Conservative Exploration using Interleaving

Sumeet Katariya
University of Wisconsin-Madison

Branislav Kveton
Google Research

Zheng Wen
Adobe Research

Vamsi Krishna Potluru
Comcast AI Labs

Abstract

In many practical problems, a learning agent may want to learn the best action in hindsight without ever taking a bad action, which is much worse than a default production action. In general, this is impossible because the agent has to explore unknown actions, some of which can be bad, to learn better actions. However, when the actions are structured, this is possible if the unknown action can be evaluated by interleaving it with the default action. We formalize this concept as learning in stochastic combinatorial semi-bandits with exchangeable actions. We design efficient learning algorithms for this problem, bound their n-step regret, and evaluate them on both synthetic and real-world problems. Our real-world experiments show that our algorithms can learn to recommend K most attractive movies without ever making disastrous recommendations, both overall and subject to a diversity constraint.

1 Introduction

Recommender systems are an integral component of many industries, with applications in content personalization, advertising, and page design (Resnick and Varian, 1997; Adomavicius and Tuzhilin, 2015; Broder, 2008). Multi-armed bandit algorithms provide adaptive techniques for content recommendation. However, although they are theoretically well-understood, they have not been widely adopted in production systems (Cremonesi et al., 2011; Schnabel et al., 2018). This is primarily due to concerns that the output of the bandit algorithm can be suboptimal or even disastrous, especially when the algorithm explores suboptimal arms. To address this issue, most industries have a default recommendation engine in production that has been well-optimized and tested for many years, and a promising new policy is often evaluated using A/B testing (Siroker and Koomen, 2013), which allocates a small α fraction of the traffic to the new policy. When the utilities of actions are independent, this is a reasonable solution that allows the new policy to be evaluated conservatively.

Many recommendation problems involve *structured actions*, such as sets of recommended movies. In these problems, the total utility of the action can be decomposed into the utilities of individual items in it, such as movies. Therefore, it is conceivable that the new policy could be evaluated in a controlled and principled fashion by *interleaving* items in the new and default actions, instead of dividing the traffic as in A/B testing. As a concrete example, consider the problem of recommending top-K movies to a new visitor (Deshpande and Karypis, 2004). A company may have a default policy that recommends a fixed set of K movies that performs well, but intends to test a new algorithm that promises to learn better movies. The A/B testing method would show the recommendations of the new algorithm to a visitor with probability α. In the initial stages, the new algorithm is expected to explore a lot to learn, and may hurt engagement with the visitor who is shown a disastrous set of movies, just to learn that these movies are not good. An arguably better approach, which does not hurt any visitor's engagement as much and gathers the same feedback on average, is to show the default well-tested movies interleaved with α fraction of new recommendations. A recent study by Schnabel et al. (2018) concluded that this latter approach is in fact better,

> "These findings indicate that for improving recommendation systems in practice, it is preferable to mix a limited amount of exploration into every impression – as opposed to having a few impressions that do pure exploration."

In this paper, we formalize the above idea and study the general case where actions are *exchangeable*, which is a mathematical formulation of the notion of interleaving. In particular, we study learning variants of maximizing an unknown linear function on an exchangeable action set subject to a conservative constraint.

In our motivating recommendation example, we require that any recommendation is always above a certain baseline qual-

Proceedings of the 22$^{\text{nd}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan. PMLR: Volume 89. Copyright 2019 by the author(s).

ity. The question that we want to answer is *what is the price for being this conservative*? In this work, we answer this question and make five contributions. First, we introduce the idea of *conservative multi-armed bandits in combinatorial action spaces*, and formulate a conservative constraint that addresses the issues raised in Schnabel et al. (2018). Existing conservative constraints for multi-armed bandit problems do not address this issue, as discussed in Section 6. Second, we propose interleaving as a solution, and show how it naturally leads to the idea of *exchangeable* action spaces. We precisely formulate *conservative interleaving bandits*, a constrained online learning problem in exchangeable action spaces. Third, we present *Interleaving Upper Confidence Bound (*`iUCB`*)*, a computationally and sample-efficient algorithm for solving our problem. The algorithm satisfies our conservative constraint by design. Fourth, we prove gap-dependent upper bounds on its expected n-step regret. The bounds are logarithmic in the number of steps n, linear in the number of items L, and increase with the level of conservatism. Finally, we evaluate `iUCB` on both synthetic and real-world problems. In synthetic experiments, we validate an extra factor in our regret bounds, which is the price for being conservative. In real-world experiments, we formulate and solve two top-K recommendation problems. To the best of our knowledge, this is the first work that studies conservatism in combinatorial bandit problems.

2 Setting

We formulate our online learning problem as a stochastic combinatorial semi-bandit (Kveton et al., 2015b; Gai et al., 2012; Chen et al., 2013), which we review in Section 2.1. In Section 2.2, we define our notion of conservativeness. In Section 2.3, we suggest interleaving as a solution and formulate it mathematically using the notion of exchangeable action spaces. Finally, in Section 2.4, we introduce our online learning problem of *conservative interleaving bandits*. To simplify exposition, we write all random variables in bold. We denote $\{1, \dots, K\}$ by $[K]$.

2.1 Stochastic Combinatorial Semi-Bandits

A *stochastic combinatorial semi-bandit* (Gai et al., 2012; Chen et al., 2013; Kveton et al., 2015b) is a tuple $(E, \mathcal{B}, P)$, where $E = [L]$ is a finite set of L items; $\mathcal{B} \subseteq \Pi_K(E)$ is a set of feasible actions, which is a subset of all sets of size K from E, $\Pi_K(E)$; and P is a probability distribution over a unit cube $[0, 1]^E$.

The learning agent interacts with this problem as follows. Let $(\boldsymbol{w}_t)_{t=1}^n$ be a sequence of n i.i.d. weights drawn from P, where $\boldsymbol{w}_t(e)$ is the weight of item $e \in E$ at time t. At time t, the agent takes action $\boldsymbol{A}_t \in \mathcal{B}$, which is a set of K items from E. The reward for taking the action is $f(\boldsymbol{A}_t, \boldsymbol{w}_t)$, where $f(A, w) = \sum_{e \in A} w(e)$ is the sum of the weights of all items in A. After taking action $\boldsymbol{A}_t$, the agent observes the weight $\boldsymbol{w}_t(e)$ of each item $e \in \boldsymbol{A}_t$.

The expected weights of items are defined as $\bar{w} = \mathbb{E}[\boldsymbol{w}]$. The learning agent is evaluated by its *expected n-step regret* $R(n) = \sum_{t=1}^n \mathbb{E}[f(A_*, \bar{w})] - \sum_{t=1}^n \mathbb{E}[f(\boldsymbol{A}_t, \bar{w}])$, where $A_* = \arg\max_{A \in \mathcal{B}} f(A, \bar{w})$ is the *best action in hindsight*.

Stochastic combinatorial semi-bandits can be used to model top-K recommendation problems as follows. The ground set E is the set of all items that can be recommended, such as movies. The action $A \in \mathcal{B}$ is any set of K movies that can be recommended jointly to the user. The weight of item e at time t, $\boldsymbol{w}_t(e)$, is an indicator of the click on item e at time t. This interaction model is known as the *document click model* (Chuklin et al., 2015).

2.2 Conservative Constraint

To avoid disastrous actions, which may contain a large number of bad items, we impose a constraint on the actions of the learning agent. This constraint is stated formally below.

Let K denote the number of items in all actions. Let B_0 be the *default baseline action*. Our constraint requires that at any time t, the action $\boldsymbol{A}_t$ of the learning agent should be comparable to or better than the baseline action B_0, in the sense that most items in $\boldsymbol{A}_t$ should be at least as good as those in B_0. More formally, we require that there exists a bijection $\rho_{\boldsymbol{A}_t, B_0} : \boldsymbol{A}_t \to B_0$ such that

$$\sum_{e \in \boldsymbol{A}_t} \mathbb{1}(\bar{w}(e) \geq \bar{w}(\rho_{\boldsymbol{A}_t, B_0}(e))) \geq (1 - \alpha)K \qquad (1)$$

holds with a high probability at any time $t \in [n]$, where α is a problem-specific *risk tolerance parameter*. In other words, the items in $\boldsymbol{A}_t$ and B_0 can be matched such that at most α fraction of the items in $\boldsymbol{A}_t$ has a lower expected reward than the matched items in B_0. We compare (1) to other notions of conservatism in the literature in Section 6.

