta^* \sum_{t=1}^T \mathbb{E} [\mathbb{1}(\mathcal{E}_t) + \mathbb{1}(\bar{\mathcal{E}}_t) \mathbb{1}(\mathcal{B}_t)] \\ &\quad + \mathbb{E} \left[\sum_{t=1}^T \mathbb{1}(\bar{\mathcal{E}}_t) \mathbb{1}(\bar{\mathcal{B}}_t) \mathbb{E}[\text{reg}_t | \tilde{I}_t] \right] \\ &\leq \delta^* (\zeta + \psi) + \sum_{j \in [K] \setminus I^*} c_j \frac{16\alpha \log T}{\Delta_j^2}, \end{aligned}$$

which proves Theorem 2.

6 Lower Bound

Before presenting the lower bound, we first define α -consistent policies.

Definition 1 Consider online policies that sequentially pull arms in I_t until one arm with state 1 is observed. If

$\mathbb{E} \left[\sum_{t=1}^T \mathbb{1}(I_t \neq I^*) \right] = o(T^\alpha)$ for any $\alpha \in (0, 1)$, the policy is α -consistent.

Lemma 5 For any ordered list I_t , the per-step regret in step t is lower bounded by $\mathbb{E}[\text{reg}_t] \geq \mathbb{E} \left[\sum_{i \in \tilde{I}_t \setminus I^*} (c_i - \theta_i) \right]$.

Proof: Consider an ordered list $I_t^* := I_t \cap I^*$, i.e., the sub-list of I_t that only contains the arms from I^* while keeping their relative order in I_t . Denote $r_t(I_t^*)$ as the reward obtained at step t by pulling the arms in I_t^* sequentially. We have

$$\begin{aligned} \mathbb{E}[\text{reg}_t] &= \mathbb{E}[r_t^* - r_t(I_t^*) + r_t(I_t^*) - r_t] \\ &\geq \mathbb{E}[r_t(I_t^*) - r_t], \end{aligned} \quad (6)$$

where inequality (6) follows from the fact that I^* maximizes the expected reward in every step.

Similar to the proof of Lemma 4, we denote the index of the last pulled arm in $\tilde{I}_t \setminus I^*$ as \tilde{i} . Then, given \tilde{I}_t and $X_{\tilde{i},t} = 0$, we have $\mathbb{E}[r_t(I_t^*) - r_t | \tilde{I}_t, X_{\tilde{i},t} = 0] = \sum_{i \in \tilde{I}_t \setminus I^*} c_i$.

If $X_{\tilde{i},t} = 1$, based on the assumption that all policy should stop pulling in a step if a state 1 is observed, we infer that the arms in $\tilde{I}_t \cap I^*$ should have state 0. If I_t^* were the list of arms to pull, with the same realization of arm states, the learner would continue pulling the remaining arms in I_t^* if there is any. Denote the net reward obtained from the rest pulling as $r(I_t^* \setminus \tilde{I}_t)$. Then, due to the definition of I^* , we have $\mathbb{E}[r(I_t^* \setminus \tilde{I}_t)] \geq 0$. Therefore, given \tilde{I}_t and $X_{\tilde{i},t} = 1$, we have $\mathbb{E}[r_t(I_t^*) - r_t | \tilde{I}_t, X_{\tilde{i},t} = 1] = \mathbb{E}[r(I_t^* \setminus \tilde{I}_t)] - (1 - \sum_{i \in \tilde{I}_t \setminus I^*} c_i) \geq \sum_{i \in \tilde{I}_t \setminus I^*} c_i - 1$.

Combining both cases, we have $\mathbb{E}[r_t^* - r_t | \tilde{I}_t] \geq \sum_{i \in \tilde{I}_t \setminus I^*} c_i - \theta_{\tilde{i}} \geq \sum_{i \in \tilde{I}_t \setminus I^*} (c_i - \theta_i)$. Taking expectation with respect to \tilde{I}_t , we obtain the lower bound for $\mathbb{E}[\text{reg}_t]$. ■

Theorem 3 Under any α -consistent policy,

$$\liminf_{T \rightarrow \infty} \frac{R(T)}{\log T} \geq \sum_{i \in [K] \setminus I^*} \frac{c_i - \theta_i}{d(\theta_i; c_i)} \quad \text{almost surely,}$$

where $d(\theta_i; c_i)$ is the KL divergence of Bernoulli distributions with means θ_i and c_i .