2.3 Exchangeable Actions

Given an algorithm that explores and suggests new actions that could potentially be disastrous, a natural way to satisfy (1) is to *interleave* most items from the default action B_0 with a few items from the new action. This is possible if the set of feasible actions $\mathcal{B} \subseteq \Pi_K(E)$ is *exchangeable*.

Definition 1 (Exchangeable set)**.** *Given a set E, a set $\mathcal{B} \subseteq \Pi_K(E)$ is exchangeable if for any two actions $A_1, A_2 \in \mathcal{B}$, there exists a bijection $\rho_{A_1, A_2} : A_1 \to A_2$ such that*

$$\forall\, G \subseteq A_1 : A_1 \setminus G \cup \{\rho_{A_1, A_2}(e) : e \in G\} \in \mathcal{B}\,. \qquad (2)$$

From now on, we assume that all sets of feasible actions $\mathcal{B}$ are exchangeable. We give examples of two exchangeable sets below.

Our first example are top-K movie recommendations from Section 1. In this example, E is the set of movies and the exchangeable set $\mathcal{B}$ are all subsets of size K from E. The bijection ρ_{A_1,A_2} between two actions $A_1, A_2 \in \mathcal{B}$ can be any bijection subject to the constraint that common items in A_1 and A_2 are mapped to each other. Formally, ρ_{A_1,A_2} is any bijection $A_1 \to A_2$ such that $\rho_{A_1,A_2}(e) = e$ for any $e \in A_1 \cap A_2$. The set $\mathcal{B}$ in this example is also known as a *uniform matroid* of rank K.

Our second example are diverse movie recommendations. Let E be the set of movies and $\mathcal{P}_1, \dots, \mathcal{P}_K$ be a partition of E, where each $\mathcal{P}_i$ represents a movie genre. Then we define the exchangeable set as

$$\mathcal{B} = \{A \in \Pi_K(E) : a_1 \in \mathcal{P}_1, \dots, a_K \in \mathcal{P}_K\}, \qquad (3)$$

where $A = \{a_1, \dots, a_K\}$. Based on the above definition, any action $A \in \mathcal{B}$ contains one movie from each genre, and hence is diverse. The bijection ρ_{A_1,A_2} between two actions $A_1, A_2 \in \mathcal{B}$ maps $e \in A_1 \cap \mathcal{P}_i$ to $e' \in A_2 \cap \mathcal{P}_i$ for all $i \in [K]$. The set $\mathcal{B}$ in this example is known as a *partition matroid* of rank K.

We briefly explain how exchangeability leads to interleaving of items and allows conservative exploration. In both movie recommendation examples, we can set A_1 to be the default baseline action and A_2 to be a newly evaluated action. A natural approach to exploring A_2 without violating the conservative constraint in (1) is through interleaving, all items in the new action A_2 are explored in $S = 1/\alpha$ steps by taking S interleaved actions. Each *interleaved action* substitutes αK unique items in A_1 for the matched items in A_2. Any such action is feasible by Definition 1.

For simplicity of exposition, we make two assumptions on α. First, $1/K \leq \alpha \leq 1/2$. This boundary condition says that we do not consider extreme non-conservative cases, where the learning agent can explore more than a half of items in a new action A_2; and extreme conservative cases, where the learning agent cannot explore safely at least one item in A_2. Second, we assume that $\alpha K \in \mathbb{N}$. This means that all items in A_2 can be observed once in exactly $S = 1/\alpha$ interleaved actions. If this latter assumption is violated, we suggest that α is set to the maximum value of $\alpha' < \alpha$ that satisfies the assumption. This setting is clearly more conservative and satisfies both of our assumptions.

2.4 Conservative Interleaving Bandits

A *conservative interleaving bandit* is variant of a stochastic combinatorial semi-bandit (Section 2.1) for conservative exploration. Formally, it is a tuple $(E, \mathcal{B}, P, B_0, \alpha)$, where E, $\mathcal{B}$, and P are defined as in Section 2.1; $\mathcal{B}$ is an exchangeable set (Definition 1), $B_0 \in \mathcal{B}$ is a default baseline action, and $\alpha \in [0, 1]$ is the risk tolerance parameter in (1). We assume that the learning agent knows E, $\mathcal{B}$, B_0, and α; and that the distribution P is unknown.

3 Algorithm

Learning in conservative interleaving bandits is non-trivial. For instance, we cannot simply take optimistic actions of existing non-conservative algorithms for combinatorial semi-bandits (Kveton et al., 2014; Talebi and Proutiere, 2016) and interleave them with $1 - \alpha$ fraction of items from the default baseline action B_0. The regret of this policy would be linear because its actions never converge to the optimal action A^*, unless all items in B_0 are optimal. If this was the case, we would not have a learning problem to start with.

In this section, we introduce our *Interleaving Upper Confidence Bound (*`iUCB`*)* algorithm, which achieves sublinear regret by continuously improving the default baseline action B_0 with a high probability. We present two variants of the algorithm, `iUCB1` and `iUCB2`. In `iUCB1`, the agent knows the expected rewards of all items in B_0, $\{\bar{w}(e) : e \in B_0\}$. In practice, these rewards could be known if the baseline policy B_0 was executed before. In `iUCB1`, the agent does not know the expected rewards of items in B_0. We refer to the common aspects of both algorithms as `iUCB`.

The pseudocode of both algorithms is in Algorithm 1. We highlight their differences in comments. Recall that K is the number of items in all actions. `iUCB` operates in rounds, which are indexed by t, and takes S interleaved actions in each round. We assume that `iUCB` has access to an oracle OPT that returns the most rewarding action for any weight vector $w \in [0, 1]^E$. When $\mathcal{B}$ are bases of a *matroid*, as in our examples in Section 2.3, OPT is a greedy algorithm for finding the maximum weight basis of a matroid and can be implemented to run in $O(L \log L)$ time (Edmonds, 1971).

In each round, `iUCB` has three stages. In the first stage (lines 9–10), `iUCB` computes high-probability *upper confidence bounds (UCBs)* $\boldsymbol{U}_t \in (\mathbb{R}^+)^E$ and *lower confidence bounds (LCBs)* $\boldsymbol{L}_t \in (\mathbb{R}^+)^E$ on the expected rewards of all items. For any item $e \in E$,

$$\begin{aligned} \boldsymbol{U}_t(e) &= \hat{\boldsymbol{w}}_{\boldsymbol{T}_{t-1}(e)}(e) + c_{n,\boldsymbol{T}_{t-1}(e)}\,, \\ \boldsymbol{L}_t(e) &= \max\{\hat{\boldsymbol{w}}_{\boldsymbol{T}_{t-1}(e)}(e) - c_{n,\boldsymbol{T}_{t-1}(e)}, 0\}\,, \end{aligned} \qquad (4)$$

where $\hat{\boldsymbol{w}}_s(e)$ is the average of the first s observed weights of item e, $\boldsymbol{T}_t(e)$ is the number of times that item e is observed in the first t steps, and

$$c_{n,s} = \sqrt{1.5 \log(n)/s} \qquad (5)$$

is the width of a high-probability confidence interval around $\hat{\boldsymbol{w}}_s(e)$, such that $\bar{w}(e) \in [\hat{\boldsymbol{w}}_s(e) - c_{n,s}, \hat{\boldsymbol{w}}_s(e) + c_{n,s}]$ holds with a high probability. Note that this is a `UCB1` confidence interval (Auer et al., 2002). It is trivial to modify our algorithm to use tighter `KL-UCB` confidence intervals (Garivier and Cappé, 2011). However, our analysis in Section 4 does not generalize straightforwardly to this setting.

In line 12, `iUCB` chooses *decision set* $\boldsymbol{D}_t$, which is the optimal action with respect to weights $\boldsymbol{U}_t$, an optimistic estimate

Algorithm 1 `iUCB` for conservative interleaving bandits.

```
1:  Input: Baseline action $B_0 \in \mathcal{B}$, risk tolerance $\alpha$
2:
3:  $S \leftarrow 1/\alpha \in \mathbb{N}$
4:  Observe $\boldsymbol{w}_0 \sim P$
5:  $\forall e \in E : \boldsymbol{T}_0(e) \leftarrow 1,\ \hat{\boldsymbol{w}}_1(e) \leftarrow \boldsymbol{w}_0(e)$
6:
7:  for $t = 1, 2, \dots$ do
8:    for all $e \in E$ do {      // Compute UCBs and LCBs}
9:      $\boldsymbol{U}_t(e) = \hat{\boldsymbol{w}}_{\boldsymbol{T}_{t-1}(e)}(e) + c_{n,\boldsymbol{T}_{t-1}(e)}$
10:     $\boldsymbol{L}_t(e) = \max\{\hat{\boldsymbol{w}}_{\boldsymbol{T}_{t-1}(e)}(e) - c_{n,\boldsymbol{T}_{t-1}(e)}, 0\}$
11:
12:   $\boldsymbol{D}_t \leftarrow \mathsf{OPT}(\boldsymbol{U}_t)$              // Compute decision set
13:   for all $e \in B_0$ do {           // Compute baseline set}
14:     if $\bar{w}(e)$ is known then {                  // iUCB1}
15:       $\boldsymbol{v}_t(e) \leftarrow \bar{w}(e)$
16:     else {                                           // iUCB2}
17:       $\boldsymbol{v}_t(e) \leftarrow \boldsymbol{U}_t(e)$
18:   for all $e \in E \setminus B_0$ do
19:     $\boldsymbol{v}_t(e) \leftarrow \boldsymbol{L}_t(e)$
20:   $\boldsymbol{B}_t \leftarrow \mathsf{OPT}(\boldsymbol{v}_t)$
21:
22:   // Take $S$ combined actions and update statistics
23:   Let $\{\boldsymbol{B}_t^s\}_{s=1}^S$ be a partition of $\boldsymbol{B}_t$ such that $|\boldsymbol{B}_t^s| =$
      $\alpha K$ for all $s \in [S]$
24:   Let $\boldsymbol{\rho}_t : \boldsymbol{B}_t \to \boldsymbol{D}_t$ be the bijection in Definition 1
25:   $\forall e \in E : \boldsymbol{T}_t(e) \leftarrow \boldsymbol{T}_{t-1}(e)$
26:   for $s = 1, \dots, S$ do
27:     Take action $\boldsymbol{A}_t = \boldsymbol{B}_t \setminus \boldsymbol{B}_t^s \cup \{\boldsymbol{\rho}_t(e) : e \in \boldsymbol{B}_t^s\}$
28:     Observe $\{\boldsymbol{w}_t(e) : e \in \boldsymbol{A}_t\}$, where $\boldsymbol{w}_t \sim P$
29:     for all $e \in \boldsymbol{A}_t$ do
30:       $\hat{\boldsymbol{w}}_{\boldsymbol{T}_t(e)+1}(e) \leftarrow \dfrac{\boldsymbol{T}_t(e)\hat{\boldsymbol{w}}_{\boldsymbol{T}_t(e)}(e) + \boldsymbol{w}_t(e)}{\boldsymbol{T}_t(e) + 1}$
31:       $\boldsymbol{T}_t(e) \leftarrow \boldsymbol{T}_t(e) + 1$
```

of $\bar{w}$. The same approach was used in *Optimistic Matroid Maximization (*`OMM`*)* of Kveton et al. (2014). However, unlike `OMM`, `iUCB` cannot take $\boldsymbol{D}_t$ because it may not satisfy our conservative constraint in (1). We refer to $\boldsymbol{D}_t$ as a *set* to distinguish it from the actions of `iUCB`.

In the second stage (lines 13–20), `iUCB` computes *baseline set* $\boldsymbol{B}_t$, which is the optimal action with respect to weights $\boldsymbol{v}_t$. We refer to $\boldsymbol{B}_t$ as a *set* to distinguish it from the actions of `iUCB`. The weights $\boldsymbol{v}_t$ are set as follows. If $e \in B_0$, we set $\boldsymbol{v}_t(e) = \bar{w}(e)$ if $\bar{w}(e)$ is known, and set $\boldsymbol{v}_t(e) = \boldsymbol{U}_t(e)$ when it is not. If $e \in E \setminus B_0$, we set $\boldsymbol{v}_t(e) = \boldsymbol{L}_t(e)$. This setting guarantees that if any item $e \in E \setminus B_0$ is chosen to $\boldsymbol{B}_t$ over any item $e' \in B_0$, its expected reward is higher than that of item e' with a high probability. As a result, the baseline is improved.

In the last stage (lines 22–31), `iUCB` takes $S = 1/\alpha$ combined actions of $\boldsymbol{D}_t$ and $\boldsymbol{B}_t$, which are guaranteed to be in $\mathcal{B}$ by Definition 1. In particular, let $\boldsymbol{\rho}_t : \boldsymbol{B}_t \to \boldsymbol{D}_t$ be the bijection in Definition 1 and $\{\boldsymbol{B}_t^s\}_{s=1}^S$ be a partition of $\boldsymbol{B}_t$ into S sets such that $|\boldsymbol{B}_t^s| = \alpha K$ for all $s \in [S]$. Then we take actions $\boldsymbol{A}_t = \boldsymbol{B}_t \setminus \boldsymbol{B}_t^s \cup \{\boldsymbol{\rho}_t(e) : e \in \boldsymbol{B}_t^s\}$ for $s \in [S]$ sequentially. Since $\boldsymbol{A}_t$ contains at least $(1 - \alpha)K$ baseline items, all of which improve over their matched items in B_0 with a high probability, the conservative constraint in (1) is satisfied.

After each action, `iUCB` updates its sufficient statistics (lines 29–31), which are used to estimate the UCBs and LCBs in the next round.

4 Analysis

This section has three subsections. In Section 4.1, we prove that `iUCB1` is conservative and bound its regret. The main challenge in our analysis is that we cannot directly apply a UCB-like argument, because the baseline set $\mathcal{B}_t$ is chosen based on lower confidence bounds. In Section 4.2, we prove analogous claims for `iUCB2`. In Section 4.3, we discuss our theoretical results.

We adopt the following conventions in our analysis. Without loss of generality, we assume that items in E are ordered such that $\bar{w}(1) \geq \dots \geq \bar{w}(L)$. The optimal action is A^*, the decision set at time t is $\boldsymbol{D}_t$, and the baseline set at time t is $\boldsymbol{B}_t$. Note that A^*, $\boldsymbol{D}_t$, and $\boldsymbol{B}_t$ belong to exchangeable action set $\mathcal{B}$, which is defined in Definition 1. At any time t, let $\boldsymbol{\pi}_t : A^* \to \boldsymbol{D}_t$ and $\boldsymbol{\sigma}_t : \boldsymbol{D}_t \to \boldsymbol{B}_t$ be the bijections in Definition 1, which are guaranteed to exist. The bijections simplify our analysis, and allow us to decompose the improvements in $\boldsymbol{D}_t$ and $\boldsymbol{B}_t$ into items in them.

For any items e and e' such that $\bar{w}(e') \geq \bar{w}(e)$, we define the *gap* as $\Delta_{e,e'} = \bar{w}(e') - \bar{w}(e)$. We also define a "good" event at time t as

$$\mathcal{E}_t = \{\forall\, e \in E : |\bar{w}(e) - \hat{\boldsymbol{w}}_{\boldsymbol{T}_{t-1}(e)}(e)| \leq c_{n,\boldsymbol{T}_{t-1}(e)}\}\,, \quad (6)$$

which is the event that $\bar{w}(e)$ is in the high-probability confidence interval around $\hat{\boldsymbol{w}}_{\boldsymbol{T}_{t-1}(e)}(e)$ for all items e at the beginning of time t.

4.1 `iUCB1`: Known Baseline Mean Rewards

First, we show that `iUCB1` is conservative. The proof of this claim is in Appendix.