Proof: According to Lemma 5, we have

$$\mathbb{E}[\text{reg}_t] \geq \mathbb{E} \left[\sum_{i \in [K] \setminus I^*} \mathbb{1}(i \in \tilde{I}_t) (c_i - \theta_i) \right]. \quad (7)$$

Therefore,

$$R(T) \geq \mathbb{E} \left[\sum_{t=1}^T \left(\sum_{i \in [K] \setminus I^*} \mathbb{1}(i \in \tilde{I}_t) (c_i - \theta_i) \right) \right] \quad (8)$$

$$= \mathbb{E} \left[\sum_{i \in [K] \setminus I^*} (c_i - \theta_i) \left(\sum_{t=1}^T \mathbb{1}(i \in \tilde{I}_t) \right) \right] \quad (9)$$

$$= \sum_{i \in [K] \setminus I^*} (c_i - \theta_i) \mathbb{E}[N_{i,T}]. \quad (10)$$

Leveraging the lower bound on $\mathbb{E}[N_{i,T}]$ from the proof of Theorem 4 in [Kveton *et al.*, 2015a], we have

$$\liminf_{T \rightarrow \infty} \frac{\mathbb{E}[N_{i,T}]}{\log T} \geq \frac{1}{d(\theta_i; c_i)}. \quad (11)$$

Combining (10) with (11), we obtain the lower bound. ■

Comparing Theorem 3 with Theorem 2, we conclude that Algorithm 1 achieves order-optimal regret performance.

7 Experiments

In this section we will resort to numerical experiments to evaluate the performances of the CC-UCB algorithm in Algorithm 1. We set $\alpha = 1.5$ and $\epsilon = 10^{-5}$. Both synthetic and real-world datasets are used.

7.1 Synthetic Data

We consider a 6-arm bandits setting with $\theta = \{0.8, 0.7, 0.6, 0.5, 0.4, 0.3\}$, and the mean of the cost $c = 0.55$ for all arms. According to the UCR-T1 policy, we have $L = 3$, i.e., the first three arms should be included in I^* . We then perform the CC-UCB algorithm under the assumption that both θ and c are unknown to the learning agent. We run it for $T = 2 \times 10^5$ steps, and average the accumulative regret over 20 runs. The result is plotted in Figure 1(a). The error bar corresponds to one standard deviation of the regrets in 20 runs. We also study the case where the mean of the cost c is known beforehand, however, the cost of each arm is still random. The result is also plotted in the same figure. As we observe, both curves increase sublinearly in T , which is consistent with the $O(\log T)$ bound we derive in Theorem 2. The regret with known cost statistics is significantly smaller than that of the unknown cost statistics case.

Next, we evaluate the impact of system parameters on the performance of the algorithm. We vary K and L , i.e., the total number of arms K , and the number of arms in I^* , respectively. We also change Δ_i , i.e., $c_i - \theta_i$, for $i \in [K] \setminus I^*$. Specifically, we set $\theta_i = 0.5$ for $i \in I^*$, $\theta_i = 0.3$ for $i \in [K] \setminus I^*$, and let c_i be a constant c across all arms. By setting Δ_{L+1} , the value of c can be determined. The cumulative regrets at $T = 10^5$ averaged over 20 runs are reported in Table 1. We observe four major trends. First, the regret increases when the number of arms K doubles. Second, the regret decreases when the number of arms in I^* (i.e., L) increases. Third, a priori knowledge of cost statistics always improve the regret. Fourth, when c is known, the regret increases as Δ_{L+1} decreases. These trends are consistent with Theorem 2. However, the dependency on Δ_{L+1} when c is unknown is not obvious from the experiment results. This might be because the algorithm depends on the estimation of the cost as well as the arm state, and the complicated interplay between cost and the optimal arm pulling policy makes the dependency on Δ_{L+1} hard to discern.