Theorem 1. `iUCB1` *satisfies* (1) *jointly at all times* $t \in [n]$ *with probability of at least* $1 - 2L/(Sn)$.

Then we prove a gap-dependent upper bound on the regret of `iUCB1`. The bound involves two kinds of gaps. For any suboptimal item e, we define its gap from the closest better optimal item,

$$\Delta_{e,\min} = \min\nolimits_{e^* \in A^* : \Delta_{e,e^*} > 0} \Delta_{e,e^*}\,. \quad (7)$$

In addition, for any optimal item e^*, we define its gap from the closest worse suboptimal item,

$$\Delta^*_{e^*,\min} = \min_{e \in E \setminus A^* : \Delta_{e,e^*} > 0} \Delta_{e,e^*}\,. \tag{8}$$

Our regret bound is stated below.

Theorem 2 (Regret of iUCB1)**.** *The expected n-step regret of* iUCB1 *is bounded as*

$$(S-1)\left(\sum_{e \in E \setminus A^*} \frac{24}{\Delta_{e,\min}} + \sum_{e^* \in A^*} \frac{12}{\Delta^*_{e^*,\min}}\right) \log n + \sum_{e \in E \setminus A^*} \frac{12}{\Delta_{e,\min}} \log n + c\,,$$

where $S = 1/\alpha$; $\Delta_{e,\min}$ and $\Delta^*_{e^*,\min}$ are defined in (7) and (8), respectively; and $c = O(SL\sqrt{\log n})$.

Proof. Let $\bar{\mathcal{E}} = \bigcup_{t=1}^{n/S} \bar{\mathcal{E}}_t$ be the event that at least one event $\mathcal{E}_t$ in (6) does not occur; and $\mathcal{E}$ be its complement, the event that all events $\mathcal{E}_t$ in (6) occur. Let $\boldsymbol{R}_t$ the stochastic regret at time t.

We decompose the expected n-step regret into those under events $\mathcal{E}$ and $\bar{\mathcal{E}}$ as

$$R(n) = \mathbb{E}\left[\mathbb{1}\left(\bar{\mathcal{E}}\right) \sum_{t=1}^{n/S} \boldsymbol{R}_t\right] + \mathbb{E}\left[\mathbb{1}\left(\mathcal{E}\right) \sum_{t=1}^{n/S} \boldsymbol{R}_t\right]. \tag{9}$$

The regret due to the first term in (9) is low. In particular, since $P(\bar{\mathcal{E}}) \leq 2LS^{-1}n^{-1}$ (Lemma 1 in Appendix) and the maximum n-step regret is Kn, the maximum contribution due to the first term is $2LK/S$.

In the rest of the proof, we analyze the second term in (9) under event $\mathcal{E}$. The key observation is that the expected regret at time t decomposes as

$$\begin{aligned} \mathbb{E}[\boldsymbol{R}_t] &= S \sum_{e^* \in A^*} \bar{w}(e^*) - (S-1) \sum_{e' \in \boldsymbol{B}_t} \bar{w}(e') - \sum_{e \in \boldsymbol{D}_t} \bar{w}(e) \\ &= S\left(\sum_{e^* \in A^*} \bar{w}(e^*) - \sum_{e \in \boldsymbol{D}_t} \bar{w}(e)\right) + \\ &\quad (S-1)\left(\sum_{e \in \boldsymbol{D}_t} \bar{w}(e) - \sum_{e' \in \boldsymbol{B}_t} \bar{w}(e')\right), \end{aligned} \tag{10}$$

where the first term reflects the regret due to the decision set and the second term reflects the regret due to interleaving with the baseline set.

The first term in (10) can be bounded as follows. Since the decision set $\boldsymbol{D}_t$ is chosen optimistically, the UCBs of items in $\boldsymbol{D}_t$ are at least as high as those of the matched items in A^*. Thus, we have that

$$\sum_{e^* \in A^*} \bar{w}(e^*) - \sum_{e \in \boldsymbol{D}_t} \bar{w}(e) \leq \sum_{e \in \boldsymbol{D}_t \setminus A^*} 2c_{n,\boldsymbol{T}_{t-1}(e)} \tag{11}$$

at any time t under event $\mathcal{E}$, by Lemma 3 in Appendix. Now we sum up the above bound for all t and get

$$\begin{aligned} &\sum_{t=1}^{n/S} \sum_{e \in \boldsymbol{D}_t \setminus A^*} 2c_{n,\boldsymbol{T}_{t-1}(e)} \\ &\quad \leq \sum_{e \in E \setminus A^*} \sum_{t=1}^{n/S} \sqrt{\frac{6 \log n}{\boldsymbol{T}_{t-1}(e)}} \mathbb{1}(e \in \boldsymbol{D}_t) \\ &\quad \leq \sum_{e \in E \setminus A^*} \sqrt{6 \log n} \left(1 + 2\sqrt{\frac{6 \log n}{\Delta^2_{e,\min}}}\right) \\ &\quad = \sum_{e \in E \setminus A^*} \frac{12}{\Delta_{e,\min}} \log n + L\sqrt{6 \log n}\,. \end{aligned} \tag{12}$$

The first inequality is from the definition of our confidence intervals. The second inequality is from two observations. First, when item e is chosen, the counter $\boldsymbol{T}_t(e)$ increases by one. Second, this event occurs at most $m = 6\Delta_{e,\min}^{-2} \log n$ times (Lemma 4 in Appendix A.1). Finally, we apply

$$\sum_{s=1}^{m} \frac{1}{\sqrt{s}} \leq 1 + 2\sqrt{m}\,. \tag{13}$$

The last equality is an algebraic manipulation.

The second term in (10) is bounded as follows. Since the baseline set $\boldsymbol{B}_t$ is chosen based on LCBs, the LCBs of items in $\boldsymbol{B}_t$ are at least as high as those of the matched items in $\boldsymbol{D}_t$. Thus, we have that

$$\sum_{e \in \boldsymbol{D}_t} \bar{w}(e) - \sum_{e' \in \boldsymbol{B}_t} \bar{w}(e') \leq \sum_{e \in \boldsymbol{D}_t \setminus \boldsymbol{B}_t} 2c_{n,\boldsymbol{T}_{t-1}(e)} \tag{14}$$

at any time t under event $\mathcal{E}$, by Lemma 5 in Appendix. Now we sum up the above bound for all t and get

$$\begin{aligned} &\sum_{t=1}^{n/S} \sum_{e \in \boldsymbol{D}_t \setminus \boldsymbol{B}_t} 2c_{n,\boldsymbol{T}_{t-1}(e)} \leq \\ &\quad \sum_{e \in E \setminus A^*} \sum_{t=1}^{n/S} \sqrt{\frac{6 \log n}{\boldsymbol{T}_{t-1}(e)}} \mathbb{1}(e \in \boldsymbol{D}_t) + \\ &\quad \sum_{e^* \in A^*} \sum_{t=1}^{n/S} \sqrt{\frac{6 \log n}{\boldsymbol{T}_{t-1}(e^*)}} \mathbb{1}(e^* \in \boldsymbol{D}_t \setminus \boldsymbol{B}_t)\,, \end{aligned} \tag{15}$$

where the inequality is from the definition of our confidence intervals.

The first term in (15) is bounded as in (12). The second term is bounded similarly, where the only difference is in the definition of the gap. In particular, if an optimal item e^* is chosen $\Omega((\Delta^*_{e^*,\min})^{-2} \log n)$ times (Lemma 6 in Appendix), it must be in the baseline set $\boldsymbol{B}_t$ and the corresponding regret is zero. Therefore, the regret due to both terms in (15) is bounded from above by

$$\sum_{e \in E \setminus A^*} \frac{12}{\Delta_{e,\min}} \log n + L\sqrt{6 \log n} + \sum_{e^* \in A^*} \frac{12}{\Delta^*_{e^*,\min}} \log n + L\sqrt{6 \log n}\,. \tag{16}$$

Finally, we add S times the upper bound in (12) and $S - 1$ times the upper bound in (16), and get our claim. □

4.2 iUCB2: Unknown Baseline Mean Rewards

First, we show that iUCB2 is conservative. The proof of this claim is in Appendix.

Theorem 3. iUCB2 *satisfies* (1) *jointly at all times* $t \in [n]$ *with probability of at least* $1 - 2L/(Sn)$.

Now we prove a gap-dependent upper bound on the regret of iUCB2.

Theorem 4 (Regret of iUCB2)**.** *The expected* n*-step regret of* iUCB2 *is bounded as*

$$(S-1)\left(\sum_{e \in E \setminus A^*} \frac{48}{\Delta_{e,\min}} + \sum_{e^* \in A^*} \frac{36}{\Delta^*_{e^*,\min}}\right) \log n + \sum_{e \in E \setminus A^*} \frac{12}{\Delta_{e,\min}} \log n + c\,,$$

where $S = 1/\alpha$*;* $\Delta_{e,\min}$ *and* $\Delta^*_{e^*,\min}$ *are defined in* (7) *and* (8)*, respectively; and* $c = O(SL\sqrt{\log n})$*.*

Proof. The proof is similar to that of Theorem 2. The only major difference is that items in the default baseline action B_0 are chosen to $\boldsymbol{B}_t$ based on their UCBs, while the other items are selected based on their LCBs.

The regret at time t decomposes as in (10), and the first term in (10) is bounded exactly as in (12). To bound the second term, we decompose the regret based on whether the item in $\boldsymbol{B}_t$ is in B_0 or not, and get that

$$\begin{aligned}
&\sum_{e \in \boldsymbol{D}_t} \bar{w}(e) - \sum_{e' \in \boldsymbol{B}_t} \bar{w}(e') \\
&= \sum_{e \in \boldsymbol{D}_t : \boldsymbol{\sigma}_t(e) \notin B_0} \bar{w}(e) - \sum_{e' \in \boldsymbol{B}_t \setminus B_0} \bar{w}(e') + \sum_{e \in \boldsymbol{D}_t : \boldsymbol{\sigma}_t(e) \in B_0} \bar{w}(e) - \sum_{e' \in \boldsymbol{B}_t \cap B_0} \bar{w}(e') \\
&\leq \sum_{e \in \boldsymbol{D}_t : \boldsymbol{\sigma}_t(e) \notin B_0} 2c_{n,\boldsymbol{T}_{t-1}(e)} + \sum_{e \in \boldsymbol{D}_t : \boldsymbol{\sigma}_t(e) \in B_0} 4c_{n,\boldsymbol{T}_{t-1}(e)}\,,
\end{aligned}$$

where $\boldsymbol{\sigma}_t(e)$ is the matched item in $\boldsymbol{B}_t$ to item e in $\boldsymbol{D}_t$. The last step follows from two observations. When $\boldsymbol{\sigma}_t(e) \notin B_0$, we follow the same proof as in (14) and get the same upper bound as in (16). When $\boldsymbol{\sigma}_t(e) \in B_0$, we apply Lemma 7 in Appendix A.2. This lemma relies on the observation that any item in $\boldsymbol{B}_t \cap B_0$ is chosen at least as often as its matched item in $\boldsymbol{D}_t$ up to any time t, which holds for any $\alpha \leq 1/2$. The final upper bound is the same as in (16), except that all terms are multiplied by 2.

Finally, we add up the contributions of all terms, which is S times the upper bound in (12) and $3(S-1)$ times the upper bound in (16), and get our claim. □

4.3 Discussion

Our regret bounds in Theorems 2 and 4 depend on two gaps. The first gap, $\Delta_{e,\min}$ in (7), measures the distance of suboptimal item e from the closest better optimal item. This gap is standard in stochastic combinatorial semi-bandits with matroid constraints (Kveton et al., 2014), which we refer to as *matroid bandits*. Matroid constraints are a weaker notion of exchangeability than that in this paper. The second gap, $\Delta^*_{e^*,\min}$ in (8), measures the distance of optimal item e^* from the closest worse suboptimal item. Similar gaps appear in top-K best-arm identification problems (Kalyanakrishnan et al., 2012). If we let

$$\Delta = \min\{\min_{e \in E \setminus A^*} \Delta_{e,\min}, \min_{e^* \in A^*} \Delta^*_{e^*,\min}\}\,,$$

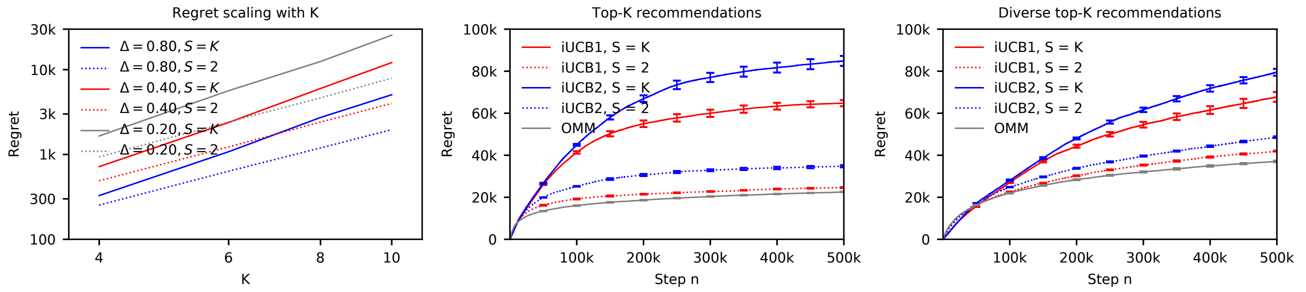
the bounds in Theorems 2 and 4 become $O(SL\Delta^{-1} \log n)$, where L is the number of items and $S = 1/\alpha$ is the number of interleaved actions in iUCB to observe each item in the decision set once. We validate this scaling empirically in Section 5.1.

When compared to matroid bandits (Kveton et al., 2014; Talebi and Proutiere, 2016), our regret bounds contain an extra factor of S. This is the *price for being conservative*. Specifically, since iUCB takes S interleaved actions to observe each item in the decision set $\boldsymbol{D}_t$ once, its regret is S times higher than that of the algorithm that can explore $\boldsymbol{D}_t$ in a single action. Note that whenever $\alpha = \Omega(1)$, as at $\alpha = 1/2$, the extra factor of $S = 1/\alpha$ is independent of K and our bounds scale as those in matroid bandits (Kveton et al., 2014; Talebi and Proutiere, 2016).

Finally, by a standard gap-dependent to gap-free reduction, where the gaps are divided in into those that are larger than ε and smaller than ε, and then ε is tuned, we have a gap-free regret bound of $O(S\sqrt{KLn \log n})$. This bound is again at most S times higher than that in matroid bandits (Kveton et al., 2014).

5 Experiments

We conduct two experiments. In Section 5.1, we validate that the regret of iUCB1 grows as suggested by our upper bound in Theorem 2. In Section 5.2, we apply iUCB to two recommendation problems. We also compare it to a non-conservative algorithm OMM (Kveton et al., 2014), which can learn optimal actions in our problems; but also severely violates the conservative constraint in (1).



(a) (b) (c)

Figure 1: **a**. The n-step regret of `iUCB1` in the synthetic problem in Section 5.1 as a function of K. **b**. The regret of `iUCB1`, `iUCB2`, and `OMM` in the top-K recommendation problem in Section 5.2. **c**. The regret of `iUCB1`, `iUCB2`, and `OMM` in the diverse top-K recommendation problem in Section 5.2.

5.1 Regret Scaling

The first experiment validates that the regret of `iUCB1` scales as suggested by our gap-dependent upper bound in Theorem 2. The ground set is $E = [K^2]$ for parameter $K > 0$ and the action set is $\mathcal{B} = \Pi_K(E)$. The i-th entry of weight vector $\boldsymbol{w}_t$, $\boldsymbol{w}_t(i)$, is an independent Bernoulli random variable with mean

$$\bar{w}(i) = 0.5(1 - \Delta \mathbb{1}(i > K))$$

for $\Delta \in (0, 1)$. From the definition of $\bar{w}$, the optimal action is $A^* = [K]$. The default baseline action are the last K items in E, $B_0 = [K^2] \setminus [K^2 - K]$. In this problem, we expect the regret of `iUCB1` to scale as $SK^2\Delta^{-1}$.