7.2 Real-world Data

In this section, we test the proposed CC-UCB algorithm using real-world data extracted from the click log dataset of Yandex Challenge [Int, 2011]. The click log contains complex user

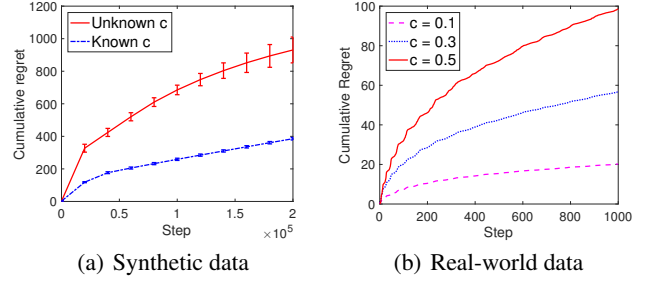


Figure 1: Cumulative regret versus step under CC-UCB.

| K | L | Δ_{L+1} | Known c | Unknown c |
|-----|-----|----------------|------------|-------------|
| 6 | 1 | 0.1 | 580.3288 | 2.2862e+03 |
| 6 | 3 | 0.1 | 352.8772 | 1.4453e+03 |
| 6 | 5 | 0.1 | 117.5846 | 364.6771 |
| 12 | 1 | 0.1 | 2.5284e+03 | 1.0225e+04 |
| 12 | 3 | 0.1 | 1.2996e+03 | 4.8120e+03 |
| 12 | 5 | 0.1 | 387.8936 | 1.3728e+03 |
| 6 | 1 | 0.05 | 1.1536e+03 | 4.7941e+03 |
| 6 | 3 | 0.05 | 697.7550 | 1.4431e+03 |
| 6 | 5 | 0.05 | 160.7688 | 212.0552 |

Table 1: T -step regret in $T = 10^5$ steps.

behaviors. To make it suitable for our algorithm, we extract data that contains click history of a specific query. We choose the total number of links for the query to be 15 and set a constant and known cost c for all of them. We estimate the probability that a user would click a link based on the dataset and use it as the ground truth θ_i . We then test the CC-UCB based on the structure of the optimal offline policy. We plot the cumulative regret in Figure 1(b) with different values of c . We observe that the cumulative regret grows sub-linearly in time, and monotonically increases as c increases, which are consistent with Theorem 2. This indicates that the CC-UCB algorithm performs very well even when some of the assumptions (such as the i.i.d. evolution of arm states) we used to derive the performance bounds do not hold.

8 Conclusions

In this paper, we studied a CCB model by taking the cost of pulling arms into the cascading bandits framework. We first explicitly characterized the optimal offline policy UCR-T1, and then developed the CC-UCB algorithm for the on-line setting. We analyzed the regret behavior of the proposed CC-UCB algorithm by analyzing two different types of error events under the algorithm. We also derived an order-matching lower bound, thus proving that the proposed CC-UCB algorithm achieves order-optimal regret. Experiments using both synthetic data and real-world data were carried out to evaluate the CC-UCB algorithm.

References

[Anantharam *et al.*, 1987] Venkatachalam Anantharam, Pravin Varaiya, and Jean Walrand. Asymptotically effi-