We vary K, Δ, and S; and report the n-step regret of `iUCB1` in 100k steps in Figure 1a. The regret is shown in log-log plots as a function of K for three values of Δ and two values of S. We observe two major trends. First, the regret grows as S and K increase, and Δ decreases. This is consistent with our theoretical analysis. Second, the growth rate is as predicted. In particular, when $S = K$ and one decision item is interleaved with $K - 1$ baseline items, the slopes of the plots are close to 3. This confirms the cubic dependence on K when $S = K$. When $S = 2$ and $K/2$ decision items are interleaved with $K/2$ baseline items, the slopes of the plots are close to 2. This confirms the quadratic dependence on K when $S = 2$.

5.2 Recommender System Experiments

In the second experiment, we apply `iUCB` to our motivating problems in Section 2.4. In both problems, we recommend K movies out of L. The attraction of movies is estimated from the *MovieLens 1M dataset* (Lam and Herlocker, 2016), where 6 thousand users give one million ratings to 4 thousand movies.

Our learning problems are formulated as follows. The set E are 200 movies from the MovieLens dataset. The set is partitioned as $E = \bigcup_{i=1}^{10} E_i$, where E_i are 20 most popular movies in the i-th most popular MovieLens movie genre that are not in $E_1, \dots, E_{i-1}$. The weight of item e at time t, $\boldsymbol{w}_t(e) \in \{0, 1\}$, indicates that item e attracts the user at time t. We set it as $\boldsymbol{w}_t(e) = 1$ if and only if the user rated item e in our dataset. This indicates that the user watched movie e before, perhaps because the movie was attractive. The user at time t is drawn randomly from all MovieLens users. The objective of the learning agent is to learn a set of items with the highest expected attraction over all users.

We study two recommendation problems. The first problem is *top-K recommendation* in Section 2.4, where $K = 10$. The exchangeable action set is $\mathcal{B} = \Pi_K(E)$, all sets of size K from E. The optimal action A^* are 10 most attractive movies. The default baseline action B_0 are the 11th to the 20th most attractive movies. We choose B_0 in this way because existing baseline policies tend to perform well.

The second problem is *diverse top-K recommendation* in Section 2.4, where $K = 10$. The exchangeable action set is defined as in (3), where each $\mathcal{P}_i$ is associated with movie group E_i. The optimal action A^* is the set of most attractive movies from all E_i. The default baseline action B_0 is the set of second most attractive movies from all E_i. Again, we choose B_0 in this way because existing baseline policies tend to perform well.

Our results are reported in Figures 1b and 1c. We observe several trends across both problems. First, the regret of all algorithms is concave, which shows that they learn better policies over time. Second, the regret of `iUCB2` is higher than that of `iUCB1`. This is because `iUCB2` does not know the values of default baseline items B_0, while `iUCB1` does. Since `iUCB2` has to estimate these values, it is more conservative and learns slower. Second, the regret increases with S. For instance, in Figure 1b, the regret at $S = K$ is almost twice as high as that at $S = 2$. This is expected since the former setting is more conservative. In particular, at $S = K$, one decision item is interleaved with $K - 1$ baseline items; while at $S = 2$, and $K/2$ decision items are interleaved with $K/2$ baseline items.

Finally, we note that OMM achieves the lowest regret. But it also violates our conservative constraints. For instance, at $S = K$, iUCB1 and iUCB2 violate none of the constraints in (1). On the other hand, OMM violates more than 16k and 158k constraints in Figures 1b and 1c, respectively, on average in 500k steps. This is one violated constraint in every three actions in the latter problem. We also note that at $S = 2$, the regret of iUCB1 approaches that of OMM. This indicates that reasonably conservative constraints, such as that one half of the recommended items are at least as good as default baseline items, can be satisfied without a major impact on regret.

6 Related Work

The idea of controlled exploration in multi-armed bandits is not new. Wu et al. (2016) studied conservatism in multi-armed bandits, where the *cumulative reward* of the learning agent is constrained to be at least $1 - \alpha$ fraction of that of the default action. In our setting, this means that the agent can take a disastrous action, with many suboptimal items, every $1/\alpha$ steps. In contrast, our *per-step constraint* in (1) prohibits this design and such disastrous actions. However, note that our setting and algorithms are less general, as they only apply to combinatorial action spaces.

A/B testing (Siroker and Koomen, 2013) can also solve constrained exploration problems. When the new and default actions are chosen randomly with probabilities α and $1 - \alpha$, respectively, the expected reward is no worse than $1 - \alpha$ fraction of that of the default action. Since this constraint is *in expectation*, A/B testing can take disastrous actions occasionally. In comparison, we satisfy our constraint in (1) *with a high-probability* at all times, and strictly avoid disastrous actions.

Online learning with matroids was introduced by Kveton et al. (2014) and later studied by Talebi and Proutiere (2016). These works do not consider any notion of conservatism. A naive generalization of these works to conservatism is problematic, as discussed at the beginning in Section 3.

Kazerouni et al. (2017) studied conservatism in linear bandits. Similarly to Wu et al. (2016), their constraint is *cumulative*. Furthermore, the time complexity of their algorithm increases with time when the expected reward of the baseline policy is unknown. In comparison, iUCB is both computationally and sample efficient.

Bastani et al. (2017) studied contextual bandits and proposed diversity assumptions on the environment. Intuitively, if the context varies a lot over time, the environment explores on behalf of the learning agent, and the agent does not have to explore. In comparison, we actively explore in a constrained fashion.

Radlinski and Joachims (2006) proposed randomizing the order of presented items to estimate their relevance in the presence of item and position biases. Their algorithm guarantees that the quality of the presented items is affected minimally. But it does not learn a better policy. The idea of interleaving has been used to evaluate information retrieval systems and Chapelle et al. (2012) validated its efficacy. Chapelle et al. (2012) did not study the problem of learning a better policy. iUCB learns a better policy. While we do not consider item and position biases in this work, we hope to do so in future work.

7 Conclusions

In this paper, we study controlled exploration in combinatorial action spaces using interleaving, and precisely formulate the learning problem in the space of exchangeable actions. Our conservative formulation is more suitable for combinatorial spaces than existing notions of conservatism. We propose an algorithm for solving our problem, iUCB, and prove gap-dependent upper bounds on its regret. iUCB exploits the idea of interleaving and can evaluate a disastrous action without ever taking it.

We leave open several questions of interest. First, how large is the class of exchangeable action spaces? We provide two examples of such spaces in Section 2.3 in relation to top-K and diverse top-K recommendations. A fairly large class of exchangeable action spaces is the class of *strongly base-orderable matroids*. The action spaces in top-K and diverse top-K recommendation problems belong to this class.

The clicks are typically biased due to the position of the item and other recommended items (Chuklin et al., 2015). Therefore, in general, it seems hard to compute unbiased estimates of item relevances with interleaving. This may be possible in some models. For instance, in the cascade model, existing algorithms for online learning to rank compute unbiased estimates of item relevances from biased clicks (Kveton et al., 2015a; Combes et al., 2015; Katariya et al., 2016; Zong et al., 2016; Li et al., 2016). It also may be possible to compute biased estimators with the right bias, that a more relevant item never appears to be less relevant than a less relevant item (Zoghi et al., 2017; Lattimore et al., 2018). We leave this for future work.

Third, we not only require the action space to be exchangeable, but also need to construct the bijection in Definition 1. The construction is straightforward in uniform and partition matroids in our experiments.

We also leave open the question of a lower bound. Finally, we wish to highlight that new ideas in our analysis of iUCB can be used to greatly simplify the original analysis of OMM in Kveton et al. (2014).

References

Gediminas Adomavicius and Alexander Tuzhilin. Context-aware recommender systems. In *Recommender systems handbook*, pages 191–226. Springer, 2015.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

Hamsa Bastani, Mohsen Bayati, and Khashayar Khosravi. Exploiting the natural exploration in contextual bandits. *arXiv preprint arXiv:1704.09011*, 2017.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013.

Andrei Z Broder. Computational advertising and recommender systems. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 1–2. ACM, 2008.

Olivier Chapelle, Thorsten Joachims, Filip Radlinski, and Yisong Yue. Large-scale validation and analysis of interleaved search evaluation. *ACM Transactions on Information Systems (TOIS)*, 30(1):6, 2012.

Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework and applications. In *International Conference on Machine Learning*, pages 151–159, 2013.

Aleksandr Chuklin, Ilya Markov, and Maarten de Rijke. Click models for web search. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 7(3):1–115, 2015.

Richard Combes, Stefan Magureanu, Alexandre Proutiere, and Cyrille Laroche. Learning to rank: Regret lower bounds and efficient algorithms. In *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, 2015.

Paolo Cremonesi, Franca Garzotto, Sara Negro, Alessandro Vittorio Papadopoulos, and Roberto Turrin. Looking for “good” recommendations: A comparative evaluation of recommender systems. In *IFIP Conference on Human-Computer Interaction*, pages 152–168. Springer, 2011.

Mukund Deshpande and George Karypis. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*, 22(1):143–177, 2004.

Jack Edmonds. Matroids and the greedy algorithm. *Mathematical programming*, 1(1):127–136, 1971.

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

Aurélien Garivier and Olivier Cappé. The kl-ucb algorithm for bounded stochastic bandits and beyond. In *Proceedings of the 24th annual Conference On Learning Theory*, pages 359–376, 2011.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. Pac subset selection in stochastic multi-armed bandits. In *ICML*, volume 12, pages 655–662, 2012.

Sumeet Katariya, Branislav Kveton, Csaba Szepesvari, and Zheng Wen. DCM bandits: Learning to rank with multiple clicks. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 1215–1224, 2016.

Abbas Kazerouni, Mohammad Ghavamzadeh, Yasin Abbasi, and Benjamin Van Roy. Conservative contextual linear bandits. In *Advances in Neural Information Processing Systems*, pages 3913–3922, 2017.

Branislav Kveton, Zheng Wen, Azin Ashkan, Hoda Eydgahi, and Brian Eriksson. Matroid bandits: Fast combinatorial optimization with learning. *arXiv preprint arXiv:1403.5045*, 2014.

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*, 2015a.

Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvari. Tight regret bounds for stochastic combinatorial semi-bandits. In *Artificial Intelligence and Statistics*, pages 535–543, 2015b.

Shyong Lam and Jon Herlocker. MovieLens Dataset. http://grouplens.org/datasets/movielens/, 2016.

Tor Lattimore, Branislav Kveton, Shuai Li, and Csaba Szepesvari. TopRank: A practical algorithm for online stochastic ranking. In *Advances in Neural Information Processing Systems 31*, pages 3949–3958, 2018.

Shuai Li, Baoxiang Wang, Shengyu Zhang, and Wei Chen. Contextual combinatorial cascading bandits. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 1245–1253, 2016.

Filip Radlinski and Thorsten Joachims. Minimally invasive randomization for collecting unbiased preferences from clickthrough. In *Logs, Proceedings of the 21st National Conference on Artificial Intelligence (AAAI*. Citeseer, 2006.

Paul Resnick and Hal R Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.

Tobias Schnabel, Paul N Bennett, Susan T Dumais, and Thorsten Joachims. Short-term satisfaction and long-term coverage: Understanding how users tolerate algorithmic exploration. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 513–521. ACM, 2018.

Dan Siroker and Pete Koomen. *A/B testing: The most powerful way to turn clicks into customers*. John Wiley & Sons, 2013.

Mohammad Sadegh Talebi and Alexandre Proutiere. An optimal algorithm for stochastic matroid bandit optimization. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 548–556. International Foundation for Autonomous Agents and Multiagent Systems, 2016.

Yifan Wu, Roshan Shariff, Tor Lattimore, and Csaba Szepesvári. Conservative bandits. In *International Conference on Machine Learning*, pages 1254–1262, 2016.

Masrour Zoghi, Tomas Tunys, Mohammad Ghavamzadeh, Branislav Kveton, Csaba Szepesvari, and Zheng Wen. Online learning to rank in stochastic click models. In *International Conference on Machine Learning*, pages 4199–4208, 2017.

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 32nd Conference on Uncertainty in Artificial Intelligence*, 2016.

A Appendix

Lemma 1. *Let $\mathcal{E}_t$ be the good event in (6). Then*

$$\mathbb{P}\left(\bigcup_{t=1}^{n/S} \bar{\mathcal{E}}_t\right) \leq \sum_{t=1}^{n/S} \mathbb{E}\left[\mathbb{1}\left(\bar{\mathcal{E}}_t\right)\right] \leq \frac{2L}{Sn}\,.$$

Proof. From the definition of our confidence intervals and Hoeffding's inequality (Boucheron et al., 2013),

$$\mathbb{P}(|\bar{w}(e) - \hat{\boldsymbol{w}}_s(e)| \geq c_{n,s}) \leq 2\exp[-3\log n]$$

for any $e \in E$, $s \in [n]$, and $t \in [n]$. Therefore,

$$\mathbb{P}\left(\bigcup_{t=1}^{n/S} \bar{\mathcal{E}}_t\right) \leq \sum_{t=1}^{n/S} \mathbb{P}(\bar{\mathcal{E}}_t) \leq \sum_{t=1}^{n/S} \sum_{e \in E} \sum_{s=1}^{tS} \mathbb{P}(|\bar{w}(e) - \hat{\boldsymbol{w}}_s(e)| \geq c_{n,s}) \leq 2 \sum_{e \in E} \frac{1}{Sn}\,.$$

This concludes our proof. □

Lemma 2. *Let A be the maximum weight action with respect to weights w. Let B be any action and let $\rho : A \to B$ be the bijection in Definition 1. Then*

$$\forall a \in A : w(a) \geq w(\rho(a))\,.$$

Proof. Fix $a \in A$ and let $b = \rho(a)$. By Definition 1, $A_b^a = A \setminus \{a\} \cup \{b\} \in \mathcal{B}$. Now note that A is the maximum weight action with respect to w. Therefore,

$$w(a) - w(b) = \sum_{e \in A} w(e) - \sum_{e \in A_b^a} w(e) \geq 0\,.$$

This concludes our proof. □

A.1 `iUCB1`: Known Baseline Means

Theorem 1. `iUCB1` *satisfies* (1) *jointly at all times $t \in [n]$ with probability of at least $1 - 2L/(Sn)$.*

Proof. At time t, the baseline set $\boldsymbol{B}_t$ is the maximum weight action with respect to $\boldsymbol{v}_t$. Therefore, by Lemma 2, there exists a bijection $\boldsymbol{\rho} : \boldsymbol{B}_t \to B_0$ such that

$$\forall b \in \boldsymbol{B}_t : \boldsymbol{v}_t(b) \geq \boldsymbol{v}_t(\boldsymbol{\rho}(b))\,.$$

From the definition of $\boldsymbol{v}_t$, $\boldsymbol{v}_t(\boldsymbol{\rho}(b)) = \bar{w}(\boldsymbol{\rho}(b))$ for any $b \in \boldsymbol{B}_t$, and thus

$$\forall b \in \boldsymbol{B}_t : \boldsymbol{v}_t(b) \geq \bar{w}(\boldsymbol{\rho}(b))\,.$$

Now suppose that event $\mathcal{E}_t$ in (6) happens. Then $\bar{w}(e) \geq \boldsymbol{L}_t(e)$ for any $e \in E$, and it follows that

$$\forall b \in \boldsymbol{B}_t : \bar{w}(b) \geq \bar{w}(\boldsymbol{\rho}(b))\,.$$

Since any action at time t contains $K(1-\alpha)$ items from $\boldsymbol{B}_t$, the constraint in (1) is satisfied when event $\mathcal{E}_t$ happens. Finally, we prove that $\mathbb{P}(\cup_t \bar{\mathcal{E}}_t) \leq 2L/(Sn)$ in Lemma 1. Therefore, $\mathbb{P}(\cap_t \mathcal{E}_t) \geq 1 - 2L/(Sn)$. This concludes our proof. □

Lemma 3. *For any $e^* \in A^*$ and $e \in \boldsymbol{D}_t$ such that $e = \boldsymbol{\pi}_t(e^*)$,*

$$\Delta_{e,e^*} \leq 2c_{n,\boldsymbol{T}_{t-1}(e)}\,. \tag{17}$$

Proof. Since the decision set $\boldsymbol{D}_t$ is chosen based on UCBs, we have that $\boldsymbol{U}_t(e) \geq \boldsymbol{U}_t(e^*)$. This leads to

$$\bar{w}(e) + 2c_{n,\boldsymbol{T}_{t-1}(e)} \geq \hat{\boldsymbol{w}}_{t-1}(e) + c_{n,\boldsymbol{T}_{t-1}(e)} = \boldsymbol{U}_t(e) \geq \boldsymbol{U}_t(e^*) \geq \bar{w}(e^*)\,,$$

which is our claim. □

Lemma 4. *For any item $e \in \boldsymbol{D}_t \setminus A^*$,*

$$\boldsymbol{T}_{t-1}(e) \leq \frac{6}{\Delta_{e,\min}^2} \log n,$$

where $\Delta_{e,\min}$ is defined in (7).