- cient allocation rules for the multiarmed bandit problem with multiple plays-part i: iid rewards. *IEEE Transactions on Automatic Control*, 32(11):968–976, 1987.
- [Audibert and Bubeck, 2010] Jean-Yves Audibert and Sébastien Bubeck. Best Arm Identification in Multi-Armed Bandits. In *the 23th Conference on Learning Theory*, Haifa, Israel, June 2010.
- [Auer et al., 2002] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.
- [Badanidiyuru et al., 2013] Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. Bandits with knapsacks. In *Proceedings of the IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 207–216, 2013.
- [Bubeck et al., 2009] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in multi-armed bandits problems. In *Proceedings of the 20th International Conference on Algorithmic Learning Theory*, pages 23–37, 2009.
- [Burnetas and Kanavetas, 2012] Apostolos Burnetas and Odysseas Kanavetas. Adaptive policies for sequential sampling under incomplete information and a cost constraint. *Applications of Mathematics and Informatics in Military Science*, page 97–112, 2012.
- [Burnetas et al., 2017] Apostolos Burnetas, Odysseas Kanavetas, and Michael N Katehakis. Asymptotically optimal multi-armed bandit policies under a cost constraint. *Probability in the Engineering and Informational Sciences*, 31(3):284–310, 2017.
- [Ding et al., 2013] Wenkui Ding, Tao Qiny, Xu-Dong Zhang, and Tie-Yan Liu. Multi-armed bandit with budget constraint and variable costs. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, pages 232–238, 2013.
- [Guha and Munagala, 2007] Sudipto Guha and Kamesh Munagala. Approximation algorithms for budgeted learning problems. In *Proceedings of the Thirty-ninth Annual ACM Symposium on Theory of Computing*, pages 104–113, New York, NY, USA, 2007. ACM.
- [Int, 2011] “Internet mathematics”. https://academy.yandex.ru/events/data_analysis/relpred2011/, 2011.
- [Kalathil et al., 2014] D. Kalathil, N. Nayyar, and R. Jain. Decentralized learning for multiplayer multiarmed bandits. *IEEE Transactions on Information Theory*, 60(4):2331–2345, April 2014.
- [Komiya et al., 2015] Junpei Komiya, Junya Honda, and Hiroshi Nakagawa. Optimal regret analysis of thompson sampling in stochastic multi-armed bandit problem with multiple plays. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1152–1161, Lille, France, Jul 2015.
- [Kveton et al., 2015a] Branislav Kveton, Csaba Szepesvari, Zheng Wen, and Azin Ashkan. Cascading bandits: Learning to rank in the cascade model. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 767–776, 2015.
- [Kveton et al., 2015b] Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvari. Combinatorial cascading bandits. In *Advances in Neural Information Processing Systems*, pages 1450–1458, 2015.
- [Radlinski et al., 2008] Filip Radlinski, Robert Kleinberg, and Thorsten Joachims. Learning diverse rankings with multi-armed bandits. In *In 25th Intl. Conf. on Machine Learning (ICML)*, pages 784–791, 2008.
- [Slivkins et al., 2013] Aleksandrs Slivkins, Filip Radlinski, and Sreenivas Gollapudi. Ranked bandits in metric spaces: Learning diverse rankings over large document collections. *J. Mach. Learn. Res.*, 14(1):399–436, February 2013.
- [Streeter and Golovin, 2008] Matthew Streeter and Daniel Golovin. An online algorithm for maximizing submodular functions. In *Advances in Neural Information Processing Systems 21*, pages 1577–1584. 2008.
- [Tran-Thanh et al., 2010] Long Tran-Thanh, Archie C. Chapman, Enrique Munoz de Cote, Alex Rogers, and Nicholas R. Jennings. Epsilon-first policies for budget-limited multi-armed bandits. In *AAAI*, 2010.
- [Tran-Thanh et al., 2012] Long Tran-Thanh, Archie Chapman, Alex Rogers, and Nicholas R. Jennings. Knapsack based optimal policies for budget-limited multi-armed bandits. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, pages 1134–1140, 2012.
- [Xia et al., 2015] Yingce Xia, Haifang Li, Tao Qin, Nenghai Yu, and Tie-Yan Liu. Thompson sampling for budgeted multi-armed bandits. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pages 3960–3966, 2015.
- [Xia et al., 2016a] Yingce Xia, Wenkui Ding, Xu-Dong Zhang, Nenghai Yu, and Tao Qin. Budgeted bandit problems with continuous random costs. In Geoffrey Holmes and Tie-Yan Liu, editors, *Asian Conference on Machine Learning*, volume 45, pages 317–332, Hong Kong, Nov 2016.
- [Xia et al., 2016b] Yingce Xia, Tao Qin, Weidong Ma, Nenghai Yu, and Tie-Yan Liu. Budgeted multi-armed bandits with multiple plays. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 2210–2216. AAAI Press, 2016.
- [Zong et al., 2016] Shi Zong, Hao Ni, Kenny Sung, Nan Rosemary Ke, Zheng Wen, and Branislav Kveton. Cascading bandits for large-scale recommendation problems. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, pages 835–844, 2016.