Proof. Since Lemma 3 holds for any $e \in \boldsymbol{D}_t$ and $e^* \in A^*$ such that $e^* = \boldsymbol{\pi}_t(e)$, if we substitute the expression for $c_{n,\boldsymbol{T}_{t-1}(e)}$ from (5) in (17), we get that

$$\Delta_{e,\min} \leq \Delta_{e,e^*} \leq 2\sqrt{\frac{1.5 \log n}{\boldsymbol{T}_{t-1}(e)}}.$$

This equation after rearrangement proves the lemma.

□

Lemma 5. *For any $e \in \boldsymbol{D}_t$ and $e' \in \boldsymbol{B}_t$ such that $e' = \boldsymbol{\sigma}_t(e)$,*

$$\Delta_{e',e} \leq 2c_{n,\boldsymbol{T}_{t-1}(e)}\,. \tag{18}$$

Proof. Since the baseline set $\boldsymbol{B}_t$ is chosen based on LCBs, we have that $\boldsymbol{L}_t(e') \geq \boldsymbol{L}_t(e)$. This leads to

$$\bar{w}(e') \geq \boldsymbol{L}_t(e') \geq \boldsymbol{L}_t(e) \geq \bar{w}(e) - 2c_{n,\boldsymbol{T}_{t-1}(e)}\,,$$

which is our claim. □

Lemma 6. *For any item $e^* \in A^* \cap \boldsymbol{D}_t \setminus \boldsymbol{B}_t$,*

$$\boldsymbol{T}_{t-1}(e^*) \leq \frac{6}{\Delta_{e^*,\min}^{*2}} \log n,$$

where $\Delta_{e^*,\min}^*$ is defined in (8).

Proof. We first claim that for any $e^* \in \boldsymbol{D}_t$, $e^* = \boldsymbol{\pi}_t(e^*)$. Assume otherwise. Then $A^* \setminus \{e^*\} \cup \{\boldsymbol{\pi}_t(e^*)\}$ is an action (by Definition 1) of size $(K-1)$, which contradicts the fact that all actions have the same cardinality K.

Lemma 5 holds for any $e \in \boldsymbol{D}_t$ and $e' \in \boldsymbol{B}_t$ such that $e' = \boldsymbol{\sigma}_t(e)$. We now set $e = e^*$ and substitute $c_{n,\boldsymbol{T}_{t-1}(e)}$ from (5) in (18) to obtain that

$$\Delta_{e^*,\min}^* \leq \Delta_{e',e^*} \leq 2\sqrt{\frac{1.5 \log n}{\boldsymbol{T}_{t-1}(e^*)}}.$$

This equation after rearrangement proves the lemma.

□

A.2 `iUCB2`: Unknown Baseline Means

Theorem 3. `iUCB2` *satisfies (1) jointly at all times $t \in [n]$ with probability of at least $1 - 2L/(Sn)$.*

Proof. At time t, the baseline set $\boldsymbol{B}_t$ is the maximum weight action with respect to $\boldsymbol{v}_t$. Therefore, by Lemma 2, there exists a bijection $\boldsymbol{\rho} : \boldsymbol{B}_t \to B_0$ such that

$$\forall b \in \boldsymbol{B}_t : \boldsymbol{v}_t(b) \geq \boldsymbol{v}_t(\boldsymbol{\rho}(b))\,.$$

Now we consider two cases. First, suppose that $b \in B_0$. Then by Lemma 2, $b = \boldsymbol{\rho}(b)$, and $\bar{w}(b) \geq \bar{w}(\boldsymbol{\rho}(b))$ from our assumption. Second, suppose that $b \notin B_0$. Then from $\boldsymbol{v}_t(b) = \boldsymbol{L}_t(b)$ and $\boldsymbol{v}_t(\boldsymbol{\rho}(b)) = \boldsymbol{U}_t(\boldsymbol{\rho}(b))$, and

$$\bar{w}(b) \geq \boldsymbol{L}_t(b) \geq \boldsymbol{U}_t(\boldsymbol{\rho}(b)) \geq \bar{w}(\boldsymbol{\rho}(b))$$

under event $\mathcal{E}_t$. Since any action at time t contains $K(1-\alpha)$ items from $\boldsymbol{B}_t$, the constraint in (1) is satisfied when event $\mathcal{E}_t$ happens.

Finally, we prove that $\mathbb{P}(\cup_t \bar{\mathcal{E}}_t) \leq 2L/(Sn)$ in Lemma 1. Therefore, $\mathbb{P}(\cap_t \mathcal{E}_t) \geq 1 - 2L/(Sn)$. This concludes our proof. □

Lemma 7. *For any $e \in \boldsymbol{D}_t$ and $e' \in \boldsymbol{B}_t \cap B_0$ such that $e' \in \boldsymbol{\sigma}_t(e)$,*

$$\Delta_{e',e} \leq 4c_{n,\boldsymbol{T}_{t-1}(e)}\,. \tag{19}$$

Proof. For items $e' \in \boldsymbol{B}_t \cap B_0$, we have that $\boldsymbol{U}_t(e') \geq \boldsymbol{L}_t(e)$. This gives us

$$\bar{w}(e') + 2c_{n,\boldsymbol{T}_{t-1}(e')} \geq \boldsymbol{U}_t(e') \geq \boldsymbol{L}_t(e) \geq \bar{w}(e) - 2c_{n,\boldsymbol{T}_{t-1}(e)}$$

This implies that

$$\Delta_{e',e} \leq 2c_{n,\boldsymbol{T}_{t-1}(e)} + 2c_{n,\boldsymbol{T}_{t-1}(e')}. \tag{20}$$

An item eliminated from the baseline set $\boldsymbol{B}_t$ is never re-introduced in the baseline set. Since $e' \in \boldsymbol{B}_t \cap B_0$, it must have never been eliminated from the baseline set. The maximum number of times e can be played is by including it in every decision set $\boldsymbol{D}_t$. In any round, since the baseline items are played $(S-1)$ times the decision set counterparts, and $S \geq 2$, we have that $\boldsymbol{T}_{t-1}(e') \geq (S-1)\boldsymbol{T}_{t-1}(e) \geq \boldsymbol{T}_{t-1}(e)$, which implies that

$$c_{n,\boldsymbol{T}_{t-1}(e')} \leq c_{n,\boldsymbol{T}_{t-1}(e)}.$$

Substituting in (20),

$$\Delta_{e',e} \leq 4c_{n,\boldsymbol{T}_{t-1}(e)}.$$

□

Lemma 8. *For any item $e^* \in A^* \cap \boldsymbol{D}_t \setminus \boldsymbol{B}_t$,*

$$\boldsymbol{T}_{t-1}(e^*) \leq \frac{12}{\Delta^{*2}_{e^*,\min}} \log n,$$

where $\Delta^*_{e^*,\min}$ is defined in (8).

Proof. The proof is identical to the proof of Lemma 6, except for an extra factor of 2 that appears because (19) contains an extra 2 as compared to (18). □